# International Journal on Advances in Networks and Services

IARIA

2009 vol. 2 nr. 1

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



Sponsored by IARIA
www.iaria.org

**International Journal On Advances in Networks and Services**
**Volume 2, Number 1, 2009**

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

**Wireless and Mobile Communications**

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

**Sensors**

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

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

**Underwater Technologies**

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

**Energy Optimization**

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

**Mesh Networks**

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

**Centric Technologies**

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

**Multimedia**

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

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

**Additional reviews by:**

- Yongning Tang, Illinois State University, USA

**International Journal On Advances in Networks and Services**
**Volume 2, Number 1, 2009**

## Foreword

The volume carries on the objectives of the journal and publishes papers covering a broad range of current issues in networking and network services. Particularly, the nine papers selected for this volume among twenty four submissions discuss problems and their proposed solutions from the areas of network routing, Quality of Service, Internet traffic analysis, and wireless local area and sensor networks. Routing is still one of the most important issues in networking along with the trends in optical and electronic switching and traffic engineering. We included two papers in this topic. The paper "Implementing and Testing a Formal Framework for Constraint-Based Routing over Scale-free Networks" describes a formal model to represent and solve the Constraint-Based Routing problem in networks. They utilize the Soft Constraint Logic Programming (SCLP) framework as a programming environment to solve the routing problem. The other paper, "Optimum routing and forwarding path arrangement in bufferless Data Vortex networks" studies the different routing and forwarding path arrangements in Data Vortex networks. While current optical fiber networks provide a vast amount of bandwidth, they lack the capabilities in optical processing and optical buffering techniques to implement most existing switching and routing architectures. A Data Vortex network is uniquely designed with three-dimensional arrangements of the routing nodes and allows for bufferless operation using deflection based routing. The minimal routing decision can be implemented electronically within distributed routing nodes. The paper proposes a modified Data Vortex network architecture based on general k-ary decoding routing. Different cases are compared to study the optimum layout.

Quality of Service (QoS) is of increasing importance in all networks, especially in next generation networks. QoS deals with the strict management of network traffic in that guarantees can be made for successful packet delivery between communicating parties of the Internet. The authors of the paper "Dynamic End-to-End QoS Provisioning and Service Composition over Heterogeneous Networks" propose a multi-service, multi-technology model based on Bandwidth Broke to provide QoS inside each domain and to ensure it on the end to end data path.

Analysis of the Internet traffic has been a focal point of research in network modeling and simulation in terms of the traffic burstiness or self-similarity. It is an extremely important question for network designers and engineers to determine if the planned or existing network is vulnerable to the harmful consequences of bursty traffic and predict the anticipated trends in network traffic. The paper "Time Dependent Lévy Flights Models for Internet Traffic" analyzes and compares the burstiness of traffic traces captured in the Internet backbone in 2003 and 2008 in terms of interarrival time. The paper illustrates the tendency of Internet traffic burstiness in recent years.

Considering the basic human desire for freedom of mobility, it is not a surprise that the demand for wireless networks is enormous. With the introduction of sensor, cellular and WLAN communications, we have seen an increasing demand for wireless services. As a result, there has been a dramatic shift in network usage: the number of wireless subscribers has exceeded the number of landline users. The last five papers correspond to the wireless networks category. The paper entitled "Multi-band Gigabit Mesh Networks: Opportunities and Challenges" discusses the benefits of multi-band gigabit mesh networks that can potentially satisfy the requirements set out in the IEEE 802.11ad proposal including the diversity gain, range extension and spatial reuse gain. The authors present the results of the simulation of a 2-hop 60 GHz mesh for an office WLAN to demonstrate that the spatial reuse gain can be very significant in 60 GHz mesh networks. The authors of the second paper in the wireless topic entitled "Automated Selection of Computing Elements for Energy Efficient Wireless Sensor Networks" propose a new statistical technique for energy consumption estimation for a specific application on various platforms. They have empirically verified the methodology on various classes of embedded processors commonly used in sensor nodes. The experimental evaluation results on various platforms will help to understand the implications of using different processing elements and their effects on the lifetime of a wireless sensor network. The paper entitled "Adjustable Multi-Sector Cellular Base Station Antenna" presents a dual beam array (DBA) for higher-order sectorization of a cellular site providing a means of cost-effective increase in network capacity without the use of additional frequency spectrum. The paper "Robustness in Sensor Networks: Difference Between Self-Organized Control and Centralized Control" studies the robustness of self-organized control against a wide range of perturbations by comparing it with centralized control in sensor networks. The authors demonstrate through simulations that self-organized control maintains the functionality of its data collection even in a variety of perturbations. The last paper "Context Modeling for Cross-layer Context Aware Adaptations" proposes a new architecture for cross-layer interactions as an alternative solution to existing layered protocol stack in wireless networks for improved delivery of real-time traffic. As example architecture, a context aware adaptive multi-homed Mobile IP environment is also discussed.

We hope the readers will find the published articles inspiring and constructive that will contribute to the discovery of new findings in the covered research areas. We thank the authors for their submissions and the reviewers for devoting their time and effort to this issue. We are also grateful to Professor Petre Dini for his energy and time to coordinate the works of the authors and reviewers.

*Tibor Gyires, Editor-in-Chief*

**International Journal On Advances in Networks and Services**
**Volume 2, Number 1, 2009**

**CONTENTS**

# Performance Evaluation for Different Arrangements of Routing and Forwarding Paths within Bufferless Data Vortex Networks

Qimin Yang

Harvey Mudd College, Engineering Department, Claremont, CA 91711,
qimin_yang@hmc.edu

## Abstract

*Extended performance evaluation is carried out on Data Vortex optical interconnection networks with different routing and forwarding path arrangements [1]. A modified Data Vortex network architecture based on general k-ary decoding routing at each node has been proposed and different cases are compared in search for the optimum layout. For bufferless implementation, the original Data Vortex networks based on binary decoding stages are shown to achieve the best combined routing performance in throughput and latency. We specifically focus on the performance comparison between the binary decoding (k=2) and 4-ary decoding (k=4) cases to illustrate the different network behaviors. The results provide insight to how the different routing and forwarding path arrangements affect the overall network performance in throughput and latency. The binary Data Vortex networks outperform 4-ary networks even though a much smaller number of cylinder levels are required in a 4-ary network. There is only slight reduction in the average packet latency within the 4-ary network, while its deflection induced traffic backpressure under bufferless operation could greatly limit the throughput and make it less desirable. Future work may include such performance evaluation when extra buffering is available at routing nodes.*

**Keywords:** Packet Switch, Interconnection Network, Optical Network, Data Vortex, Deflection.

## 1. Introduction

Packet switched interconnection networks are key subsystems in high capacity data communication systems and multi-processor supercomputer systems [2-3]. As I/O ports or high-speed processors that are connected through such networks upgrade dramatically, the interconnection networks must be able to handle very high data rates (tens of Gbit/s) as well as to support a large number of communication ports (on the order of thousands). The key network performance such as throughput and latency must be able to sustain as such networks scale to larger sizes and higher bit rates. A natural way to achieve the higher bandwidth is using optical packet switched interconnections. Current optical fiber and optical amplifier technologies provide enormous operation bandwidth with hundreds of densely packed wavelength division multiplexing (WDM) channels each running at bit rate of tens of Gibit/s. It is thus rather easy to accomplish the high transmission bandwidth in optics. On the other hand, there is still very limited capability in optical processing and optical buffering techniques [4-5]. As a result, the main challenge in these interconnection networks is to handle traffic routing and traffic contention. This has led to difficulty in adapting most existing switching architectures for optical implementations. For example Banyan and Butterfly networks are popular and effective as self-routing electrical switching fabrics networks, however it is very challenging to implement them in the optical domain because of the lack of RAM buffering at each node. Even though pure deflection routing (vs. store and forward routing) is possible, Banyan and Butterfly networks require the deflected packet to travel around the network diameter in order to return to an open path that leads to the target output port. Thus the deflection penalty is prohibitively high and it induces large latency as well as poor network throughput in these two-dimensional network topologies [6].

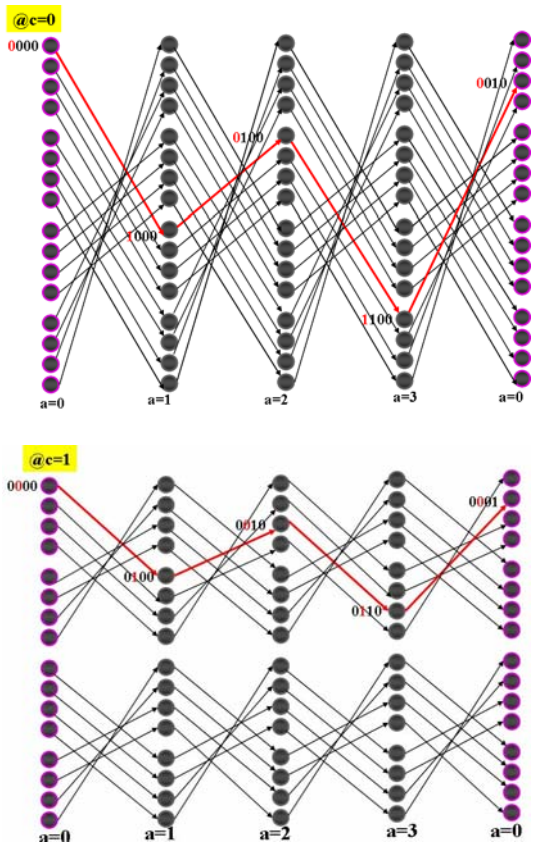
To take advantages of optics while avoiding extensive buffering and processing optically, Data Vortex network architecture is designed to be a great alternative for the purpose and it is particularly suitable for optical system implementation. The network routing performance has been studied extensively in earlier works and its system implementation and physical layer limitations have also been

addressed in a small scale experimental testbed by the research group in Columbia University [7-12]. In this paper, a modified or generalized *k*-ary decoding Data Vortex architecture with multiple header bits decoding stages has been proposed as a potential implementation. The proposal is based on the attractive feature of smaller forwarding hops in the higher *k*-ary decoding Data Vortex networks. Due to different arrangements of routing and forwarding paths, it is essential to study the combined routing performance under different traffic conditions in these extended network architectures.

The organization of the paper is as follows: Section 2 describes the background of the original Data Vortex network design. Section 3 presents the proposed *k*-ary decoding Data Vortex architecture that is extended from the original binary decoding networks. Section 4 presents the details of the performance evaluation through simulation study. Different network and traffic cases are included to thoroughly study the network behaviors as well as the scalability of the results. The comparison study is only focused on between binary and 4-ary networks due to much higher deflection penalty disadvantage in higher *k* systems. The latency performance is also broken down to routing hops, forwarding hops and deflection hops for the analysis and the traffic distribution among cylinder levels within the networks are presented to support our findings. Finally Section 5 concludes and summarizes the study.

## 2. Background: Original Data Vortex architecture

Data Vortex network is uniquely designed with three dimensional arrangements of the routing nodes. Due to the additional dimension, it allows for bufferless operation using deflection based routing while requiring minimal routing decision that can be implemented electronically within distributed routing nodes. This switching architecture implements a single-packet-routing rule at each node through a traffic control mechanism, and the topology provides multiple open paths to each target address so that deflection routing encounters a much smaller latency penalty (in 2 hops) that is also independent of the network diameter. These network characteristics allow for great network scalability and achieve good throughput and latency performance even for very large network sizes. The bufferless operation offers the simplest possible contention resolution in the optical domain [7-8]. In the physical layer implementation, optical techniques such as dense wavelength division multiplexing (DWDM) are used for achieving ultra high data throughput as well as for simple header bit extraction and decoding. With the available DWDM techniques and the broadband fast switching devices such as Semiconductor Optical Amplifier (SOA), Data Vortex networks allow for relatively short packet (tens of nanoseconds to hundreds of nanoseconds) for efficient operation. This is achieved simply by stacking the data bits along the abundant wavelength channels available within the amplifier bandwidth. Each of the binary header bits uses an additional wavelength channel so that simple and inexpensive filtering and detection can be used for header extraction. More details on physical layer implementation and system limitation can be found in [9-10].



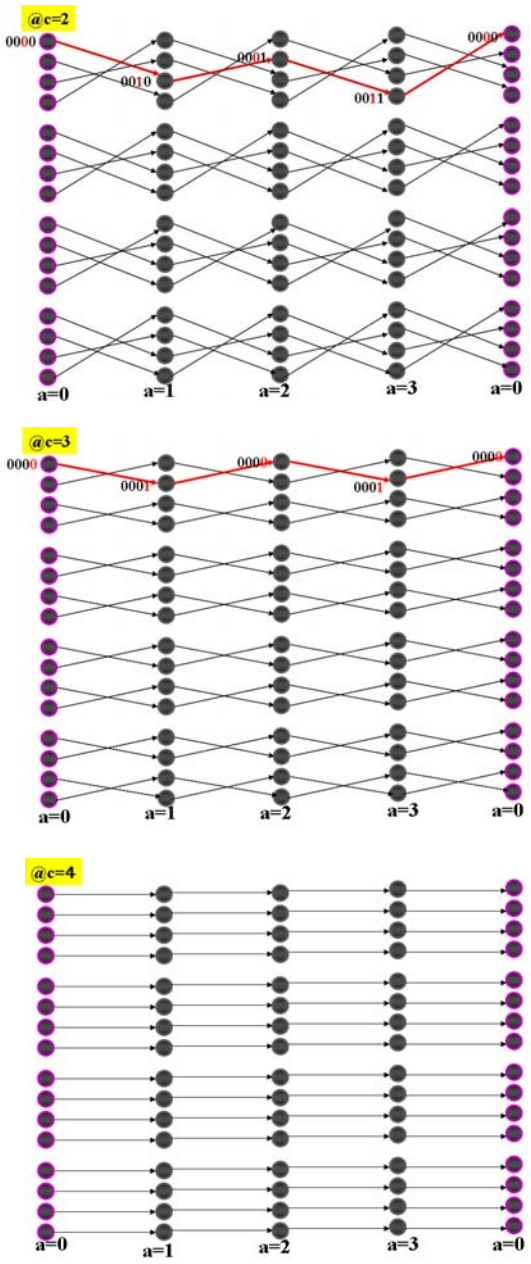



**Fig.1.** Routing nodes and intra-cylinder links within Data Vortex of *A*=4, *H*=16 and *C*=5

The Data Vortex network can be viewed as a multi-stage interconnection network (MIN). The routing nodes are arranged in concentric cylinders with *A*, *H* and $C = \log_2 H + 1$ designating the number of nodes along the angle, height and cylinder respectively. An example of *A*=4, *H*=16 and *C*=5 Data Vortex network is shown in Fig.1 with proper index of angle (a) cylinder (c) and height (h, shown in binary format) location of the nodes. The intra-cylinder link patterns at each cylinder level are specifically shown from the outermost (*c=0*) level to the innermost (*c=4*) level. These links repeat the same pattern from angle to angle for simple implementation and they route a packet back and forth between two height groups, i.e. with specific (highlighted in red) binary bit alternating between "1" and "0". The inter-cylinder links (not shown in Fig.1) are to forward a packet to an inner neighbor cylinder while maintaining its height location, i.e. a node at angle *a*, cylinder *c* and height of *h* will be connected to an inner node at angle *a+1*, cylinder *c+1* and same height of *h*. Therefore, inter-cylinder paths simply appear to be parallel paths between the cylinders [6]. The network is wrapped around as cylinders, therefore, nodes at angle *a=3* is connected back to nodes at angle *a=0* in a network of *A*=4. The last cylinder maintains the exit height position and it serves as an optical buffering stage in case electrical buffers at the output ports. It is also necessary if angular resolution is required for system implementation when only a specific exit angle is connected to the output ports.

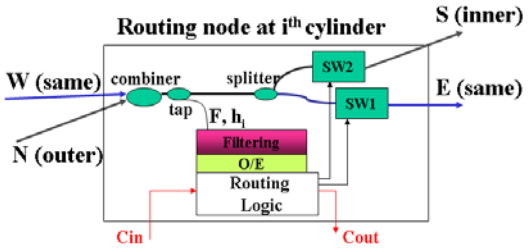



**Fig.2** Routing node implementation at i^th cylinder

The packet routing in the Data Vortex network is operated in a synchronous and slotted fashion. Packet length is typically chosen to be the same as the hop latency for a simple and efficient implementation. Each node directs a single arrival packet to the next node not only based on the packet's target address, but also based on the inner node's traffic. In Fig.2, a routing node at i^th cylinder and its routing logic is shown, and the routing decision is based on the packet frame bit (which tells whether or not a packet is arriving), the corresponding i^th header bit (which matches the i^th bit of node height or not) as well as the electrical control bit sent from its inner competing node (which permits or blocks the outer traffic). Both the frame and header bits can be extracted by a small power tap and through passive filtering and low packet rate optical/electronic (O/E) conversion. The traffic control signal is generated at each node to

properly permit or block the packet of the outer cylinder so that the single-packet-routing rule is always satisfied for bufferless operation for all the routing nodes [11].

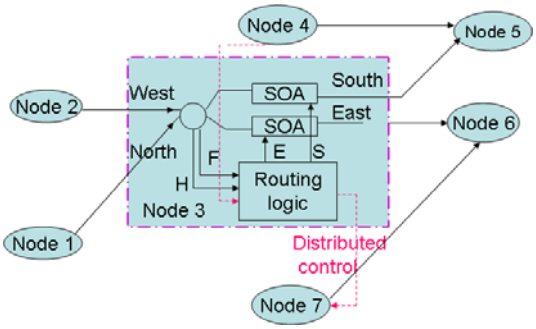



**Fig.3** Distributed control signal among routing nodes

Fig.3 shows the distributed control signals (in dashed line) among the routing nodes on different cylinders. As an example shown, node 3 and node 4 both send packets to node 5, therefore node 4 of the inner cylinder generates a control signal for node 3 to set up the single packet routing rule. Similarly node 3 generates a control signal for its outer cylinder's competing node 7 since they both send packet to node 6. Packets receiving the blocking control are deflected to stay on the current cylinder which acts as virtual buffers with a two hop delay penalty. Once the packet arrives at the correct target, it exits the network in the innermost cylinder. As mentioned, the last cylinder allows for additional optical buffering if necessary. If not all angles at exit are connected to output ports, the last cylinder also deals with angular address resolution of the packet. More details on the angular resolution and choice of angles vs. network height or I/O ports have been discussed in [12].

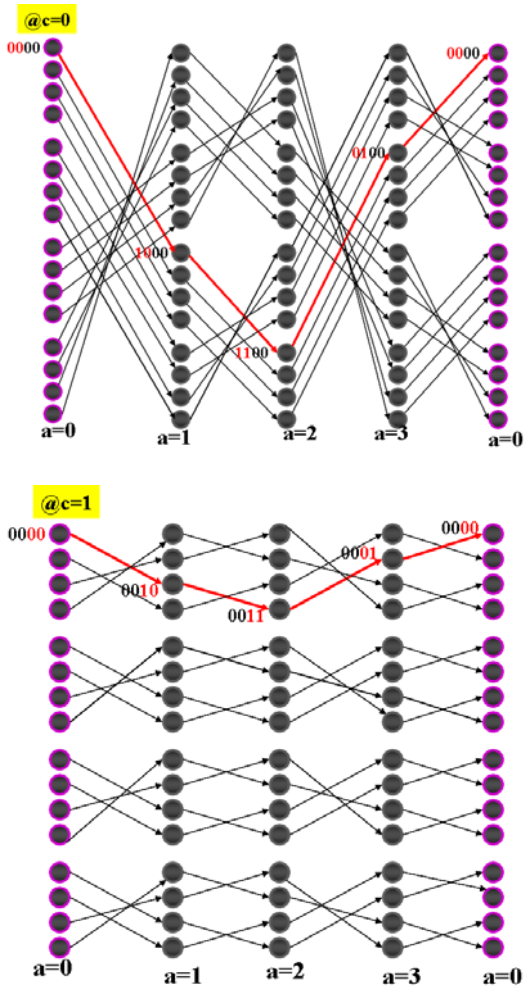
As mentioned above the routing in the Data Vortex network is progressed through a series of binary-tree decoding stages. Each node only decodes a single header bit in the binary target address that corresponds to the node's specific cylinder, and it decides to route the packet further in the current level (current cylinder) or forward the packet to the next level (inner cylinder). Successful forwarding is also dependent on the availability of open control signal due to the traffic condition of the inner cylinder. At different traffic conditions, overall traffic latency is accumulated through routing, deflection and forwarding hops. In this study, we are interested in the optimum arrangement of the routing and forwarding paths. In particular, we explore the potentials of non-binary tree decoding stages.

## 3. Extended to *k*-ary Data Vortex

Since the header decoding is extremely simple with passive filtering and low speed electronics for detection, it is possible to allow for multiple header bits decoding at each stage. In general, we can implement the Data Vortex network using *k*-ary decoding (i.e. $\log_2 k$ header-bit-decoding) at each routing node, where the binary-tree decoding specially uses $k$=2. We are interested in exploring alternative implementations of the Data Vortex network as well as verifying the optimum arrangement of the routing and forwarding paths in these networks. Because each packet spends at least $C$ (number of cylinders) hops just forwarding from input port to output port assuming they do not need to stay on the cylinders for additional routing or for traffic contention induced deflection, it is important to study whether or not the overall latency has been minimized under the original Data Vortex network design, or if alternative arrangement of routing and forward paths would change or improve the latency or routing performance. The optimum layout should achieve a best combined routing performance in data throughput and latency. In extending the binary-tree decoding in the Data Vortex network to general *k*-ary decoding at each stage, we maintain the single packet routing condition and the usage of bufferless routing nodes to facilitate the optical implementation of the networks. Therefore, the performance study in this paper is bounded by such design constraints. Future work may address the performance variation in the case of networks with node buffering capabilities, however such networks must require additional hardware cost and implementation complexity [8].

In the case of binary tree decoding in regular Data Vortex networks, each node decodes one header bit, and the routing on a specified cylinder chooses one out of two groups (upper vs. lower group or specific header bit being "1" vs. "0"), and the deflection latency penalty in the network is two hops. If we extend the concept to general *k*-ary decoding, each stage then decodes $(\log_2 k)$ bits, and each hop along the same cylinder allows for the packet to choose one out of $k$ groups (i.e. specific $(\log_2 k)$ bits alternate

among all possible *k* combinations). When the corresponding $(\log_2 k)$ header bits in the target address are matched with the $(\log_2 k)$ bits in the routing node height address, the packet will proceed to the next inner cylinder if the corresponding traffic control opens the path at the same time. So the same traffic control mechanism is set up for the purpose as that in the original Data Vortex network. Otherwise, the packet stays on the current cylinder for further routing or deflection. Since no buffering is necessary the routing logic is kept as minimal as possible. Compared to binary Data Vortex networks, the deflected packet also undergoes longer delay penalty due to the need to go through all *k* hops to return to the matched height group or to the open path to the next cylinder routing.



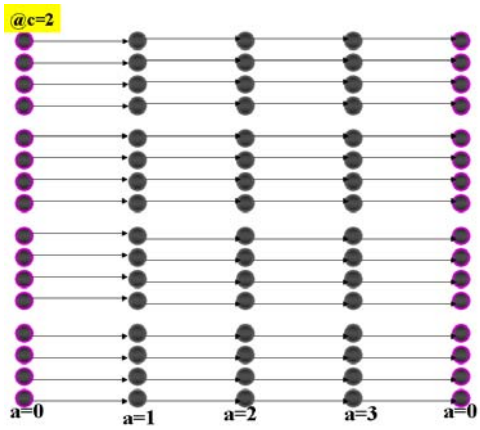




**Fig.4** Routing patterns at each of the three cylinders in a 4-ary decoding Data Vortex network. *A*=4, *H*=16, $C = \log_4 H + 1 = 3$

In order to allow for a complete permutation of *k* groups based on the corresponding $(\log_2 k)$ header bits, the intra-cylinder routing paths are slightly modified from the binary-tree decoding networks. In Fig.4 an example of *4*-ary decoding network is shown where each hop decodes 2 header bits in a network of *A*=4, *H*=16 and the number of cylinders is $C = \log_4 H + 1 = 3$ . Note the interconnection patterns at different cylinders in the binary decoding Data Vortex network are combined and also reversed at proper angles to construct the *4*-ary decoding networks. In such an arrangement, a packet takes *k* hops to go through all *k* possible groups along the cylinder, so the deflection latency penalty would increase to 4 hops in *4*-ry decoding networks, which is the obvious disadvantage of using bigger value of *k*. On the other hand, in extended *k*-ary Data Vortex implementation, the number of cylinders required is $C = \log_k H + 1$, assuming the last cylinder maintains the same height just for output buffering purpose as that of regular Data Vortex network. As a result, the forwarding latency or number of cylinders is much smaller with a larger value of *k*. In this study, we choose *H* so that $\log_k H$ is kept an integer for simplicity. We are specifically interested in *4*-ary networks and its performance comparison with regular binary-decoding Data Vortex network for gain insight of optimum arrangement of routing and forwarding paths. We expect that larger *k* (*k*>4) would cause too much deflection latency and traffic backpressure, which will be verified

in the *4*-ary network study. The node implementation in *4*-ary network is slightly modified with the need to filter and detect 2 header bits in parallel instead of a single header bit and it is shown in Fig.5. While the routing node complexity and speed is kept at the same level, the additional filter and detector increase the hardware cost slightly especially when the number of routing nodes are large.

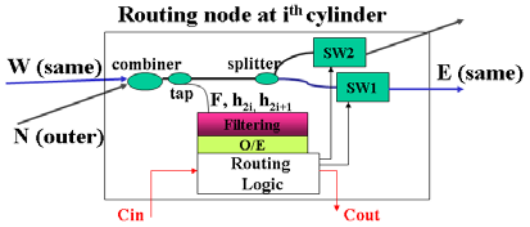



**Fig.5** Routing node implementation at i[th] cylinder in *4*-ary decoding network

## 4. Performance evaluation

In order to compare the network performance with different *k*-ary decoding schemes and find the optimum arrangement, a C/C++ event simulator is specially developed to evaluate these networks with various operation and traffic conditions. In Data Vortex, once the packets are accepted at injection point, they are routed through the network without loss. Therefore, the measure of through performance is calculated as the rate of successful injection. The latency and latency variation statistics is collected by examining the packets that reach the output ports during the simulation. We particularly focus on comparison between the binary-tree decoding network and a network using *k*=4 decoding scheme due to much more significant deflection latency penalty and traffic backpressure in larger *k* cases.

### 4.1 Network Cost and Throughput and Latency Performance

To make a fair comparison, the number of input ports is kept the same and the number of routing nodes and routing links that mainly determine the network cost are either same or in the comparable range. First we studied a regular Data Vortex network, network P with *C*=9 and *H*=256. Packets at only injected at a single angle i.e. $A_{in}$=1 (so number of I/O is the same as the *H*) in a network of *A*=4. We compare its routing performance with two networks Q (*A*=8, *C*=5, *H*=256) and Q' (*A*=7, *C*=5, *H*=256) with *4*-ary decoding both inject using $A_{in}$=1 to keep the same number of I/O ports as that in network P. In both Q and Q', every stage or cylinder level decodes 2 bits and locates 1 out of 4 height groups. Since the number of cylinders *C*=5 is almost half of that in network P, we allow network Q and Q' to have about twice of the network angles for a similar hardware cost. The detailed hardware comparison is listed in Table 1 below. For the same number of I/O ports, the cost of network P is between that of network Q and network Q'. In this study, we assume no angular resolution is required; therefore, packets that arrive at the correct height will immediately exit the network and be converted to electronic domain. Sufficient electrical buffers are assumed to accept any arrival packet at the output port. If additional angular resolution is required, we shall keep that in mind when we examine the results of different angle networks because a larger angle network generally requires additional hops in the last cylinder before packets exit the optical network.

**Table.1** Hardware comparison in network P, Q and Q'

| | k=2 Network P | k=4 Network Q | k=4 Network Q" |
|---|---|---|---|
| Number of I/O, H | 256 | 256 | 256 |
| Number of angles | 4 | 8 | 7 |
| Number of cylinders | 9 | 5 | 5 |
| Number of nodes | 9216 | 10240 | 8960 |

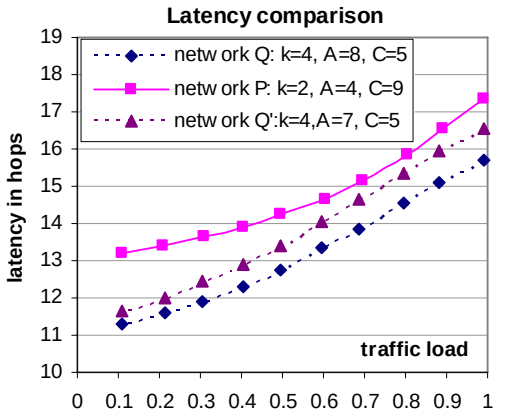



**Fig.6** Latency comparison of network P, Q and Q'

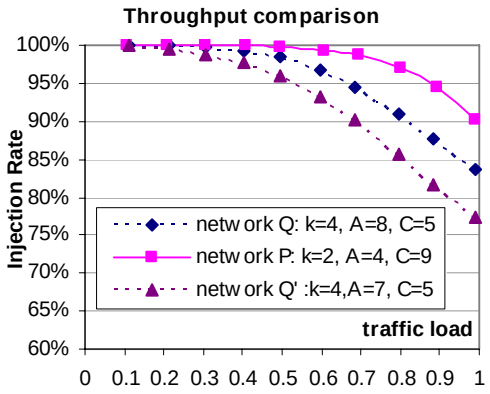



**Fig.7** Throughput comparison of network P, Q and Q'

Fig.6 and Fig.7 have shown the latency and throughput performance comparison of the three networks under different traffic loads respectively. For simplicity, all traffics are random and uniform traffic from each I/O port within this study. As shown, overall the regular binary decoding Data Vortex network (in solid line) outperforms the networks with 4-ary decoding. Even though the number of cylinders is much smaller in network Q and Q', for a similar cost, their throughput performances are significantly worse than network P especially at higher load conditions. The average latency of arrival packets is shown to be slightly better in network Q and Q'. However, if angular resolution is required, then network Q and Q' would also encounter more delay in the last cylinder due to the larger A, therefore the gain in latency is not necessarily noticeable.

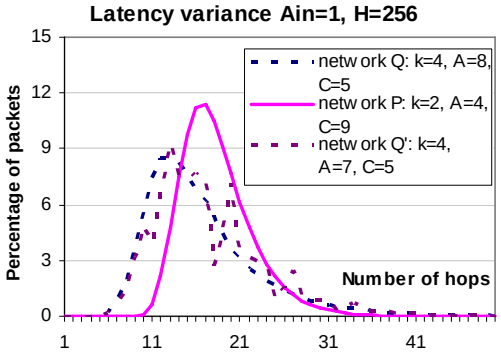



**Fig.8** Latency variance in network P, Q and Q' when load=1.0.

The latency distribution in these three networks is shown in Fig.8 for a case of fully loaded condition, i.e. the applied traffic load=1.0. The results have indicated that network Q and Q' both have pushed part of the packets to much shorter latency (left side of the distribution curve moves earlier), however due to the larger deflection delay which also induces more traffic backpressure, the overall distribution has wider deviation range in these *4*-ary networks, and its tail also extends noticeably further even though the percentage of packets with very large latency is kept small. In comparison, the binary tree decoding Data Vortex has a narrow and confined distribution curve.

## 4.2 Latency Performance in breakdown categories

Next, we studied a case where two networks chosen have the exact same hardware cost for the given same number of I/O ports. Network M in binary tree decoding Data Vortex has *A*=5, *C*=9 and *H*=256 whereas network N using *4*-ary decoding stages has *A*=9, *C*=5 and *H*=256. Both networks use a single angle injection $A_{in}$=1 for the simple comparison. The detailed hardware comparison is shown in Table 2 below. The throughput and latency performance under different traffic loads are shown in Fig.9 and Fig.10 respectively. In the average latency plot in Fig.10, we also plot individual category of delay such as average deflection hops and average routing hops to gain further insights. The forwarding number of hops is not shown but it is fixed to the number of cylinders. As seen, it verifies the better throughput performance in binary Data Vortex network especially at heavier traffic load conditions given the same network cost. The average number of hops in network N is slightly better, however keep in mind it may experience additional hops in angular resolution if compared to that in network M. Fig.10 also shows why it doesn't gain much advantage in latency performance in *4*-ry network even though its forwarding hops (C=5) is 4 hops smaller than that in the binary network (C=9). The number of the routing hops on average is shown to be about 3~4 hops more in network N compared to that in network M. In this case, the deflection hop only counts those hops of staying in the current cylinder due to unavailability of open control signal, and the penalty hops are lumped to the routing hops. Therefore, the results show that the deflection probability is pretty close in two networks under different traffic conditions, however the routing hops is much larger in network N due to larger deflection hop penalty

and generally more hops required to match the permutations even in regular routings.

**Table.2** Hardware in network M and N

| | *k*=2 Network M | *k*=4 Network N |
|---|---|---|
| Number of I/O, *H* | 256 | 256 |
| Number of angles | 5 | 9 |
| Number of cylinders | 9 | 5 |
| Number of nodes | 11520 | 11520 |

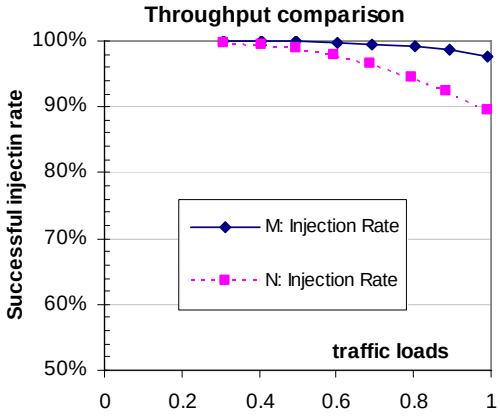



**Fig.9** Throughput comparison in network M and N.

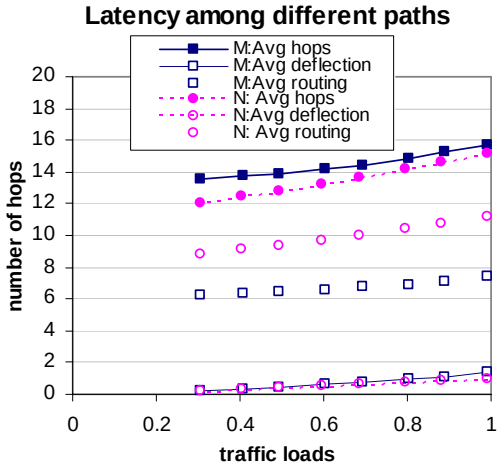



**Fig.10** Latency comparison in network M and N

Fig.11 shows the latency distribution curves of the two networks in the case of load=1.0. Similar to the results in Fig.8, it is found that certain packets go through a smaller number of hops leading to the earlier front trail of the distribution curve, however the overall distribution is wider and the ending tail is longer as a result in the *4*-ry network. On average the routing hops in network N is larger than that in network M, and this cancels out the advantage of less forwarding hops. The latency and the latency distribution performance are also closely related to the throughput performance because of the backpressure effect in traffic. If more packets are pushed through the network in a faster pace, it allows for better throughput, otherwise, the packet occupies the network resource which causes additional deflection and delay. The statistics of all the packets within the network prove that *4*-ary routing does not bring sufficient benefit in latency and throughput performance even with a much smaller number of the forwarding hops.

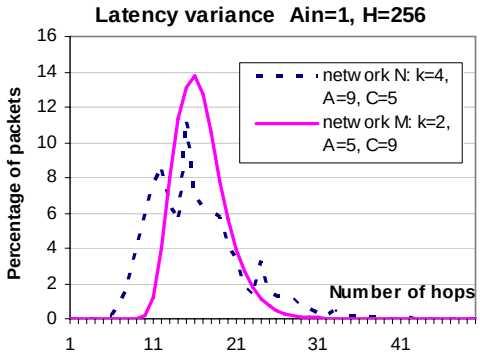



**Fig.11** Latency variation at load=1.0 in network M and N

It is also important to study the performance comparison for different network sizes. For this purpose, we used two additional network $M_2$ (*A*=6, *C*=11, *H*=1024) and $N_2$ (*A*=11, *C*=6, *H*=1024) both with $A_{in}$=1. These two networks are chosen because they share the same hardware cost for the same number of I/O ports while support much larger network height or I/O numbers. The detailed hardware comparison is shown in Table 3.

**Table.3** Hardware in network $M_2$ and $N_2$

| | *k*=2 Network $M_2$ | *k*=4 Network $N_2$ |
|---|---|---|
| Number of I/O, *H* | 1024 | 1024 |
| Number of angles | 6 | 11 |
| Number of cylinders | 11 | 6 |
| Number of nodes | 67584 | 67584 |

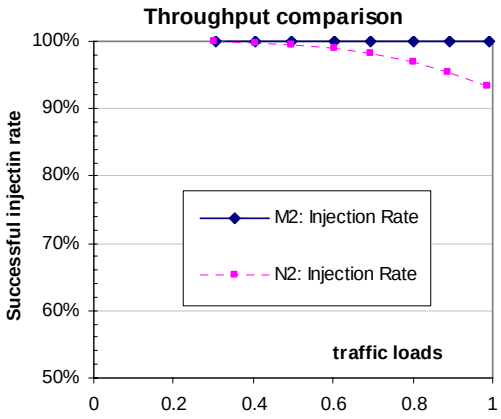



**Fig.12** Throughput comparison in network $M_2$ and $N_2$

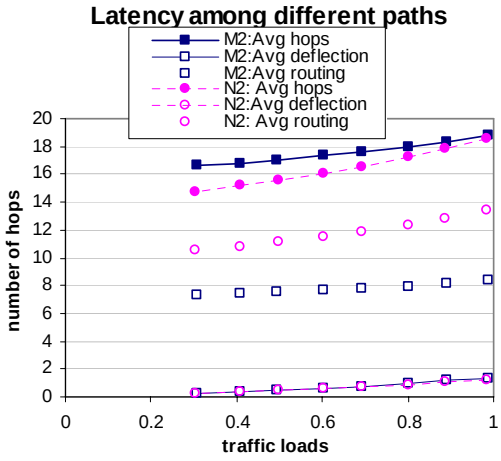



**Fig.13** Latency comparison in network $M_2$ and $N_2$

The routing performances in throughput and latency of $M_2$ and $N_2$ are compared in Fig.12 and Fig.13 respectively. As shown, the performance difference between these two networks follows a very similar trend as that in the comparisons of network M and N. The results have confirmed that at different network sizes and network load conditions, the binary decoding Data Vortex network almost always outperforms the *4*-ary network, mainly due to its inherent small deflection latency penalty and more frequent encounter of open paths between different cylinder levels. Therefore, binary decoding Data Vortex provides the best combined routing performance with the optimum routing and forwarding paths arrangement for bufferless operations.

In addition to the overall routing performance, the latency distribution curves also show some non smoothness in *4*-ry decoding networks which is not present in regular Data Vortex network. It seems to be dependent on the angle of the network. To study this further, several networks with $H$=256 and a varying network angle are compared and shown in Fig.14, and all of them use *4*-ary decoding stages with a single angle injection, i.e. $A_{in}$=1. The distribution curves show that for the fixed number of I/O ports (fixed $H$ and $A_{in}$), a larger network angle result in earlier leading edge of the distribution curve due to relatively more redundancy in the network resource. It is also shown that at angles that are integer multiple of $k$ such as $A$=8 and $A$=12, the distribution curve is rather smooth because of regular distribution of the traffic pattern and equal probability to each node. On the other hand, if $A$ is not an integer multiple of $k$, traffic may not be evenly distributed among groups of nodes, and this leads to multiple peaks in the distribution curves or rather non-smooth distribution. Only when the number of angles $A$ is relatively large, such non-smoothness becomes insignificant due to contributed hardware redundancy. In comparison, in the binary decoding Data Vortex network, the latency distribution smoothness is rather insensitive to the number of network angles whether $A$ is even or odd.

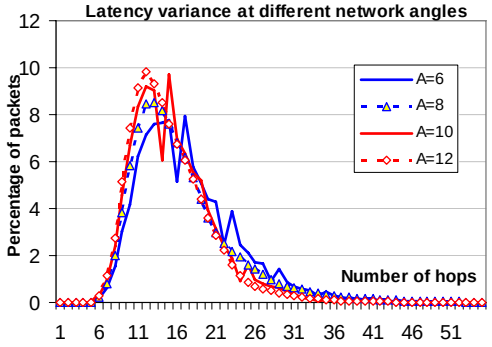



**Fig.14** Smoothness of latency distribution curve at different network angles

## 4.3 Traffic distribution within different cylinder levels

We can gain further insight of the network behaviors by examining the traffic pressure among the cylinders. To compare the two different architectures with different routing and forwarding path arrangements, we record the traffic load or packet count of each specific cylinder at different operation conditions. Packets on the specific cylinder and the ones

entering the cylinders are counted as the packet of the cylinder, and each cylinder's packet count is monitored for comparison study.

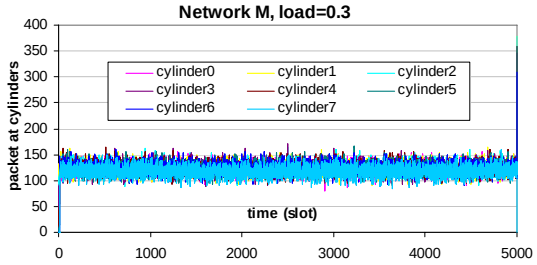



**Fig. 15** Packet count of each cylinder in network M with load of 0.3 during simulation

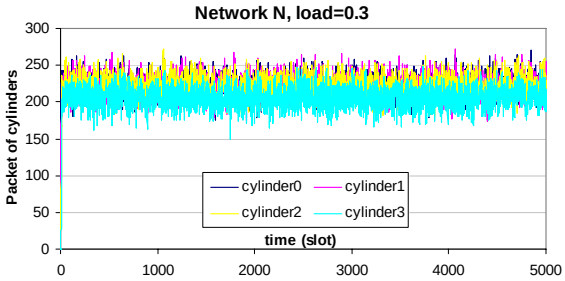



**Fig.16** Packet count of each cylinder in network N with load of 0.3 during simulation

We specifically compared the traffic or packet count at each cylinder level in both lightly loaded and heavily loaded conditions. In Fig. 15 and Fig. 16, the packet counts over simulation time (5000 time slots) under traffic load of 0.3 in network M and network N are shown for comparison. To get better view, the ranges of the packet / traffic load (once the traffic reaches a relatively steady state after the initial packet injection) at each cylinder are also shown in Fig. 17 and Fig. 18 respectively. We found that under this lightly loaded condition, both networks distribute their traffic among different levels pretty evenly, and there is only slight difference between outer cylinders and inner cylinders, which indicate no significant traffic back pressure buildups in both network M and network N. In our study, since no angular resolution is required, the last cylinder's packet count is not shown due to immediate exit at the stage. The absolute level of packet count in two networks may not provide a direct comparison due to different number of cylinders, but the pattern and difference between different cylinder levels are compared and focus of the study.

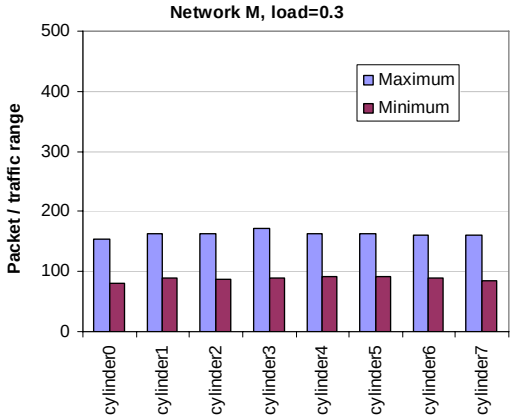



**Fig.17** Traffic range at each cylinder in network M with load of 0.3 during network operation

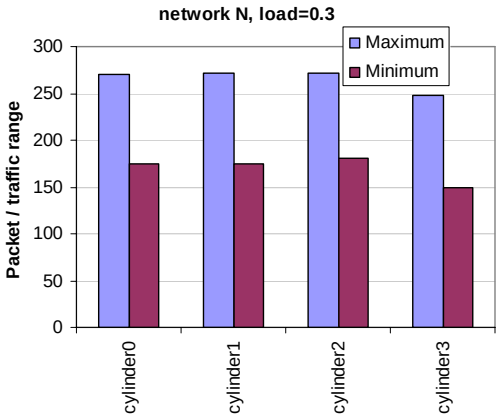



**Fig.18** Traffic range at each cylinder in network N with load of 0.3 during network operation

The same networks M and N under a heavier load of 0.8 are also studied for the comparison purpose. The results of the traffic distribution among different cylinder levels are shown in Fig. 19, Fig. 20, Fig. 21 and Fig. 22 respectively. We observed that in network M the outer cylinders bear very similar level of traffic loads, and only the last few inner cylinders carry slightly less loads (more visible compared with lightly loaded condition). On the other hand, in network N, while the total number of cylinders is much less, and outermost cylinder carries significantly more traffic compared with that of the inner cylinders. The difference in each cylinder is much more visible compared to that in network M, and such difference is also more significant in this heavier load condition than that in the lightly loaded network. There are a couple factors contributed to the difference. First of all, network N has about double the angle (for the same hardware cost), which amplify the traffic load difference

for each cylinder by a factor of 2 given the same height *H* in both networks. The second factor is due to more accumulated traffic backpressure at the outer cylinders in network N than that in network M. Because of longer deflection penalty and generally longer routing steps at each cylinder in *4*-ary networks, packets are staying in the cylinders for statistically longer period of time. So in comparison, the traffic backpressure is less in binary Data Vortex network, which creates much evenly distributed traffic among the different cylinder levels. The most inner cylinders always carry slightly lower loads compared to their outer cylinders because its input traffic is rather balanced or smoothed after the outer cylinder's routings. The traffic distribution among cylinders explains the overall more effective routing from the binary Data Vortex network, and thus it explains its higher throughput performance compared to its *4*-ary counterpart with the same I/O port and same hardware cost.

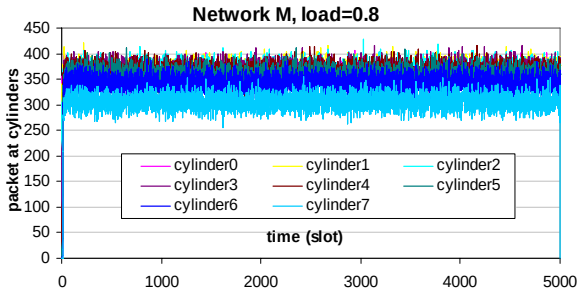



**Fig. 19** Packet count of each cylinder in network M with load of 0.8 during the simulation

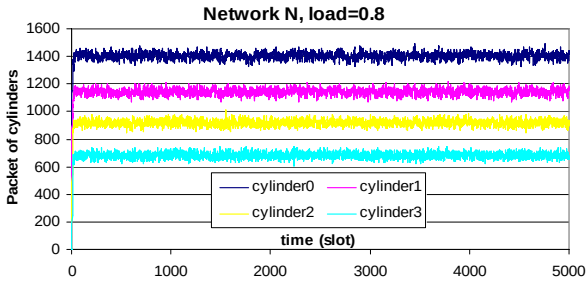



**Fig.20** Packet count of each cylinder in network N with load of 0.8 during the simulation

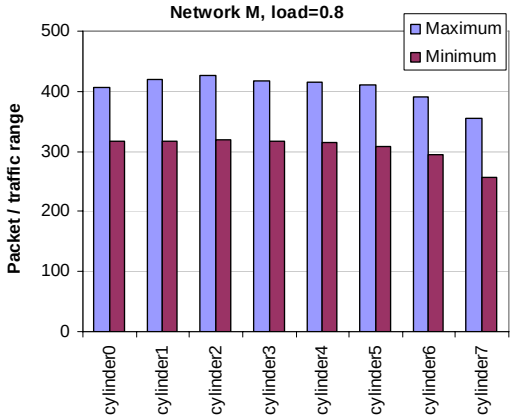



**Fig.21** Traffic range at each cylinder in network M during network operation

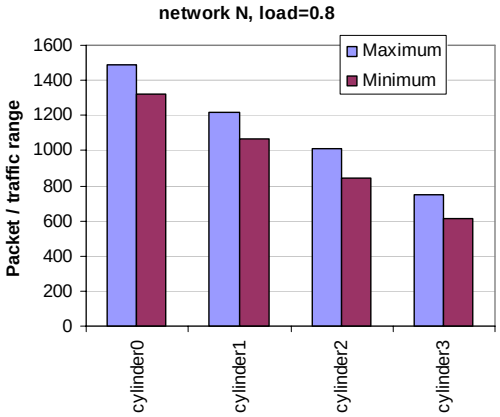



**Fig.22** Traffic range at each cylinder in network N during network operation

## 5. Conclusion

We have explored the potential of general *k*-ary decoding scheme in the Data Vortex networks. The comparison study has focused on studying the network behavior difference between the *4*-ary decoding network and the regular binary-tree decoding Data Vortex network. The results have concluded that overall routing performance is optimum with the original binary Data Vortex networks even though its forwarding latency is much longer than that in a *4*-ary network of a similar network cost. The main reason is that binary decoding provides the lowest deflection latency penalty, which in turn reduces the traffic backpressure during bufferless deflection based operation. Therefore, without any additional buffering at node, the binary Data Vortex networks allow for the system to push the most

packet traffic through at the lowest average latency. Future work may explore the effect of buffering capability within the nodes, however additional hardware cost must be included in the consideration.

## 6. Reference:

[1] Qimin Yang, "Optimum routing and forwarding path arrangement for bufferless Data Vortex Networks", *The Seventh International Conference on Networking (ICN 2008)*, Mexico, April 13-18, 2008.

[2] H. Jonathan Chao and Ben Liu, "High Performance Switches and Routers", *Wiley-Interscience*, ISBN 0470053674, 2007.

[3] Gorgios I. Papadimitriou, Chrisoula Papazoglou, Andreas S. Pomportsis, "Optical Switching: Switch Fabrics, Techniques and Architectures", *Journal of Lightwave Technology*, Vol.21, No. 2, pp.384-405, Feb 2003.

[4] Haijun Yang and S.J. Ben Yoo, "All-Optical Variable Buffering Strategies and Switch Fabric Architecture for Future All-Optical Data Routers", *Journal of Lightwave Technology*, Vol. 23, No.10, pp.3321-3330, Oct 2005.

[5] Chowdhury, A. Yong-Kee Yeo, Jianjun Yu, Gee-Kung Chang, "DWDM reconfigurable opticaldelay buffer for optical packet switched networks", *IEEE Photonics Technology letters*, Vol. 18, No.10, pp.1176-1178, May 2006.

[6] B.E.Swekla and R.I.Macdonald, "Tandem Banyan Switching Fabric with Dilation", *Electronics Letters*, Vol.27, No.19, pp.1770-1772, 1991.

[7] C. Hawkins, B.A. Small, D.S. Wills, K. Bergman, "The Data Vortex, an All Optical Path Multicomputer Interconnection Network", *IEEE Transactions on Parallel and Distributed Systems*, Vol. 18, No.3, pp. 409-420, March 2007.

[8] Assaf Shacham and Keren Bergman, "On contention resolution in the data vortex optical interconnection networks", *Journal of Optical Networking*, Vol.6, pp.777-788, 2007.

[9] A. Shacham, B.A.Small, O.Liboiron-Ladouceur, K. Bergman, "A fully implemented 12x12 data vortex optical packet switching interconnection networks", *Journal of Lightwave Technology*, vol.23, pp.3066-3075, 2005.

[10] O. Liboiron-Ladouceur, B.A. Small, K. Bergman, "Physical Layer Scalability of WDM Optical Packet Interconnection Networks," *Journal of Lightwave Technology*, Vol. 24, No.1, pp. 262-270, Jan 2006.

[11] Qimin Yang and Keren Bergman, "Traffic Control and WDM Routing in the Data Vortex Packet Switch", IEEE Photonics Technology Letters, Vol. 14, No.2, pp. 236-238, Feb 2002.

[12] Cory Hawkins and D.Scott Wills, "Impact of Number of Angles on the Performance of the Data Vortex Optical Interconnection Networks", *Journal of Lightwave Technology*, Vol.24, pp.3288-3294, 2006.

# Implementing and Testing a Formal Framework for Constraint-Based Routing over Scale-free Networks*

Stefano Bistarelli
Dipartimento di Matematica e Informatica
Università di Perugia,
Via Vanvitelli 1, Perugia, Italy
bista@dipmat.unipg.it

Francesco Santini
IMT Istituto di Studi Avanzati
Piazza San Ponziano 6, Lucca, Italy
f.santini@imtlucca.it

## Abstract

*We propose a formal model to represent and solve the* Constraint-Based Routing *problem in networks. To attain this, we model the network adapting it to a weighted* or *graph (unicast delivery) or* and-or *graph (multicast delivery), where the weight on a connector corresponds to the cost of sending a packet on the network link modelled by that connector. We use the Soft Constraint Logic Programming (SCLP) framework as a convenient declarative programming environment in which to solve the routing problem. In particular, we show how the semantics of an SCLP program computes the best route in the corresponding graph. The costs on the connectors can be described also as vectors (multidimensional costs), with each component representing a different* Quality of Service *metric value. At last, we provide an implementation of the framework over scale-free networks with the ECLiPSe programming environment, and we present the obtained results.*

**Keywords:** Constraint-Based Routing, Quality of Service, Scale-free Networks, Soft Constraint Logic Programming.

## 1 Introduction

Towards the second half of the nineties, Internet Engineering Task Force (IETF) and the research community have proposed many models and mechanisms to meet the demand for network *Quality of Service* (QoS). The classical routing problem has consequently been extended to include and to guarantee the QoS [35]: *QoS routing* [35, 15] denotes a class of routing algorithms that base path selection decisions on a set of QoS requirements or constraints, in addition to the destination. As defined in [15], QoS is a set of service requirements to be met by the network while transporting a flow. Service requirements have to be expressed in some measurable metric, such as bandwidth, number of hops, delay, jitter, cost and loss probability of packets.

In this paper we propose a formal framework based on *Soft Constraint Logic Programming* (SCLP) [4, 6] in which it is possible to represent and solve QoS-Routing [9] (and CBR in general). First, we will describe how to represent a network configuration in a corresponding *or* graph (for the unicast delivery scheme) or *and-or* graph (for multicast), mapping network nodes to graph nodes and links to graph *connectors*. In the following, we will generally use the term *and-or* graph, or simply graph. QoS link costs will be translated into multidimensional costs for the associated connectors. Afterwards, we will propose the SCLP framework [4, 6] as a convenient declarative programming environment in which to specify and solve such problem. SCLP programs are an extension of usual Constraint Logic Programming (CLP) programs where logic programming is used in conjunction with soft constraints, that is, constraints which can be satisfied at a certain level. In particular, we will show how to represent an *and-or* graph as an SCLP program, and how the semantics of such a program computes the best route the corresponding weighted *and-or* graph (with route we will consider both multicast tree and unicast paths). SCLP is based on the general structure of *c-semiring* (or simply semiring), having the two operations $\times$ and +: the $\times$ is used to combine the costs, while the partial order defined by $+$ operation (see Section 3), is used to compare the costs. Notice that the cartesian product of two semirings is a semiring [7], and this can be fruitfully used to describe multi-criteria problems. In Section 6, we will suggest an implementation of the proposed framework to really test the performance on scale-free networks generated ad-hoc. In scale-free networks some nodes act as "highly

*Partially supported by Istituto di Informatica e Telematica (IIT-CNR) Pisa, and Dipartimento di Scienze, Università "G. d'Annunzio", Pescara, Italy.

connected hubs" (i.e., high degree), although most nodes are of low degree. Moreover, these networks maintain their clustered structure during their growth. Scale-free networks represent the state-of-the-art topology (since replaced random networks) and can help reducing the complexity of our framework. This paper extends the work presented in [1] with a new implementation and new test results (see Section 6.2).

### 1.1 Structure of the Paper

A more theoretical version of this work is represented by [10]. The paper is organized as follows: in Section 2 we present some general background information about routing and scale-free networks. Section 3 features the SCLP framework, while Section 4 depicts how to represent a network environment with an *and-or* graph. In Section 5 we describe the way to pass from *and-or* graphs to SCLP programs, showing that the semantic of SCLP program is able to compute the best route in the corresponding *and-or* graph. Then, in Section 6, which represents the new content w.r.t. [1], we propose a practical implementation of the framework with a description on how to improve the performance. Lastly, Section 7 present the related work and and Section 8 draws the final conclusions.

## 2 Constraint-Based Routing and Scale-free Networks

**Constraint-Based Routing.** *Constraint-Based Routing* [35] (CBR) refers to a class of routing algorithms that base path selection decisions on a set of requirements or constraints, in addition to destination criteria. These constraints may be imposed by QoS needs (i.e., QoS-Routing) or administrative policies (i.e., Policy-Routing). The aim of CBR is to reduce the manual configuration and intervention required for attaining traffic engineering objectives [30]; for this reason, CBR enhances the classical routing paradigm with special properties, such as being resource reservation-aware and demand-driven.

Policy-Routing selects paths that conform to administrative rules and *Service Level Agreements* (SLAs) stipulated among service providers and clients. For example, routing decisions can be based on the applications or protocols used, size of packets or identity of the communicating entities. Policy constraints can help improving the global security of the network and also help the resource allocation problem that includes business decisions. QoS routing instead attempts to simultaneously satisfy multiple QoS requirements requested by real-time applications: e.g., video conference, distributed simulation, stock quotes or multimedia entertainment.

Multiple metrics can certainly represent the requests more accurately than using a single measure. However, it is well known that the problem of finding a route subject to multiple constraints is inherently hard [35]. When some metrics take real or unbounded integer values [12], satisfying two boolean constraints (saying whether or not a route is feasible), or a boolean constraint and a quantitative constraint (i.e., optimizing a metric) is NP-complete [34, 35, 12]. For example the set of constraints $C = (delay \leq 40msec, \min(Cost))$ is intractable. For this reason, most of the implemented algorithms in this area apply heuristics to reduce the complexity. The unicast problem can be reconducted to the generic *Multi-Constrained Optimal Path* problem [12], while the multicast case refers to the *Constrained Steiner Tree* [35]; both these problems are NP-complete in their nature.

Regarding unicast QoS Routing, in [21] the authors propose another heuristic approach for the multi-constrained optimal path problem (defined a *H_MCOP*), which optimizes a non-linear function (for feasibility) and a primary function (for optimality). The approach proposed in [23] exploits the dependencies among resources, e.g., available bandwidth, delay, and buffer space, to simplify the problem; then, a modified version Bellman-Ford algorithm can be used. Multicast QoS routing is generally more complex than unicast QoS routing, and for this reason less proposals have been elaborated in this area [35]: in MOSPF [26] the authors extend the classical (unicast) OSPF algorithm in order to optimize the delay, while the *Delay Variation Multicast Algorithm* (DVMA) [31] computes a multicast tree with both bounded delay and bounded jitter. Also, delay-bounded and cost-optimized multicast routing can be formulated as a Steiner tree: an example approach is *QoS-aware Multicast Routing Protocol* [13] (QMRP).

**Scale-free Networks.** In Section 6.2 we present some results obtained by testing our framework with generated scale-free networks, since several works as [19, 33] show that Internet topology can be modeled with such model. Small-world networks may belong to three classes: single-scale, broad-scale, or scale-free depending on their connectivity distribution $P(k)$, which is the probability that a randomly selected node has exactly $k$ edges. Scale-free networks follow a power law of the generic form $P(k) \backsim k^{-\gamma}$ [19]: in words, in these networks some nodes act as "highly connected hubs" (with a high degree), although most nodes are of low degree. Intuitively, the nodes that already have many links are more likely to acquire even more links when new nodes join in the graph: this is the so-called "rich gets richer" phenomenon. These hubs are the responsible for the small world phenomenon. The consequences of this behavior are that, compared to a random graph with the same size and the same average degree, the average path

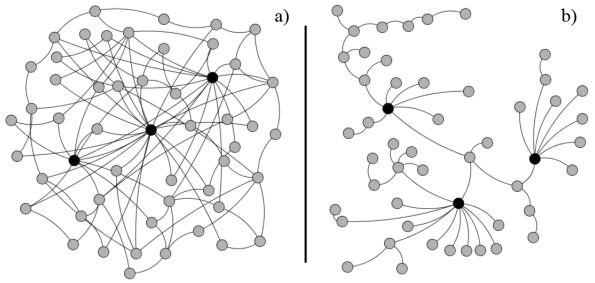



**Figure 1.** *a)* **a network with a clustering coefficient of** $0.1$**, and** *b)* **with a clustering coefficient of** $0.62$**.**

length of the scale-free model is somewhat smaller, and the clustering coefficient of the network is higher, suggesting that the graph is partitioned in sub-communities. As an example, see in Figure 1 the difference between a scale-free network with a very high clustering coefficient (i.e., Figure 1b) and a network with a lower one (Figure 1a). Black nodes show the big hubs in both networks, and it is graphically visible how Figure 1b is more partitioned in sub-communities.

Several works as [19, 33] show that Internet topology can be modeled with scale-free graphs: in [33], the authors distinguish between the *Autonomous System* (AS) level, where each AS refers to one single administrative domain of the Internet, and the *Internet Router* level (IR). At the IR level, we have graphs with nodes representing the routers and links representing the physical connections among them; at the AS level graphs each node represents an AS and each link represents a peer connection trough the use of the *Border Gateway Protocol* (BGP) protocol. Each AS groups a generally large number of routers, and therefore the AS maps are in some sense a coarse-grained view of the IR maps. The scale-free property of both these kinds of graphs is confirmed in [33] with a $\gamma = 2.1 \pm 0.1$, even if IR graphs have a power-law behavior smoothed by an exponential cut-off: for large $k$ the connectivity distribution follows a faster decay, i.e., we have much less nodes with a high degree. This truncation is probably due to the limited number of physical router interfaces. In [14] the authors prove that scale free networks with $2 < \gamma < 3$ have a very small diameter, i.e., $\ln \ln N$, where $N$ is the number of nodes in the graph.

## 3 Soft Constraint Logic Programming

The SCLP framework [4, 6], is based on the notion of *c-semiring* introduced in [7]. A c-semiring $S$ is a tuple $\langle A, +, \times, \mathbf{0}, \mathbf{1} \rangle$ where $A$ is a set with two special elements ($\mathbf{0}, \mathbf{1} \in A$) and with two operations $+$ and $\times$ that satisfy certain properties: $+$ is defined over (possibly infinite) sets of elements of $A$ and thus is commutative, associative, idempotent, it is closed and $\mathbf{0}$ is its unit element and $\mathbf{1}$ is its absorbing element; $\times$ is closed, associative, commutative, distributes over $+$, $\mathbf{1}$ is its unit element, and $\mathbf{0}$ is its absorbing element (for the exhaustive definition, please refer to [7]). The $+$ operation defines a partial order $\leq_S$ over $A$ such that $a \leq_S b$ iff $a + b = b$; we say that $a \leq_S b$ if $b$ represents a value *better* than $a$. Other properties related to the two operations are that $+$ and $\times$ are monotone on $\leq_S$, $\mathbf{0}$ is its minimum and $\mathbf{1}$ its maximum, $\langle A, \leq_S \rangle$ is a complete lattice and $+$ is its lub. Finally, if $\times$ is idempotent, then $+$ distributes over $\times$, $\langle A, \leq_S \rangle$ is a complete distributive lattice and $\times$ its glb.

**Table 1. A simple SCLP program.**

```
s(X)   :- p(X,Y).     q(a)   :- t(a).
p(a,b) :- q(a).       t(a)   :- 2.
p(a,c) :- r(a).       r(a)   :- 3.
```

Semiring-based constraint satisfaction problems (SCSPs) are constraint problems where each variable instantiation is associated to an element of a c-semiring $A$ (to be interpreted as a cost, level of preference or, in this case, as a trust/reputation level), and constraints are combined via the $\times$ operation and compared via the $\leq_S$ ordering. Varying the set $A$ and the meaning of the $+$ and $\times$ operations, we can represent many different kinds of problems, having features like fuzziness, probability, and optimization. In Section 4, the set $A$ is used to collect the values of a QoS metric, the $\times$ operator to combine them into a result for a complete end-to-end route, and $+$ to find the best route w.r.t. the chosen QoS metric.

A simple example of a SCLP program over the semiring $\langle N, min, +, +\infty, 0 \rangle$, where $N$ is the set of non-negative integers and $D = \{a, b, c\}$, is represented in Table 1. The intuitive meaning of a semiring value like 3 associated to the atom $r(a)$ (in Table 1) is that $r(a)$ costs 3 units. Thus the set $N$ contains all possible costs, and the choice of the two operations $min$ and $+$ implies that we intend to minimize the sum of the costs. This gives us the possibility to select the atom instantiation which gives the minimum cost overall. Given a goal like $s(x)$ to this program, the operational semantics collects both a substitution for $x$ (in this case, $x = a$) and also a semiring value (in this case, 2) which represents the minimum cost among the costs for all derivations for $s(x)$. To find one of these solutions, it starts from the goal and uses the clauses as usual in logic programming, except that at each step two items are accumulated and combined with the current state: a substitution and a semiring value (both provided by the used clause). The combination of these two items with what is contained in the current goal

is done via the usual combination of substitutions (for the substitution part) and via the multiplicative operation of the semiring (for the semiring value part), which in this example is the arithmetic $+$. Thus, in the example of goal $s(X)$, we get two possible solutions, both with substitution $X = a$ but with two different semiring values: 2 and 3. Then, the combination of such two solutions via the $min$ operation give us the semiring value 2.

## 4 Using *and-or* Graphs to Represent Networks with QoS Requirements

An *and-or* graph [25] is defined essentially as a hypergraph. Namely, instead of arcs connecting pairs of nodes there are hyperarcs connecting an $n$-tuple of nodes ($n = 1, 2, 3, \ldots$). The arcs are called *connectors* and they must be considered as directed from their first node to all others. Formally an *and-or* graph is a pair $G = (N, C)$, where $N$ is a set of *nodes* and $C$ is a set of connectors $C \subseteq N \times \bigcup_{i=0}^{k} N^i$. Note that the definition allows 0-connectors, i.e., connectors with one input and no output node. In the following of the explanation we will also use the concept of *and* tree [25]: given an *and-or* graph $G$, an *and* tree $H$ is a *solution tree of $G$ with start node $n_r$*, if there is a function $g$ mapping nodes of $H$ into nodes of $G$ such that: *i)* the root of $H$ is mapped in $n_r$, and *ii)* if $(n_{i_0}, n_{i_1}, \ldots, n_{i_k})$ is a connector of $H$, then $(g(n_{i_0}), g(n_{i_1}), \ldots, g(n_{i_k}))$ is a connector of $G$.

Informally, a solution tree of an *and-or* graph is analogous to a path of an ordinary graph: it can be obtained by selecting exactly one outgoing connector for each node, and we use the resulting tree to model the multicast delivery. The unicast case is even simpler: we use an *or* graph (i.e., a classical graph) to represent the network and selecting one connector for each node clearly results in a path (not a tree).

In Figure 2 we directly represent a very simple network as a weighted *and-or* graph. Each of the nodes can be easily cast in a corresponding node of the *and-or* graph. In Figure 2, different icons feature the different role of the node in the network: the source of packets $n_0$, the routers $n_1$, $n_2$ and $n_3$, a subnetwork $n_5$ or plain receiver host $n_4$). To model the networks links between two nodes we use 1-connectors: $(n_0, n_1)$, $(n_1, n_2)$, $(n_1, n_3)$, $(n_2, n_4)$, $(n_3, n_4)$ and $(n_3, n_5)$. We remind that the connectors are directed, and thus, for example the connector $(n_0, n_1)$ means that $n_0$ can send packets to $n_1$. Moreover, since we are possibly interested in a multicast communication, we need to represent the event of sending the same packet to multiple destinations at the same time. To attain this, in Figure 2 we can see the two 2-connectors $(n_1, n_2, n_3)$ and $(n_3, n_4, n_5)$: we draw these $n$-connectors (with $n > 1$) as curved oriented arcs where the set of their output nodes corresponds to the destination nodes of the 1-connectors traversed by the

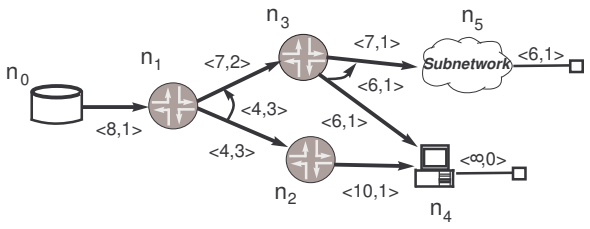



**Figure 2. A network in *and-or* graph representation.**

curved arc. Considering the ordering of the nodes in the tuple describing the connector, the input node is at the first position and the output nodes (when more than one) follow the orientation of the related arc in the graph (in Figure 2 this orientation is lexicographic). Notice that in the example we decided to use connectors with dimension at most equal to 2 (i.e., 2-connectors) for sake of simplicity. However it is possible to represent whatever cardinality (e.g., $n$) of multicast destination nodes (i.e., with a $n$-connector). 0-connectors are represented as a line ending with a square in Figure 2 and are added only for receiver nodes.

In the example we propose here, we are interested in QoS link-state information concerning only the bandwidth and a generic money cost (e.g., to supply the service or to maintain a device). Bandwidth and cost can be seen as either QoS or policy constraints. Therefore, each link cost of the network can be labeled with a 2-dimensional cost for the related connector. For example, the pair $\langle 8, 1\rangle$ for the connector $(n_0, n_1)$ tells us that the maximum bandwidth on that represented link is 80Mbps and a cost of 10€. In general, we could have a cost expressed with a $v$-dimensional vector, where $v$ is the number of metrics to be taken in account while computing the best distribution tree. In the case when a connector represent a multicast delivery (i.e., a $n$-connector with $n > 1$), its cost is decided by assembling the costs of all the $n$ links with the composition operation $\circ$, which takes as many $v$-dimensional cost vectors as operands, as the $n$ number of links represented by the connector. For this example, the result of $\circ$ is the minimum bandwidth and the highest cost, ergo, the worst QoS metric values among the considered links:

$$\circ(\langle b_1, c_1\rangle, \langle b_2, c_2\rangle, \ldots, \langle b_n, c_n\rangle) \longrightarrow$$

$$\langle \min(b_1, b_2, \ldots, b_n), \max(c_1, c_2, \ldots, c_n)\rangle$$

For example, the cost of the connector $(n_1, n_2, n_3)$ in Figure 2 is $\langle 4, 3\rangle$, since the costs of connectors $(n_1, n_2)$ and $(n_1, n_3)$ are respectively $\langle 4, 3\rangle$ and $\langle 7, 2\rangle$: $\circ(\langle 4, 3\rangle, \langle 7, 2\rangle) = \langle 4, 3\rangle$. All the costs of the connectors are reported in Table 2.

Then, we need some algebraic framework to model our preferences for the links in order to find the best route; to attain this, we use the semiring structure as described in Section 3. Since we are interested in maximizing the bandwidth of the distribution tree, we can use the c-semiring $S_{Bandwidth} = \langle \mathcal{R}^+, \max, \min, 0, +\infty \rangle$ (otherwise, we could be interested in finding the route with the minimal feasible bandwidth with $\langle \mathcal{R}^+, \min, \min, +\infty, 0 \rangle$, for traffic engineering reasons). We can use $S_{Cost} = \langle \mathcal{R}^+, \min, +, +\infty, 0 \rangle$ as the semiring to represent the cost, if we need to minimize it (here, $+$ is the arithmetic operator). Since the composition of c-semirings is still a c-semiring [7], $S_{Network} = \langle \langle \mathcal{R}^+, \mathcal{R}^+ \rangle, +', \times', \langle 0, +\infty \rangle, \langle +\infty, 0 \rangle \rangle$ is the adopted semiring, where $+'$ and $\times'$ correspond to the vectorization of the $+$ and $\times$ operations in the two c-semirings: $\langle b_1, c_1 \rangle +' \langle b_2, c_2 \rangle = \langle \max(b_1, b_2), \min(c_1, c_2) \rangle$ and $\langle b_1, c_1 \rangle \times' \langle b_2, c_2 \rangle = \langle \min(b_1, b_2), c_1 + c_2 \rangle$.

Clearly, the problem of finding best route is multi-criteria, since both bandwidth and delay must be optimized. We consider the criteria as independent among them, otherwise they can be rephrased to a single criteria [34]. Thus, the multidimensional costs of the connectors are not elements of a totally ordered set, and it may be possible to obtain several routes for the same destination (or destinations, if looking for a multicast distribution), all of which are not *dominated* by others, but which have different incomparable costs. The set of constraints for our problem is $C = (\max(Bandwidth), \min(Cost))$, which are both quantitative constraints: the semiring structure is suitable for metric optimization (i.e., to represent quantitative constraints), but in Section 5 we will apply also boolean constraints, e.g., only paths with $Cost < 22$€.

For each possible receiver node, the cost of its outgoing 0-connector will be always included in every route reaching it. As a remind, a 0-connector has only one input node but no destination nodes. If we consider a receiver as a plain node (e.g., $n_4$ in Figure 2), we can set this cost as the $\mathbf{1}$ element of the adopted c-semiring ($\mathbf{1}$ is the unit element for $\times$), since the cost to reach the node is already completely described by the other connectors in the route: practically, we associate the highest possible QoS values to this 0-connector, in this case infinite bandwidth and null cost. Otherwise we can imagine a receiver as a more complex subnetwork (as $n_5$ in Figure 2), and thus we can set the cost of the 0-connector as the cost needed to finally reach a node in that subnetwork (as the cost $\langle 6, 1 \rangle$ for the 0-connector after node $n_5$ in Figure 2), in case we do not want, or cannot, show the topology of the subnetwork, e.g., for security reasons.

**Table 2. The CIAO program representing all the routes over the weighted *and-or* graph problem in Figure 2.**

```
:- module(network,_,_).
:- use_module(library(lists)).

min([X, Y], X) :- X < Y.
min([X, Y], Y) :- X >= Y.
max([X, Y], X) :- X > Y.
max([X, Y], Y) :- X =< Y.

times([B1, C1], [B2, C2], [B, C]) :-
    min([B1, B2], B),
    C is (C1 + C2).

Leaves
leaf([n4], [1000, 0]).
leaf([n5], [6, 1]).

Edges
edge(n0, [n1], [8, 1]).
edge(n1, [n2], [4, 3]).
edge(n1, [n3], [7, 2]).
edge(n2, [n4], [10, 1]).
edge(n3, [n4], [6, 1]).
edge(n3, [n5], [7, 1]).

5) routeList([X|Xs], Z, [B, C]):-
     route(X, Z1, [B1, C1]),
     append(Z1, Z2, Z),
     routeList(Xs, Z2, [B2, C2]),
     times([B1, C1], [B2, C2], [B, C]).

1) connector(X, [Y], L, [B,C]):-
     nocontainsx(L, Y),
     edge(X, Y, [B,C]).

1) connector(X, [Y|Ys], L, [B,C]):-
     edge(X, Y, [B1,C2]),
     nocontainsx(L,Y),
     insert_last(L, Y, Z),
     connector(X, Ys, Z, [B2,C2]),
     min([B1,B2], B]),
     max([C1,C2], C).

2) route(X, [X], [B, C]):-
     leaf([X], [B, C]).

3) route(X, Z, [B, C]):-
     connector(X, W, [], [B1, C1]),
     routeList(W, Z, [B2, C2]),
     times([B1, C1], [B2, C2], [B, C]).

4) routeList([], [], [100, 0]).
```

## 5 *And-or* graphs using SCLP

In this Section, we represent the *and-or* graph in Figure 2 with a program in SCLP. programming environment and the semiring structure is a very parametric tool where to represent several and different cost models, with respect to QoS metrics. Using this framework, we can easily solve the Constraint-Based Routing problem by querying for either multicast trees or unicast paths.

To represent the network edges (i.e., 1-connectors), in SCLP we can write clauses like $edge(n_1, n_2) : -\langle 4, 3 \rangle$, stating that the graph has a connector from $n_1$ to nodes $n_2$ and $n_3$ with a bandwidth cost of 40Mbps and a money cost of 30€. Other SCLP clauses can properly describe the structure of the route we desire to search over the graph.

We chose to represent an *and-or* graph with a program in *CIAO Prolog* [11], a system that offers a complete Prolog system supporting ISO-Prolog and several extensions. CIAO Prolog has also a fuzzy extension, but since it does not completely conform to the semantic of SCLP defined in [6] (due to interpolation in the interval of the fuzzy set), we decided to use the CIAO operators among constraints (as $<$ and $\leq$), and to model the $\times$ operator of the c-semiring with them. For this reason, we added the cost of the connector in the head of the clauses, differently from SCLP clauses which have the cost in the body of the clause.

From the weighted *and-or* graph problem in Figure 2 we can build the corresponding CIAO program of Table 2 as follows. The set of network edges (or 1-connectors) is high-

lighted as *Edges* in Table 2. Each fact has the structure

$$edge(source\_node, [dest\_nodes], [bandwidth, cost])$$

e.g., the fact $edge(n_1, [n_2], [4, 3])$ represents the 1-connector of the graph $(n_1, n_2)$ with bandwidth equal to $40$Mbps and cost $30$€. The *Rules 1* in Table 2 are used to compose the edges (i.e., the 1-connectors) together in order to find all the possible $n$-connectors with $n \geq 1$, by aggregating the costs of 1-connectors with the $\circ$ composition operator, as described in Section 4 (the lowest of the bandwidths and the greatest of the costs of the composed 1-connectors). Therefore, with these clauses (in *Rules 1*) we can automatically generate the set of all the connectors outgoing from the considered node (in Table 2, *no-contains* and *insert_last* are CIAO predicates used to build a well-formed connector). The *Leaves* in Table 2 represent the 0-connectors (a value of $1000$ represents $\infty$ for bandwidth). The *time* rule in Table 2 mimics the $\times$ operation of the semiring proposed in Section 4: $S_{Network} = \langle\langle \mathcal{R}^+, \mathcal{R}^+\rangle, +', \times', \langle 0, +\infty\rangle, \langle +\infty, 0\rangle\rangle$, where $+'$ is equal to $\langle \max, \min\rangle$ and $\times'$ is equal to $\langle \min, +\rangle$, as defined in Section 4. At last, the rules 2-3-4-5 of Table 2 describe the structure of the routes we want to find over the graph. *Rule 2* represents a route made of only one leaf node, *Rule 3* outlines a route made of a connector plus a list of sub-routes with root nodes in the list of the destination nodes of the connector, *Rule 4* is the termination for *Rule 5*, and *Rule 4* is needed to manage the junction of the disjoint sub-routes with roots in the list $[X|Xs]$; clearly, when the list $[X|Xs]$ of destination nodes contains more than one node, it means we are looking for a multicast route. When we compose connectors or trees (*Rule 2* and *Rule 5*), we use the *times* rule to compose their costs together. In *Rule 5*, *append* is a CIAO predicate used to join together the lists of destination nodes, when the query asks for a multicast route.

To solve the CBR problem it is enough to perform a query in the Prolog language: for example, if we want to compute the cost of all the multicast trees rooted at $n_0$ and having as leaves the nodes representing the receivers (in this case, $n_4$ and $n_5$), we have to perform the query $route(n_0, [n_4, n_5], [B, C])$, where $B$ and $C$ variables will be instantiated with the bandwidth and cost of the found trees. For this query, the best output (in terms of the adopted QoS metrics) of the CIAO program corresponds to the cost of the tree in Figure 3*a*, i.e., $\langle 6, 5\rangle$, since $\times'$ computes the *minimum bandwidth - cost sum* of the connectors.

The best unicast path between $n_0$ and $n_4$ can instead be found with the query $route(n_0, [n_4], [B, C])$, and it is represented in Figure 3*b*; its cost is $\langle 6, 4\rangle$. Notice that the best path or tree is directly computed by the SCLP engine as described in the example in Section 3: given a query, the operational semantics collects a semiring value which represents the best cost (w.r.t. the $+$ operator) among the costs

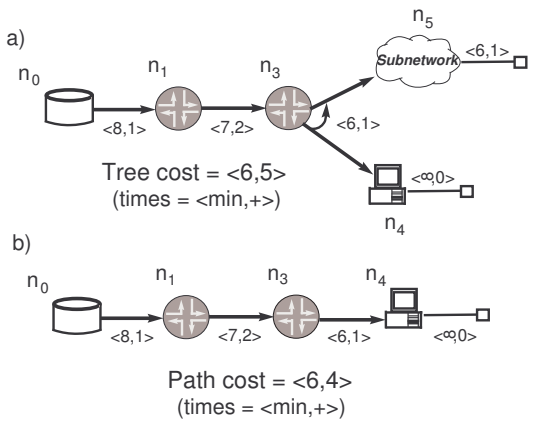



**Figure 3.** *a)* **The best multicast tree among** $n_0$ **and** $n_4$**-**$n_5$**, and** *b)* **the best unicast path between** $n_0$ **and** $n_4$**.**

of all the derivations satisfying the query. In Table 2, the SCLP engine is prototyped with a CIAO Prolog program.

As anticipated in Section 4, semiring structures are the ideal to represent quantitative constraints since the $+$ operation of the semiring defines a partial order over $A$ (see Section 3), i.e., over the set of QoS metric values. This operation can be consequently used to optimize the route. However, also boolean constraints, e.g., a route is accepted only if its cost is below a given threshold (e.g., $Cost < 30$€), can be modeled in our framework. For example, with the query $route(n_0, [n_4], [B, C]), C < 3$ no path is returned since the best possible path in Figure 3 has a money cost equal to $4$. The $C < 3$ requirement can be directly embedded in the *times* rule of the CIAO program Table 2, in order to also optimize the search by stopping it as soon as $C < 3$ is no longer true.

Other constraints that could be easily represented in our framework are those based on modalities [8], where each link has an associated information about the modality to be used to traverse it. For example, a list of protocols, ports or applications admitted on that link (e.g., RSVP, port $80$, VPN), or reserved time slots. shortest-paths have been studied in [8].

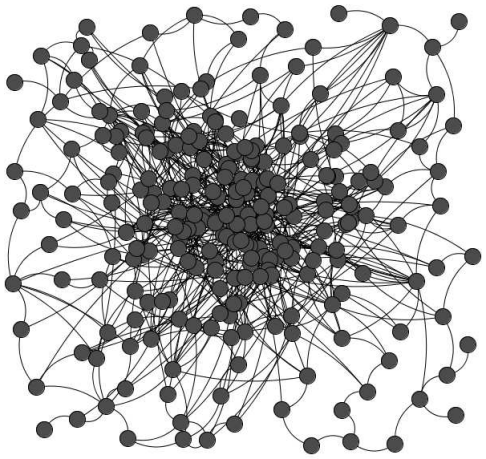
## 6 Implementing the Framework

To develop and test a practical implementation of our model, we adopt the *Java Universal Network/Graph Framework* (JUNG) [28], a software library for the modeling, analysis, and visualization of a graph or network. With this library it is also possible to generate scale-free networks according to the preferential attachment proposed in [3]: each time a new vertex $v_n$ is added to the network $G$, the proba-

bility $p$ of creating an edge between an existing vertex $v$ and $v_n$ is $p = (degree(v) + 1)/(|E| + |V|)$, where $|E|$ and $|V|$ are respectively the current number of edges and vertices in $G$. Therefore, vertices with higher degree have a higher probability of being selected for attachment. We generated the scale-free network in Figure 4 (the edges are undirected) and then we automatically produced the corresponding program in CIAO (where the edges are directed), as shown in Section 5. This translation can be easily achieved by writing a text file (from the same Java program generating the network) with all the clauses representing the edges. The clauses that find the best paths/trees are instead always the same ones.

The statistics in Figure 4 suggest the scale-free nature of our network: a quite high clustering coefficient, a low average shortest path and a high variability of vertex degrees (between average and max). These features are evidences of the presence of few big hubs that can be used to shortly reach the destinations. To generate the network in Figure 4, we used the JUNG constructor *public BarabasiAlbertGenerator(int init_vertices, int numEdgesToAttach, boolean directed, boolean parallel, int seed)* with parameters respectively instantiated to $100, 3, false, false, 1$: $init\_vertices$ represents the number of unconnected "seed" vertices that the graph should start with, *numEdgesToAttach* is the number of edges that should be attached from the new vertex to pre-existing vertices at each time step; the following two instantiated parameters state that we want directed and not parallel edges in the graph, while the last parameter is a random number seed. Then, the *public void evolveGraph(int numTimeSteps)* Java method instructs the algorithm to evolve the graph *numTimeSteps* time steps (instantiated to 200) and returns the most current evolved state of the graph.

However, with the CIAO program representing the network in Figure 4, all the queries we tried to perform over that graph were explicitly stopped after 5 minutes without discovering the best QoS route solution. Therefore, a practical implementation definitely needs a strong performance improvement: in Section 6.1 and Section 6.2 we show some possible solutions that could all be used also together. In Section 6.1 we suggest that *tabling* techniques could help for such problem. In Section 6.2 we show an implementation of the exactly same program in ECLiPSe [2]: in addition, we use branch-and-bound to prune the search and we claim that only this technique is sufficient to experience a feasible response time for the queries.



| Nodes | Edges | Clustering | Avg. SP |
|---|---|---|---|
| 265 | 600 | 0.13 | 3.74 |
| Min Deg | Max Deg. | Avg. Deg | Diameter |
| 1 | 20 | 4.52 | 8 |

**Figure 4. The test scale-free network and the related statistics.**

### 6.1 Tabled Soft Constraint Logic Programming and Network Decomposition

In logic programming, the basic idea behind *tabling* (or *memoing*) is that the calls to tabled predicates are stored in a searchable structure together with their proven instances: subsequent identical calls can use the stored answers without repeating the computation.

Tabling improves the computability power of Prolog systems and for this reason many programming frameworks have been extended in this direction. Due to the power of this extension, many efforts have been made to include it also in CLP, thus leading to the *Tabled Constraint Logic Programming* (TCLP) framework. In [16] the authors present a TCLP framework for constraint solvers written using attributed variables; however, when programming with attributed variables, the user have to take care of of many implementation issues such as constraint store representation and scheduling strategies. A more recent work [32] explains how to port *Constraint Handling Rules* (CHR) to XSB (acronym of *eXtended Stony Brook*), and in particular its focus is on technical issues related to the integration of CHR with tabled resolution: as a result, a CHR library is presently combined with tabling techniques within the XSB system. CHR is a high-level natural formalism to specify constraint solvers and propagation algorithms. This represents a promising framework where to solve QoS

routing problems and improve the performance (for example, tabling efficiency is shown in [29]), since soft constraints have already been successfully ported to the CHR system [5]. Hence, part of the soft constraint solving can be performed once and reused many times.

### 6.2 An branch-and-bound implementation in ECLiPSe

As shown in Section 4, the representation of the outgoing edges of node in the multicast model can be composed by a total of $O(2^n)$ connectors, thus in the worst case it is exponential in the number of graph nodes. This drawback, which is vigorously perceived in strongly connected networks, and together with considering a real case network linking hundreds of nodes, would heavily impact on the time-response performance during a practical application of our model. Therefore, it is necessary to elaborate some improvements to reduce the complexity of the tree search, for example by visiting as few branches of the SCLP tree as possible (thus, restricting the solution space to be explored). For this reason, we provide a further implementation by using the ECLiPSe [2] system.

ECLiPSe is a software system for the development and deployment of constraint programming applications, e.g., in the areas of planning, scheduling, resource allocation, timetabling, transport and more. It contains several constraint solver libraries, a high-level modelling and control language, interfaces to third-party solvers, an integrated development environment and interfaces for embedding into host environments [2]. In particular, we exploit the *branch_and_bound* library in order to reduce the space of explored solutions and consequently improve the performance. Branch-and-bound is a well-known technique for optimization problems, which is used to immediately cut away not promising partial solutions, by basing on a "cost" function. Unfortunately, as far as we know, ECLiPSe does not support tabling techniques (introduced in Section 6.1) and therefore it cannot be adopted to compose the benefits of both techniques.

In Figure 5 we show a program in ECLiPSe that represents the unicast QoS routing problem for the scale-free network in Figure 4. We decided to show only the unicast case for sakes of clarity, but feasible time responses can be similarly obtained for the multicast case (i.e., searching for a tree instead of a plain path) by working on the branch-and-bound interval of explored costs, as we will better explain in the following. Clearly, in Figure 5 we report only some of the 600 edges of the network.

The code in Figure 5 has been automatically generated with a *Java* program using JUNG, as done for the CIAO program in Section 6: the corresponding text file is 30Kbyte. The size can be halved by not printing the reverse

```
:- lib(ic).
:- lib(branch_and_bound).
:- lib(lists).

edge(n0,[n192], [9, 2]).
edge(n1,[n119], [4, 2]).
edge(n2,[n183], [5, 9]).
edge(n2,[n23], [7,7]).
edge(n2,[n260], [2, 1]).
edge(n2,[n115], [6, 9]).
edge(n2,[n156], [9, 4]).
edge(n2,[n4], [6, 5]).
        .
        .
        .
edge(n263,[n167], [2, 4]).
edge(n263,[n191], [6, 9]).
edge(n263,[n70], [5, 2]).
edge(n263,[n108], [6, 4]).
edge(n263,[n26], [5, 9]).
edge(n263,[n46], [8, 5]).
edge(n263,[n171], [6, 7]).
edge(n263,[n35], [6, 3]).
edge(n264,[n102], [6, 4]).
edge(n264,[n189], [3, 1]).
edge(n264,[n68], [8, 6]).
edge(n264,[n119], [5, 9]).
edge(n264,[n156], [5, 1]).
```

```
path(X, [Y], C, D, L, [Y]):-
    edge(X, [Y], [A, B]),
    C #= A + B,
    nonmember(Y, L),
    D is 1.

path(X, [Y], C, D, L, N):-
    C1 #>= 0, C2 #>=0,
    C1 #= A + B,
    C #= C1 + C2,
    D #= 1 + D2,
    edge(X, [Z], [A, B]),
    nonmember(Z, L),
    append(L, [Z], L2),
    path(Z, [Y], C2, D2, L2, N2),
    append(N2, [Z], N).

searchpath_bb(X, Y, C, D, L, N):-
    D #>= 1, D #=< 16,
    C #>= 0, C #=< 160,
    minimize(path(X, [Y], C, D, L, N2), C),
    append(N2, [X], N).

searchpath_all(X, Y, C, D , K, L, N):-
    findall(C, path(X, [Y], C, D, K, N2), L),
    append(N2, [X], N).
```

**Figure 5. The representation in ECLiPSe (with branch-and-bound optimization) of the QoS routing problem for the network in Figure 4; clearly, only some of the $600$ edges are shown.**

links and generating them with a specific clause, if each link and its reverse one have the same cost.

The branch-and-bound optimization is achieved with $minimize(+Goal, ?Cost)$ (importing the *branch_and_bound* library) in the $searchpath\_bb$ clause in Figure 5, where the *Goal* is a nondeterministic search routine (the clauses that describe the *path* structure) that instantiates a *Cost* variable (i.e., the QoS cost of the path) when a solution is found. Notice that for each of the edges of the network we randomly generated two different QoS costs by using the *java.util.Random Class*, each of them in the interval $[1..10]$. Therefore, the cost of a link is represented by a pair of values: the cost of the path is computed by summing the two QoS features together (i.e., *A* and *B* in Figure 5): we compute $w_1A + w_2B$ and we suppose $w_1 = w_2 = 1$, i.e., the composed cost of a link is in the interval $[2..20]$. The reason we compose the two costs together is that ECLiPSe natively allows to apply a branch-and-bound procedure focused only on a single cost variable (it can be extended to consider more costs).

The two clauses $searchpath\_bb$ and $searchpath\_all$ represent the queries that can be asked to the system: they respectively use and not use the branch-and-bound optimization, i.e., $searchpath\_all$ finds all the possible paths in order to find the best one. In order to describe the structure

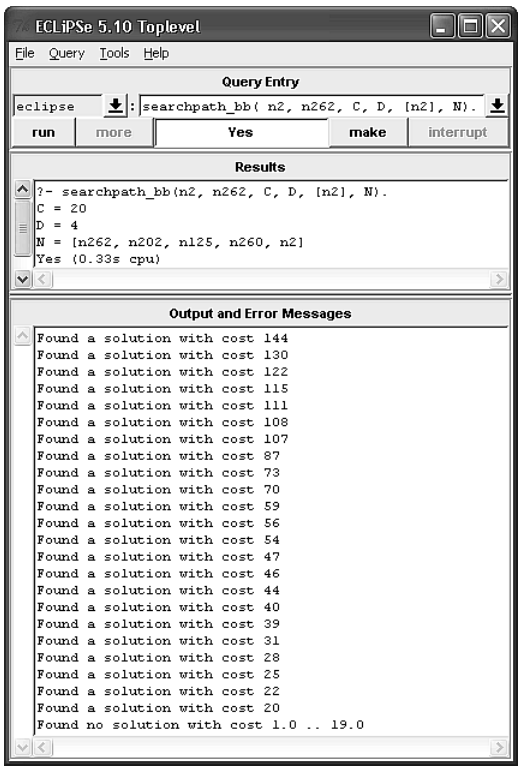



**Figure 6. The ECLiPSe shell with the query $searchpath\_bb(n6, n261, C, D, [n6], L)$ and the corresponding found result for the program in Figure 5.**

of a $searchpath\_bb$ query (see Figure 5), we take as example $searchpath\_bb(n2, n262, C, D, [n2], L)$: with this query we want to find the best path between the nodes $n2$ and $n262$, $C$ is the cost of the path (used also by the branch-and-bound pruning), $D$ is the number of hops, $L$ (in Figure 5) is the list of already traversed nodes and $N$ is a list used to collect the nodes of the path (in reverse order). The result of this query is reported in Figure 6, by showing directly the ECLiPSe window: the best cost value (i.e., 20) was found after 0.33 seconds with a path of 4 hops, i.e., $n2$-$n260$-$n125$-$n202$-$n262$.

The query $searchpath\_all(n6, n261, C, D, K, [n6], N)$ ($K$ is the list of solutions found by the *findall* predicate), which does not use the branch-and-bound pruning (and constraints), was explicitly interrupted after 10 minutes without finding an answer for the goal. Other queries are satisfied in even less than one second, depending on the efficiency of the pruning efficiency for the specific case.

To better describe and accelerate the search we added also some constraints, which are explained in Table 3. In Figure 5 we also import the hybrid integer/real interval arithmetic constraint solver of ECLiPSe to use them, i.e.,

| | |
|---|---|
| D #>= 1,<br>D #=< 16 | These two constraints are used to limit the depth (i.e. the number of hops) of the path we want to find. For the example in Fig. 15 it was computed as *Diameter x 2 = 8 x 2 = 16*. It is a good overestimation since we are dealing with a scale-free network (see Sec 7.1). |
| D #= 1 + D2 | Used to compute the depth of the path. |
| C1 #>= 0,<br>C2 #>=0,<br>C1 #= A + B,<br>C #= C1 + C2 | Four constraints are used to compute the cost of the path: it is the cost of an edge (i.e. *C1* is obtained by summing the two *QoS* features *A* and *B*) plus the cost of the remaining part of the path (i.e. *C2*). Clearly, both *C1* and *C2* must be greater than *0*. |
| C #>= 0,<br>C #=< 160 | Used to limit the space of cost values: its reduction sensibly improves the performance. It is possible to start the search with a small threshold and then raise it if no solution is found. For the example in Fig. 15 it was computed as the maximum possible cost of a path: *EdgeMaxCost x Diameter = 20 x 8 = 160*. |

**Table 3. The description of the constraints used in Figure 5.**

the *ic* library. Notice that the constraints depending on the *Diameter* of the network (i.e., 8, as shown in Figure 4) limit the search space and provides a mild approximation at the same time: in scale-free networks, the average distance between two nodes can be $\ln \ln N$, where $N$ is the number of nodes [14] (see also Section 2). This property of scale-free networks clearly helps in improving the performance of our model, and scale-free networks show better end-to-end performance in general [27]. Therefore, considering a max depth of the path as twice the diameter value (i.e., 16) still results in a large number of alternative routes, since, for the scale-free network in Figure 4, this value is 4-5 times the average shortest path of the network (i.e., 3.74 as shown in Figure 4). Notice that, after the execution of the program in Figure 5, if no solution is found or we want to try if the obtained solution is really the one with the best cost, it is possible to change the constraints in Table 3 by trying disjoint intervals (e.g., $C\,\#> 160$, $C\,\#<= 320$ or $D\,\#> 16$, $D\,\#<= 32$), and then executing the program one more time (since performance permit to do so). The bound values for these intervals can be directly obtained from the statistics acquired during the network generation (see Figure 4). Notice that without the constraints in Table 3, the branch-and-bound optimization alone cannot improve the performance below the 5 minutes threshold.

In order to show the scalability property of our framework, in Table 5 we summarize the performance results of $k$ queries executed on three distinct scale-free networks with a different number of $n$ nodes: $n = 50$ ($50 < 2^6$), $n = 265$ ($265 < 2^9$) i.e., the network in Figure 4 and $n = 877$ ($877 < 2^{10}$). The $k$ number of queries is respec-

| Nodes | Queries | Min Time | Avg. Time |
|---|---|---|---|
| 50 | 30 | $\sim$ 0s | 0.1s |
| 265 | 45 | 0.02s | 4.08s |
| 877 | 50 | 0.5s | 4.89s |
| Nodes | Avg. Cost | Avg. Depth | Max Depth |
| 50 | 17.54 | 3.04 | 7 |
| 265 | 29.8 | 5.46 | 11 |
| 877 | 37.72 | 6.72 | 14 |

**Table 4. Some performance statistics obtained with the ECLiPSe framework (with branch-and-bound), collected on three different size networks (i.e., $50$, $265$ and $877$ nodes). On each network we performed $50$ queries.**

tively scaled with the network size: $k = 30$ (i.e., $6 \times 5$), $k = 45$ (i.e., $9 \times 5$) and $k = 50$ (i.e., $10 \times 5$). These statistics are related to the *Min/Average Time* needed to obtain a path, its *Average Cost* and its *Max/Average Depth*. For each query, the source and destination nodes have been randomly generated. We can see that *Min Time* sensibly differs from the *Average Time*, and this is due to the poor efficiency of the branch-and-bound pruning in some cases. However, this technique performs very well in most of cases, as the low *Average Time* in Table 5 shows (even for $n = 877$). The performance results in Table 5 have been collected on a *Pentium M 1.7Ghz* and *1Gb* of memory.

Comparable performance results are achievable as well also for the multicast case, by enforcing the structure of the tree with other ad-hoc constraints: for example, by constraining the width of the searched tree to the number of the multicast receivers in the query, since it is useless to find wider trees. Moreover, the problem can be first over-constrained and then relaxed step-by-step if no solution is found. For example, we can start by searching a solution in the cost interval $[0..35]$ and then, if the best solution is not included in this interval, setting the interval to $[36..70]$ (and so on until the best solution is found). Notice that in this way we strongly speed-up the search while preserving all the information, due to the characteristics of the branch-and-bound technique. This behaviour can be easily reproduced in ECLiPSe, since the customizable options of $bb\_min(+Goal, ?Cost, ?Options)$ (i.e., another clause to express branch-and-bound) include the $[From..To]$ interval parameters.

Finally, the ECLiPSe system can be used to further improve the performance, since it is possible to change the parameters of branch-and-bound, e.g., by changing the strategy after finding a solution [2]: *continue* search with the newly found bound imposed on Cost, *restart* or perform a *dichotomic* after finding a solution, by splitting the remaining cost range and restart search to find a solution in the lower sub-range. If it fails, the procedure assumes the upper sub-range as the remaining cost range and splits again. Moreover, it is possible to add *Local Search* to the tree search, and to program specific heuristics [2].

| Nodes | Queries | Min Time | Avg. Time |
|---|---|---|---|
| 50 | 50 | 10.63s | 107s |

**Table 5. Performance reported for the multicast program in Figure 7.**

Just as a first example, in Figure 7 we provide an ECLiPSe implementation also for the multicast routing case. Figure 7 does not report the imported libraries (which are the same of Figure 5) and the facts representing the edges in the graph. This program represents a first step towards a fast solution for the problem: even with only branch-and-bound techniques the problem becomes solvable inside the framework (without, the computation takes too much time and needs to be interrupted), as the results obtained for the network with $50$ nodes: with $50$ queries (one sender to 3 receivers) we have obtained an average response time of $107$ seconds, with a minimum response time of $10.63$ (see Table 5). The *disJoint* clauses are used to prevent the search from visiting the same node twice.

## 7 Related Work

Concerning the related works, in [18] and [20] the authors adopt a hypergraph model in joint with semirings too, but the minimal path between two nodes (thus, not over an entire tree) is computed via a hypergraph rewriting system instead of SCLP. At the moment, all these frameworks are not comparable from the computational performance point of view, since they have not yet been implemented. Even the work in [24] presents some general algebraic operators in order to handle QoS in networks, but without any practical results. We compare our work only with other theoretical frameworks, since our study aims at representing general routing constraints in order to solve different problems: due to the complexity of QoS routing, state-of-the-art practical solutions (presented in Section 2) deal only with a subset of metrics and constraints. On the other hand, a more general framework can help to analyze the problem from a global point of view, not linked to specific algorithms. With *Declarative routing* [22], a routing protocol is implemented by writing a simple query in a declarative query language (like Datalog as in [22]), which is then executed in a distributed fashion at some or all of the nodes. It is based on the observation that recursive query languages are a *natural-fit* for expressing routing protocols. However, the authors of [22] did not go deep in modelling QoS fea-

```
disJoint(L1,L2,X):-
   member(A,L1),
   X\=A,
   member(A,L2),    !,   fail.

disJoint(L1,L2,X).

tree(X, [X], L, 0, [X]):-
   leaf([X], [_, _]).

tree(X, Z, L, Q, Nodes):-
   Q #= C1 + C2, C2 #>= 0,
   CL #>= 1, CL #=< 3,
   connector(X, W, [], C1),
   length(W, CL),
   disJoint(L, W, X),
   append(L, W, K),
   treeList(W, Z, K, C2, Nodes).

treeList([], [], L, 0, L).

treeList([X|Xs], Z, L, Q, Nodes):-
   C1 #>= 0, C2 #>= 0, Q #= C1 + C2,
   tree(X, Z1, L, C1, Nodes1),
   disJoint(L,Z1,X),
   append(L, Z1, K),
   treeList(Xs, Z2, K, C2, Nodes2),
   append(Z1, Z2, Z),
   append(Nodes1,  Nodes2, NNN),
   sort(NNN,Nodes).

route(X, Y, Q, Nodes):-
   Q #>= 0, Q #=< 50,
   minimize(tree(X, Y, [X], Q, Nodes), Q)

connector(X, [Y], L, Q):-
   edge(X, [Y], [B,C]),
   Q #= B + C,
   append(L, [Y], Z),
   sort(Z,Z),
   nonmember(Y, L).

connector(X, [Y|Ys], L, Q):-
   Q2 #>= 0, Q #= (B1 + C1 + Q2),
   sort(L,L),
   edge(X, [Y], [B1,C1]),
   nonmember(Y,L),
   append(L, [Y], Z),
   sort(Z,Z),
   connector(X, Ys, Z, Q2).
```

**Figure 7. The ECLiPSe program for the multicast routing.**

tures, and we think that c-semirings represent a very good method to include these metrics.

To go further, aside the elegant formalization due to the SCLP framework, we build a bridge to a real implementation of the model (Section 6) and several ideas to improve the experienced performance. The final SCLP tool can be used to quickly prototype and test different routing paths. As far as we know, other formal representations completely miss this practical implementation [24]. Therefore, our paper vertically covers the problem: from theoretical to practical aspects, without reaching the performance of existing routing algorithms implemented inside the routers, but thoroughly and expressively facing the problem. The drawback of being so expressive is clearly represented by resulting performance: however, our goal is to deal with the off-line study of a network (e.g., to plan the laying of new cables and routers) and the shown performance easily permit to do so; in this sense, to build a routing table in a proactive way corresponds to a faster answer provided to the final user [17]. In this case our expressivity can be used to easily optimize the sets of QoS metrics (and features) for which no algorithm has been provided yet, especially for the less-studied multicast case [35] (only delay and cost metrics are optimized).

## 8 Conclusion

We have described a method to represent and solve the CBR problem with the combination of *and-or* graph and the declarative SCLP environment: the best multicast or unicast route found on an *and-or* graph corresponds to the semantics of a SCLP program. The route satisfies multiple constraints regarding QoS requirements, e.g., minimizing the global bandwidth consumption, reducing the delay, or accepting only the routes that use $k$ hops at most. The semiring structure is a very parametric tool where to represent different QoS metrics. Since it is well-known that even a shortest path problem with two or more independent metrics is NP-complete (see Section 1), we have proposed a framework based on AI techniques (i.e., soft constraints). The convenience is to use a declarative framework where constraints on the routes can be easily represented. Moreover we have provided a practical implementation of the framework and a test on a scale-free network, whose results are quite promising. We have used the ECLiPSe programming environment in order to use the branch-and-bound library to improve the results. The framework can be used to prototype and test new constraints in small networks (i.e., 100-1000 nodes) or parts of wider graphs.

Concerning future works, we want to produce more tests, also with different scale-free/small-world topology generators. We plan to improve the computational results by adding to the program some clauses that describe the topology of the network. Moreover, we will study ad-hoc memoization techniques to reduce the complexity of big hubs.

## References

[1] S. Bistarelli and F. Santini. A formal and practical framework for constraint-based routing. In *Seventh International Conference on Networking (ICN)*, pages 162–167. IEEE Computer Society, 2008.

[2] K. R. Apt and M. Wallace. *Constraint Logic Programming using Eclipse*. Cambridge University Press, New York, NY, USA, 2007.

[3] A. L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286:509, 1999.

[4] S. Bistarelli. *Semirings for Soft Constraint Solving and Programming*, volume 2962 of *Lecture Notes in Computer Science*. Springer, 2004.

[5] S. Bistarelli, T. Frühwirth, and M. Marte. Soft constraint propagation and solving in chrs. In *SAC '02: Proceedings of the 2002 ACM symposium on Applied computing*, pages 1–5, New York, NY, USA, 2002. ACM Press.

[6] S. Bistarelli, U. Montanari, and F. Rossi. Semiring-based constraint logic programming. In *Proc. IJCAI97 (Morgan Kaufman)*, pages 352–357. Morgan Kaufman, 1997.

[7] S. Bistarelli, U. Montanari, and F. Rossi. Semiring-based constraint solving and optimization. *Journal of the ACM*, 44(2):201–236, 1997.

[8] S. Bistarelli, U. Montanari, and F. Rossi. Soft constraint logic programming and generalized shortest path problems. *Journal of Heuristics*, 8(1):25–41, 2002.

[9] S. Bistarelli, U. Montanari, F. Rossi, and F. Santini. Modelling multicast qos routing by using best-tree search in and-or graphs and soft constraint logic programming. *Electr. Notes Theor. Comput. Sci.*, 190(3):111–127, 2007.

[10] S. Bistarelli, U. Montanari, F. Rossi, and F. Santini. Unicast and multicast QoS routing with soft constraint logic programming. *CoRR*, abs/0704.1783, 2007.

[11] F. Bueno, D. Cabeza, M. Carro, M. Hermenegildo, P. López-García, and G. Puebla. The ciao prolog system: reference manual. Technical Report CLIP3/97.1, School of Computer Science, Technical University of Madrid (UPM), 1997.

[12] S. Chen and K. Nahrstedt. An overview of quality of service routing for next-generation high-speed networks: Problems and solutions. *IEEE Network*, 12(6):64–79, 1998.

[13] S. Chen, K. Nahrstedt, and Y. Shavitt. A QoS-aware multicast routing protocol. In *INFOCOM Joint Conference of the IEEE Computer and Communications Societies (3)*, pages 1594–1603. IEEE, 2000.

[14] R. Cohen and S. Havlin. Scale-free networks are ultrasmall. *Phys. Rev. Lett.*, 90(5):058701, Feb 2003.

[15] E. Crawley, R. Nair, B. Rajagopalan, and H. Sandick. RFC 2386: A framework for QoS-based routing in the Internet, August 1998. Informational.

[16] B. Cui and D. S. Warren. A system for tabled constraint logic programming. In *CL '00: Proceedings of the First International Conference on Computational Logic*, pages 478–492, London, UK, 2000. Springer-Verlag.

[17] S. R. Das, R. Castaneda, J. Yan, and R. Sengupta. Comparative performance evaluation of routing protocols for mobile, ad hoc networks. In *Mobile Networks and Applications*, pages 153–161, 1998.

[18] R. De Nicola, G. L. Ferrari, U. Montanari, R. Pugliese, and E. Tuosto. A formal basis for reasoning on programmable QoS. In *Verification: Theory and Practice*, volume 2772, pages 436–479. Springer, 2003.

[19] M. Faloutsos, P. Faloutsos, and C. Faloutsos. On power-law relationships of the internet topology. In *SIGCOMM '99*, pages 251–262. ACM Press, 1999.

[20] D. Hirsch and E. Tuosto. SHReQ: coordinating application level QoS. In *SEFM '05: Software Engineering and Formal Methods*, pages 425–434. IEEE Computer Society, 2005.

[21] T. Korkmaz and M. Krunz. Multi-constrained optimal path selection. In *INFOCOM*, pages 834–843, 2001.

[22] B. T. Loo, J. M. Hellerstein, I. Stoica, and R. Ramakrishnan. Declarative routing: extensible routing with declarative queries. In *SIGCOMM '05: Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 289–300, New York, NY, USA, 2005. ACM.

[23] Q. Ma and P. Steenkiste. Quality of service routing for traffic with performance guarantees, May 1997.

[24] Z. Mammeri. Towards a formal model for qos specification and handling in networks. In *IWQoS*, pages 148–152. IEEE, 2004.

[25] A. Martelli and U. Montanari. Optimizing decision trees through heuristically guided search. *Commun. ACM*, 21(12):1025–1039, 1978.

[26] J. Moy. OSPF Version 2. RFC 2328 (IETF Standard), Apr. 1998.

[27] H. Ohsaki, K. Yagi, and M. Imase. On the effect of scale-free structure of network topology on end-to-end performance. In *SAINT '07: Proceedings of the 2007 International Symposium on Applications and the Internet*, page 12, Washington, DC, USA, 2007. IEEE Computer Society.

[28] J. O'Madadhain, D. Fisher, S. White, and Y. Boey. The JUNG (Java Universal Network/Graph) framework. Technical report, UC Irvine, 2003.

[29] I. V. Ramakrishnan, P. Rao, K. F. Sagonas, T. Swift, and D. S. Warren. Efficient tabling mechanisms for logic programs. In *International Conference on Logic Programming*, pages 697–711. The MIT Press, 1995.

[30] E. Rosen, A. Viswanathan, and R. Callon. IETF-RFC3031: Multiprotocol Label Switching Architecture, 2001.

[31] G. N. Rouskas and I. Baldine. Multicast routing with end-to-end delay and delay variation constraints. *IEEE Journal of Selected Areas in Communications*, 15(3):346–356, 1997.

[32] T.Schrijvers and D. S. Warren. Constraint handling rules and tabled execution. In B. Demoen and V. Lifschitz, editors, *ICLP*, volume 3132 of *Lecture Notes in Computer Science*, pages 120–136. Springer, 2004.

[33] A. Vazquez, R. Pastor-Satorras, and A. Vespignani. Internet topology at the router and autonomous system level, 2002.

[34] Z. Wang and J. Crowcroft. Quality-of-service routing for supporting multimedia applications. *IEEE Journal on Selected Areas in Communications*, 14(7):1228–1234, 1996.

[35] O. Younis and S. Fahmy. Constraint-based routing in the internet: basic principles and recent research. *IEEE Communications Surveys and Tutorials*, 5(1):2–13, 2003.

# Adjustable Multi-Sector Cellular Base Station Antenna

Senglee Foo, Member, *IEEE*, Bill Vassilakis, Senior Member, *IEEE*

Powerwave Technologies, 1801 E. St. Andrew Place, Santa Ana, CA, 92705
Email: Senglee.Foo@pwav.com, Tel: 714-466-1437

***Abstract* — This article presents a dual beam array (DBA) for higher-order sectorization of a cellular site. This technique provides a means of cost-effective boosting network capacity without the use of additional frequency spectrum. The DBA comprises of a multi-column array and a beam forming circuit, which produces two overlapping beams with adjustable beam patterns. The proposed antenna is an adjustable cylindrical sector array, composed of three separate linear array columns. This structure allows for the formation of two overlapping beams with amplitude and phase excitations which can be implemented using a compact and low loss circuit. The adjustable offset displacement for the center column array allows for the refinement and adjustment of the pattern characteristics of the two overlapping beams. This method results in a beam split loss of less than 0.5dB in comparison to a single beam case. Performance parameters such as the beam cross-over loss and pattern discrimination between beams can also be adjusted in-situ for optimum operation.**

***Index Terms* — Beam forming network, multi-beam, base station antenna, smart antenna, multi-sector array.**

## I. INTRODUCTION

Higher order sectorization provides a means of increasing cellular network capacity and providing optimum coverage without the use of additional frequency spectrum. Partial use of higher order sectorization is particularly useful for cost-effective accommodation of service growth in a localized service area within a cellular coverage. In these localized 'hotspot' areas antennas with multiple beams of narrower beamwidth and higher directivity can be used to increase the overall capacity. For instance, one of the 65 degree sectors within a cell site that has a higher traffic may be served using two overlapping beams of 33 degrees. This method allows an increase in an overall network capacity by using multiple columns of linear array arranged on a cylindrical curvature. This arrangement results in an amplitude and phase excitation that can be implemented using a simple and low-loss beam-forming network. It produces two symmetrical beams with respect to the azimuth boresight. Radiation patterns of the two beams are overlapped in the manner that the coverage of the cellular sector can be optimized using the adjustable beam array.

Multibeam patterns of a multi-column array are typically formed using a combination of hybrids, couplers and phase shifters [2-4]. This general beam forming method often incurs an additional front-end loss as a result of circuit path losses and signal split between beams. Furthermore, a general beam forming method often requires multiple crossing between input feed lines, which can cause difficulties in the actual beam-forming circuit implementation. This article is an extension of the previous reports [1][5], which presents the concept and test results of a proposed dual beam array (DBA) and the associated beam forming structure for use in efficient beam forming of two overlapping beams. Details of simulations and test results are also included. The adjustable DBA and the associated BFN technology under this development are patent pending and are strictly proprietary to Powerwave Technologies.

A brief summary of the concept, design goals, and critical parameters of the proposed DBA is given in Section II. Section III describes theoretical background and feasibility of the implementation using a three-column array, while the proposed concept and implementation of the compact 3-to-2 microstrip beam forming network is given in Section IV. The geometry and performances of the broad beamwidth aperture-coupled patch is described in Section V. EM Simulations using HFSS and measured radiations patterns using spherical near-field range is given in Section VI and Section VII, respectively. Section VIII briefly compares and discusses the theoretical and measured results. Section IX concludes the paper.

## II. DUAL BEAM ARRAY

Fig. 1 shows the concept of the proposed dual beam array. Each of the overlapping beams has a typical HPBW of 33 deg. The design is such that the combined pattern of the array matches the required coverage of a typical cellular sector (65 degree coverage). The dual beam array can potentially increase the overall capacity because of the narrower beams and higher antenna directivities. However, because the two beams are adjacent to each other, compromises in antenna performances such as signal interference between the two beams and hand-over loss are often required. By using a three-column array with a movable center column, these compromises can be made adjustable in-situ to meet any particular need.

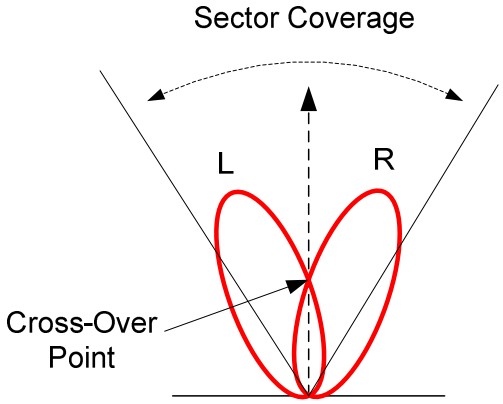



Fig.1. Dual beam concept

Fig. 2 shows a schematic of the 3x2 beam forming network (BFN). Two simultaneous beams are formed using three separate columns of linear array, which consists of a number of dual polarized aperture-couple patches (ACP).

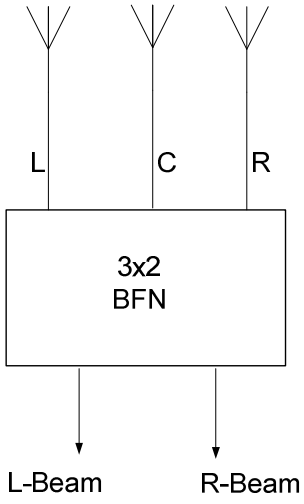



Fig. 2. Three-column dual beam BFN

Table 1 gives a summary of the desired antenna parameters of the DBA. In this case, the antenna is designed for UMTS band, 1700MHz to 2200MHz. The total coverage angle is typically 67 degree. Beamwidth of the dual beams is expected to be approximately 33 degree.

Key parameters of the dual-beam array are: (1) Aperture size and overall RF beam forming losses, (2) signal interference between the two adjacent beams, (3) loss factor at the cross-over (hand-over) point. A typical DBA using conventional beam forming technique cannot deliver optimum performance in all these aspects.

Table 1: Antenna Parameters

| | Parameter | |
|---|---|---|
| 1 | Frequency | 1710 - 2150 MHz |
| 2 | Az Beamwidth | Individual : 33 to 45 deg |
| | | Combined : 67 to 92 deg |
| 3 | El Beamwidth | 6 - 8 deg |
| 4 | Directivity | 19.7 to 21.3 dBi |
| 5 | Polarization | ±45 deg |
| 6 | El Tilt | 6-7 deg (RET) |
| 7 | Cross-pol Level | < -20dB |
| 8 | Cross-over loss | 2.5 dB to 6.5 dB |
| 9 | Upper SLL | -18 dB (relative to main beam, 30 deg above horizon) |
| 10 | Port-port Isolation | 30 dB |
| 11 | F/B | 30 dB |
| 12 | Return Loss | -14 dB |
| 13 | Power Handling | 200W (CW) |
| 14 | Passive Intermod | -150 dBc (3rd order) |
| 15 | Mechanical | |
| | Length | 1.4m |
| | Width | 28cm |
| | Depth | 15cm |

The goal of this study is to develop a DBA which will permit trade-off and compromises between these parameters in order to achieve the optimum performance for a given application. The design will allow trade-off between signal interference and the cross-over loss between the two beams.

### III. 3-COLUMN DUAL BEAM ARRAY

The proposed antenna can be perceived as a superposition of two partially-filled ring arrays. The azimuth pattern of the two-ring circular array can be varied by adjusting the relative dimension of the radiuses of the rings. Fig. 3 depicts the theoretical model of a general two-ring array.

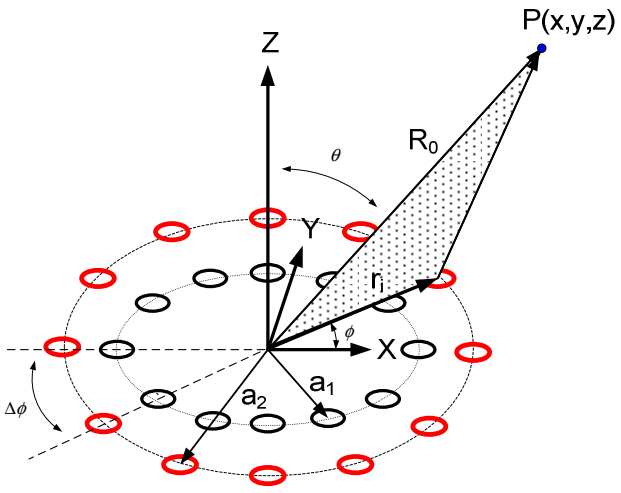



Fig.3. Theoretical two-ring concept of the dual beam array

A simplified mathematical model for the total field of the three-column two-ring array is

$$\hat{F}(\theta,\phi) = I_1 * \hat{E}_1(\theta,\phi) * \exp[\,jka_1 \sin\theta\cos(\phi-\Delta\phi)] + I_0 * \hat{E}_0(\theta,\phi) * \exp[\,jka_2 \sin\theta\cos(\phi)] + I_{-1} * \hat{E}_{-1}(\theta,\phi) * \exp[\,jka_1 \sin\theta\cos(\phi+\Delta\phi)] \quad (1)$$

Where,

$$\hat{E}_1(\theta,\phi) = \hat{E}_0(\theta,\phi-\Delta\phi) \quad (2)$$

$$\hat{E}_{-1}(\theta,\phi) = \hat{E}_0(\theta,\phi+\Delta\phi) \quad (3)$$

$\hat{E}_0(\theta,\phi)$ represents element pattern of radiators on the center column. $I_1$ and $I_{-1}$ are complex coefficients of excitations for radiators 1 and -1. Radiuses of the two rings are $a_1$ and $a_2$, respectively. $k$ is the wave number. $\Delta\phi$ is the angular spacing between radiating elements.

Fig. 4 shows the front and cross-sectional views of the 30-element, three-column, cylindrical sector array. The array is designed to operate in a typical wireless communication band, 1700 MHz to 2200 MHz.

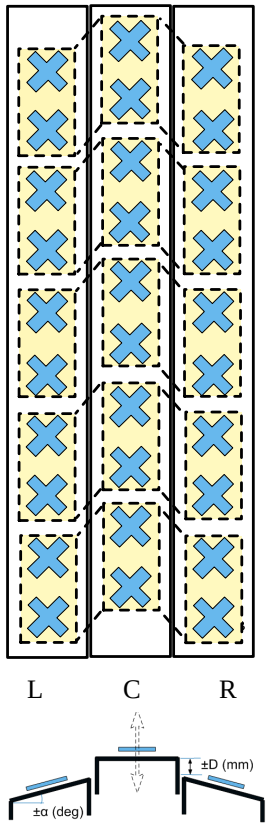



Fig. 4. Three-column array with adjustable center column

The radiating elements are aperture-coupled patches. Each of the three linear arrays is mounted on a separate reflector. Radius of the inside ring, $a_1$, is determined by the subtend angle of the two edge reflectors, **α**. Radius of the outside ring, $a_2$, is set and adjusted by the vertical displacement, D, which can be varied between -5mm to +25mm in the direction of the mechanical boresight direction.

The relative slope of the two edge columns, α, with respect to the center column, is critical in achieving the required pattern shapes and beam cross-over loss. Typically, this angle is set between 20 degree and 30 degree with respect to the center column. With these parameters, the dual beam patterns can be maintained over a relatively broad frequency bandwidth. Fig. 5 shows a simulated 65˚ full coverage beam pattern and three independent narrow beams at 2200 MHz. For these analyses, the angle (α) is set at 20 degree. The half-power-beamwidth (HPBW) of each individual narrow beam is approximately 33 degree, which provides combined azimuth coverage of 65 degree for a typical cell sector.

For some applications, it is possible that signals are transmitted at one polarization using the broad beam pattern (65 degree), while the receive signals at other polarization using the two narrow beams. For other applications, both transmit and receive signals operate using the dual beams. A narrow center beam can also be formed if necessary. Combinations of these can be formed depending on the application.

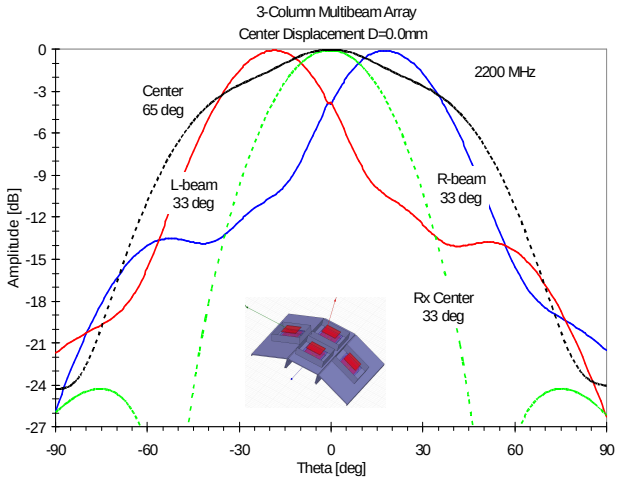



Fig. 5. Simulated beam patterns of the three-column array

With the displacement distance of the center column set at D=0mm (reflector surface of the center column levels with the top edges of the two edge columns), the three narrow beams and the 65˚ broad beam pattern can be formed using the amplitude and phase excitations given in Table 2.

Table 2. Amplitude and Phase excitations for various beams

| Beam | Left (L) | Center (C) | Right (R) |
|---|---|---|---|
| **Left 33˚** | (1, 0˚+Δφ) | (0.74, 0˚) | (0.34,-180˚+Δφ) |
| **Center 33˚** | (0.5, 0˚+Δφ) | (1.0, -30˚) | (0.5, 0˚+Δφ) |
| **Right 33˚** | (0.34,-180˚+Δφ) | (0.74, 0˚) | (1.0, 0˚+Δφ) |
| **65˚ Beam** | (0.5, -45˚) | (1.0, 0˚) | (0.5, -45˚) |

Δφ represents an additional phase adjustment, which can be introduced into the excitation by the addition of a fixed line length on the feed line, or varying the relative displacement distance of the center column, D. This adjustable phase allows further optimization of beam parameters such as the beam cross-over losses and pattern discrimination.

## IV. BEAM FORMING CIRCUIT

Fig. 6 shows the amplitude and phase excitations of the 3-to-2 beam forming network for the dual beam patterns. Fig. 7 and 8 show the equivalent signal diagram and the implementation of the compact dual beam former using micorstrip transmission line method.

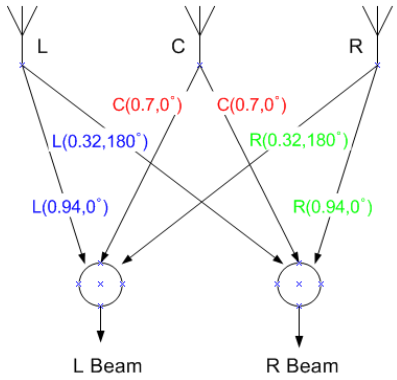



Fig. 6. Excitations function of the dual beam former

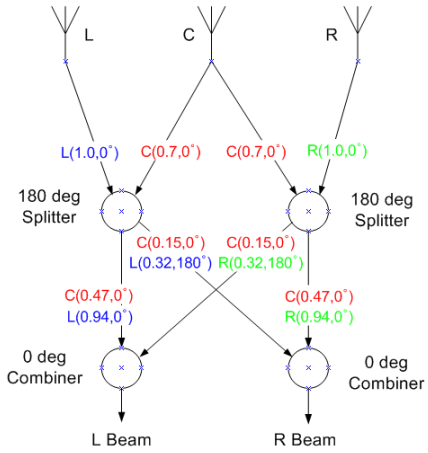



Fig. 7. Equivalent excitation signal diagram of the dual beam former

The proposed 3-to-2 beam former is implemented using two unequally-split 180deg splitters and two in-phase Wilkinson combiners. This simple implementation has an added advantage of excellent isolations between antenna ports as shown in Fig. 9, simulated using Agilent ADS. These are inherent merits from port cancellation at the sum and difference ports of the 180 degree splitters.

Another significant advantage of this implementation is the low beam-split signal losses. Each of the dual beams are formed using a 180 degree splitter (10 dB) on the corresponding edge column and a 3dB (0 deg) splitter on the center column. The total signal loss due to the beam split is less than 0.5 dB in comparison to the single beam case (center 33˚), which has an excitation taper of (0.5, 1.0, 0.5). This is a direct result of the optimum phase and amplitude tapers from the array configuration. Furthermore, the path loss is also minimized because of the compact circuit design.

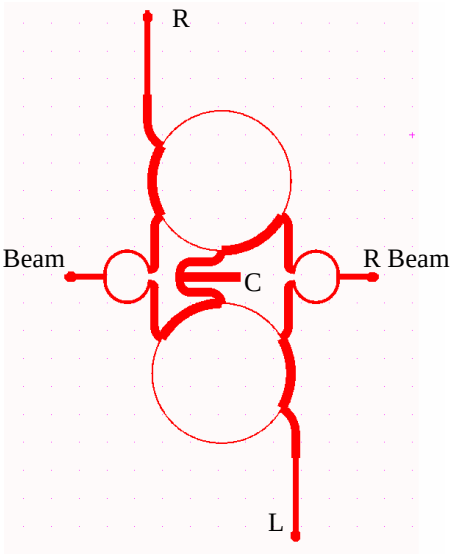



Fig. 8. Microstrip implementation of the compact dual beam former

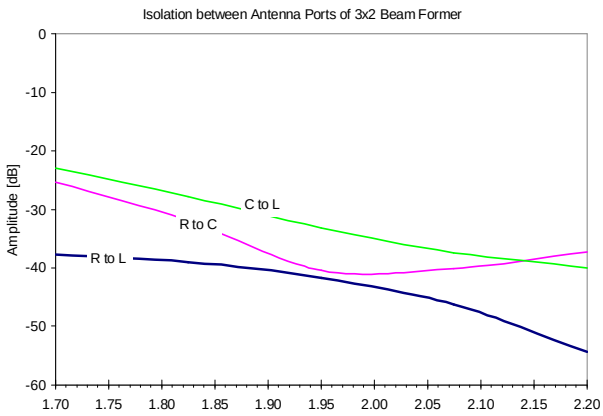



Fig. 9. Isolation between antenna ports of dual beam former

## V. APERTURE-COUPLED PATCH (ACP)

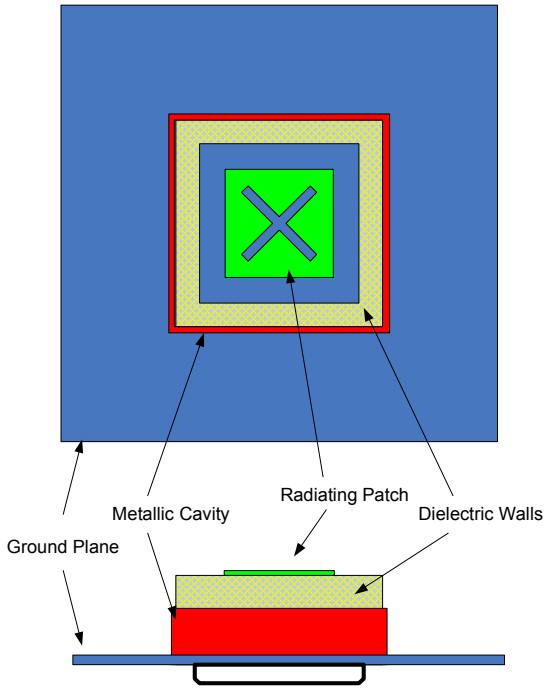
One of the critical elements of this development is the dual-polarized radiators with relatively broad beamwidth over a large operating frequency. For optimum array performance, it is desirable that the azimuth pattern of the radiator can be adjusted to allow optimization of the azimuth beam pattern, port isolations, and the overall directivity of the array.

Use of metallic boundaries in the near-field enables broadening of HPBW of a patch antenna up to approximately 90 deg at the expense of the overall frequency bandwidth. The proposed method of dielectric fortification [5] between the radiator and the metallic boundaries provides a systematic means for significant broadening of HPBW over a large frequency bandwidth. Subsequently, this method allows optimization of array performance by selecting appropriate thickness or height of the dielectric material. The dielectric loading in this manner does not seem to degrade performance in the cross polarization pattern as long as the dielectric loading is symmetrical in all four directions.

Fig. 10 shows the isometric and cross-sectional views of a dielectric fortified stacked patch. In this case, the aperture-coupled stacked patch is dual linearly polarized. Two radiating patches are fed by a pair of orthogonal cross slots on the bottom ground. The radiating patches are centered in a square area with a perimeter formed by four dielectric walls with the outside dimension of approximately half-wave length. The outside surfaces of the dielectric walls can be backed by electrically conductive walls. This arrangement allows a compact construction of the patches. The top radiating patch can be conveniently flash-mounted on top of the dielectric walls, while the lower radiating patch are secured at a predetermined height from the ground via small recessed grooves cut onto the inside surfaces of the dielectric walls.

The electrically conductive layer may be of the equal height as the dielectric walls, or preferably recessed from the top of the dielectric walls for better frequency bandwidth. For a given dielectric material with a fixed height, the HPBW is directly proportional to the thickness of the dielectric walls.

A fullwave FEM model of a dual-polarized aperture-coupled stacked patch with dielectric fortification is simulated using the Ansoft HFSS. For the purposes of these demonstrations, FR-4 ($\varepsilon_r$=4.6) is assumed for dielectric material. The height of dielectric walls is fixed at 20mm and the height of the metallic boundaries is kept at 14mm. Fig. 11 shows the simulated azimuth HPBW for various thicknesses of dielectric walls. As indicated in the figure, the HPBW is varying from 70 degree to 125 degree for value of the dielectric thickness between 0 mm and 7.5mm.



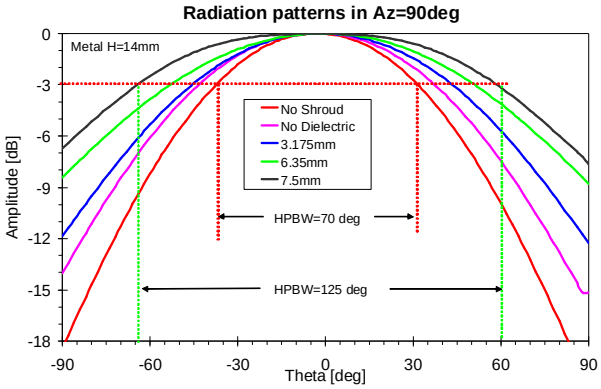




Fig. 11: HPBW of the dielectric fortified stacked patch

## VI. EM SIMULATIONS

A 4-element sub-array model of the three-column array is simulated using the Ansoft 3D full-wave Finite Element Method (FEM) HFSS. For these analyses, the subtend angle is set to 20 degree. Fig. 12 and 13 show the simulated azimuth patterns at 1700 MHz and 2200 MHz with the displacement distance (D) set at 0mm.

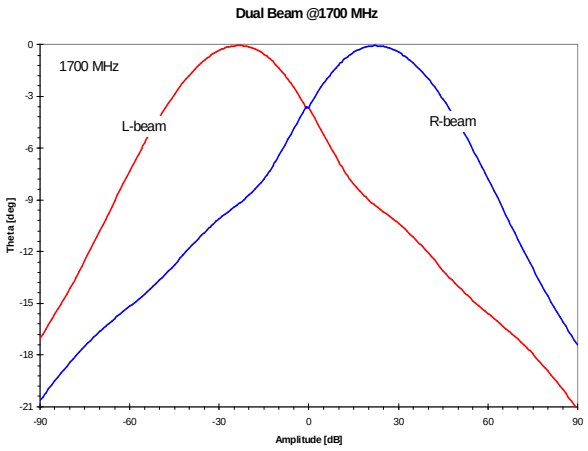



Fig. 12. Simulated dual beam patterns at 1700 MHz

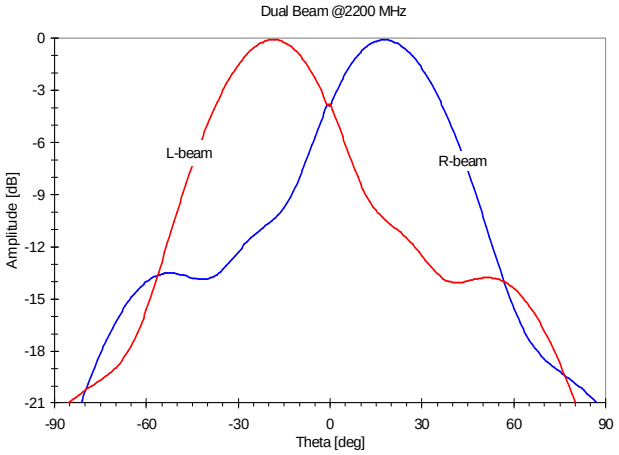



Fig. 13. Simulated dual beam patterns at 2200 MHz

At this setting, the beam cross-over loss at azimuth=0 deg is between 3.5dB@1700 MHz to 3.7dB@2200 MHz (D =0mm). These beam patterns and the cross-over losses can be varied by introducing an additional phase offset between the center column and the two edge columns using the adjustable displacement feature of the array. Fig 14 shows comparisons of the dual beam patterns for the displacement distance at various positions: -5mm, 0mm, and +15mm.

When displacement distance (D) is between -5mm and +15mm, the beam cross-over loss is varying between -1.6dB to -4.9 dB. The lowest beam cross-over loss is -1.6 dB when the displacement distance is at D=-10mm. This low cross-over loss, however, comes at the expense of pattern discrimination performance (4dB at beam peak).

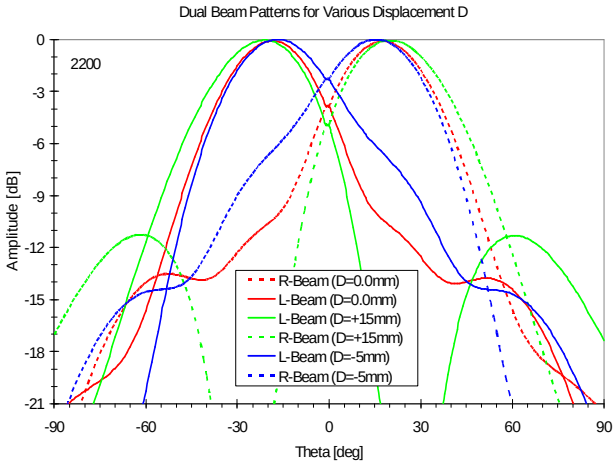



Fig. 14. Dual beam patterns for various displacement distance D

On the other hand, at D=+15mm, the dual beams has an optimum pattern discrimination of over 24dB at the expense of the beam cross-over loss of -4.9dB. Table 3 summarizes the dual beam performances for various displacement distance D.

Table 3. Pattern parameters at various displacement distance D

| Displacement Distance | Cross-over Loss | Discrimination At Peak | HPBW (deg) |
|---|---|---|---|
| D=-10 mm | -1.6 dB | 4 dB | 36 |
| D=-5.0mm | -2.3 dB | 6.5 dB | 35 |
| D= 0.0mm | -3.7 dB | 10.5 dB | 33 |
| D=+15mm | -4.9 dB | 24 dB | 35 |

These results clearly demonstrated the advantage of the variable displacement of the center column, which allows optimization of performance between hand-over loss (cross-over loss) and interference discrimination.

## VII. Measured Results

Fig. 15 show the prototype of the dual-beam array constructed based on the principle of the three-column variable beamwidth array presented in the previous sections. A total of five 3x2 microstrip BFNs are used to feed the 30-element array. Two 1-to-5 elevation power combiner & phase shifter (RET) are used to distribute the BFN Outputs of the left (L) and right (R) beams. This allows separate beam tilt of the R and the L beams in elevation plane between 0 and 7 degree.

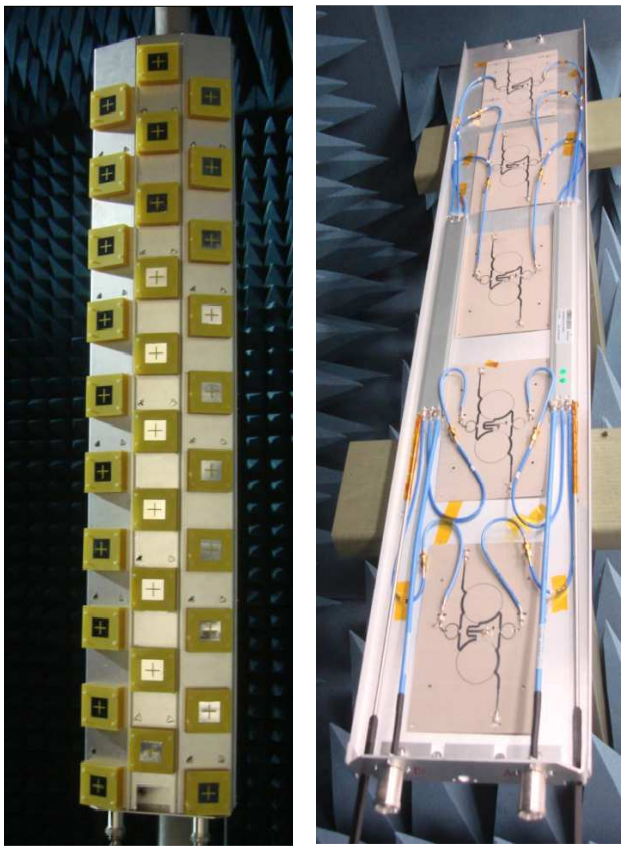


Fig.15.Prototype of the 3-Column DBA and BFN Feeds

Fig. 16 shows the detailed layout of the 3x2 BFN circuit. Outputs of the three linear arrays (Left, Center, Right) are fed to the BFNs at L, C and R, respectively. The Wilkinson combiners produce the Right and Left beams as indicated.

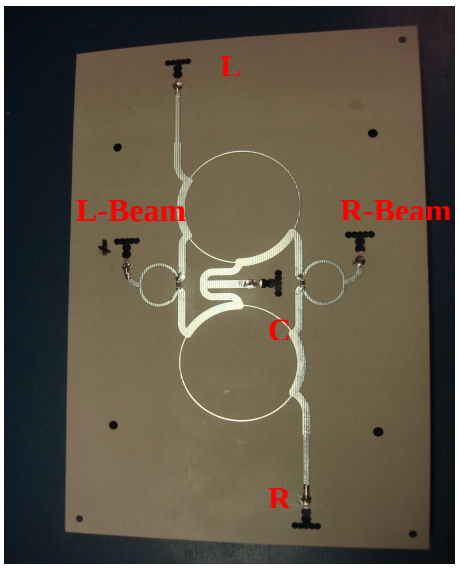



Fig.16. 3-Column Dual Beam Array and BFN Feeds

Table 4 gives the measured amplitudes and phases of the BFN.

Table 4: Measured BFN excitation functions

| | 1700 MHz | | 2200 MHz | |
|---|---|---|---|---|
| R-Beam | Amp | Phase | Amp | Phase |
| Center | 0.96 | 0.0 | 0.92 | 0.0 |
| Right | 1.0 | - 37.0 | 1.0 | -45.0 |
| Left | 0.44 | -200.0 | 0.376 | -223.0 |
| L-Beam | | | | |
| Center | 0.96 | 0.0 | 0.92 | 0.0 |
| Right | 0.44 | -200.0 | 0.376 | -223.0 |
| Left | 1.0 | -37.0 | 1.0 | -45.0 |

Antenna patterns of the array were measured in the Powerwave spherical near-field chamber in Santa Ana, California. Two polarizations (slant ±45deg) are measured at two RET settings (0 deg tilt and 7 deg tilt).

Fig. 17 shows the measured azimuth beam patterns (1710MHz, 1950MHz, 2150 MHz) of the DBA when the RET is set to 0 deg and center displacement D=-10mm. At this setting, the measured HPBW is between 32deg to 39deg with cross-over loss between -3.5dB to -3.9dB. The array produces very low cross-polar field components at well below -20 dB. Fig. 18 shows the measured azimuth beam patterns of the DBA when the RET remains set at 0 deg but the center displacement D is moved to -5mm. The HPBW is reduced very slightly, but the cross-over loss is increased to between -4.0dB to -4.4dB. Fig. 19 gives the measured patterns when the displacement is set to 0mm. The HPBW is reduced to between 31 deg and 37 deg, while the cross-over loss is approximately -4.9dB for all frequencies. As shown in Fig. 20, as the displacement is moved up to +10mm, the HPBW is increased to over 36deg and the cross over loss is over 5.9dB.

Fig. 21 to 23 show the azimuth patterns of the DBA when the RET is set to 7 deg. These results are similar to the previous when RET is set to 0deg. However, the cross-over losses are generally lower. For instance, the cross-over loss is reduced to -2.3 dB when the displacement D=-10mm. This is significantly lower than the previous case. However, this comes at the price of lower beam discrimination.

Fig. 24 gives the measured elevation patterns of the array at 1710MHz and 2150MHz when RET=0deg, for various displacement D. Similarly, Fig. 23 gives the elevation patterns for RET=7deg. Apparently, the offset distance (D) does not seem to affect the beam patterns in the elevation plane significantly. However, SLLs tend to increase as the offset distance decreases.

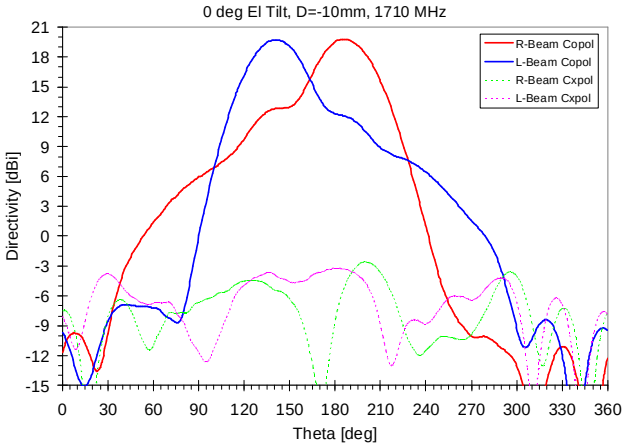



Fig.17(a) Measured DBA Az Pattern (0deg Tilt, D=-10mm, 1710MHz)

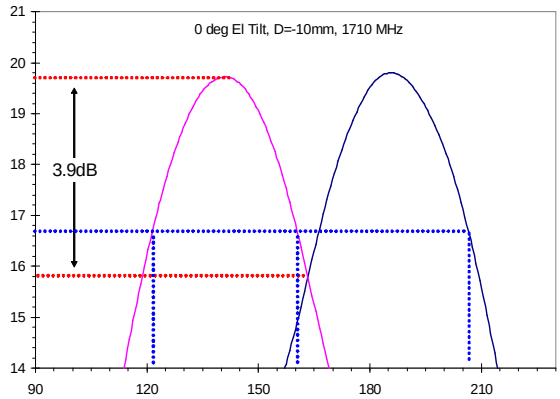



Fig.17(b) Measured cross-over loss (0deg Tilt, D=-10mm, 1710MHz)

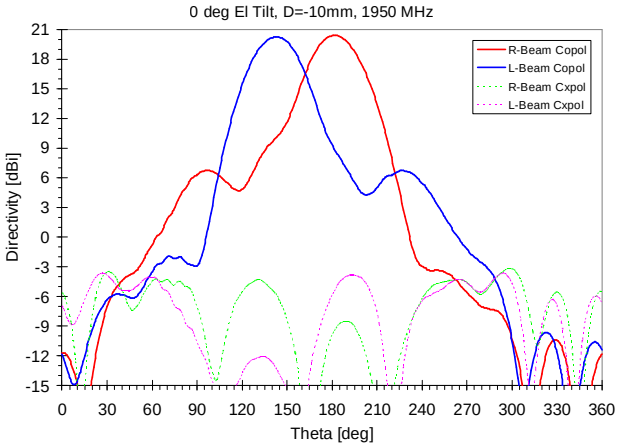



Fig.17(c) Measured DBA Az Pattern (0deg Tilt, D=-10mm, 1950MHz)

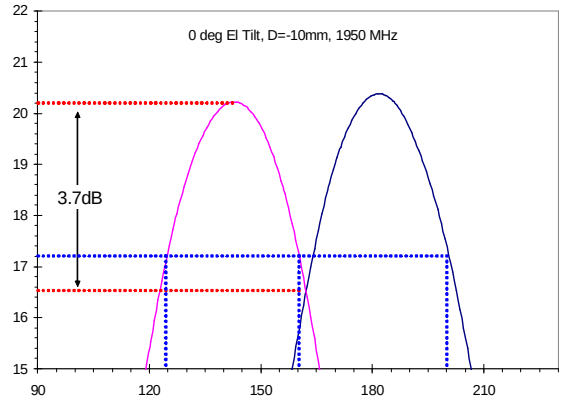



Fig.17(d) Measured cross-over loss (0deg Tilt, D=-10mm, 1950MHz)

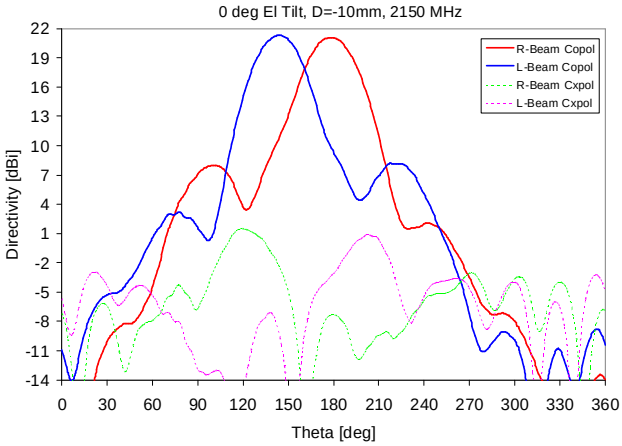



Fig.17(e) Measured DBA Az Pattern (0deg Tilt, D=-10mm, 2200MHz)

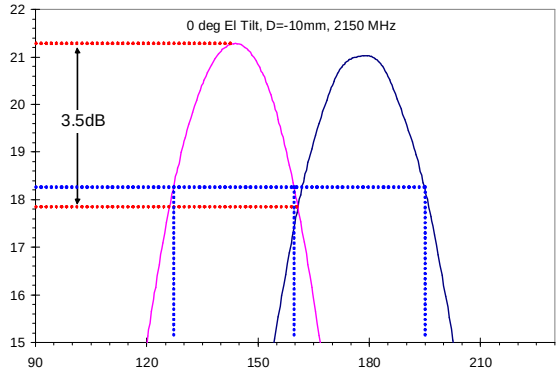



Fig.17(f) Measured cross-over loss (0deg Tilt, D=-10mm, 2200MHz)

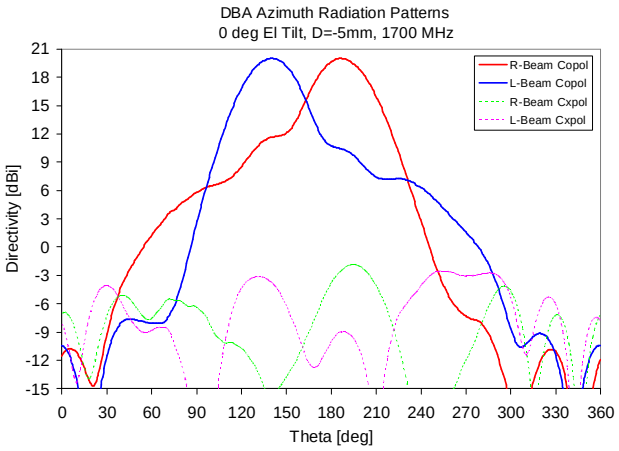



Fig.18(a) Measured DBA Az Pattern (0deg Tilt, D=-5mm, 1710MHz)

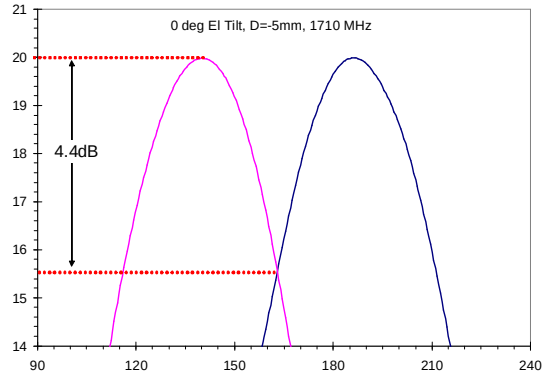



Fig.18(b) Measured cross-over loss (0deg Tilt, D=-5mm, 1710MHz)

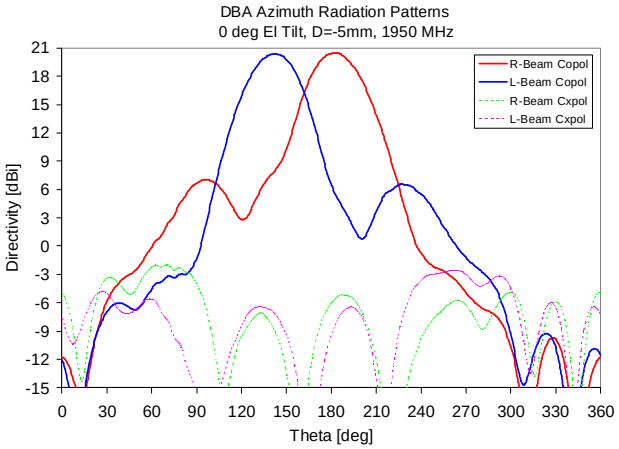



Fig.18(c) Measured DBA Az Pattern (0deg Tilt, D=-5mm, 1950MHz)

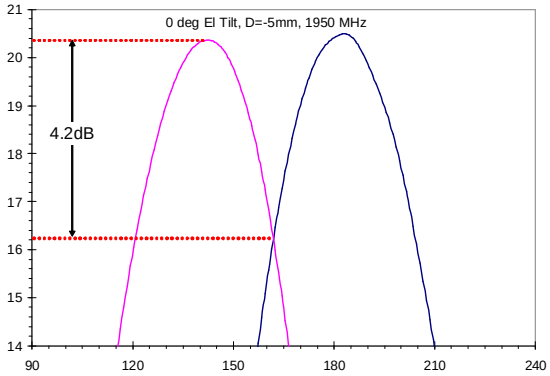



Fig.18(d) Measured cross-over loss (0deg Tilt, D=-5mm, 1950MHz)

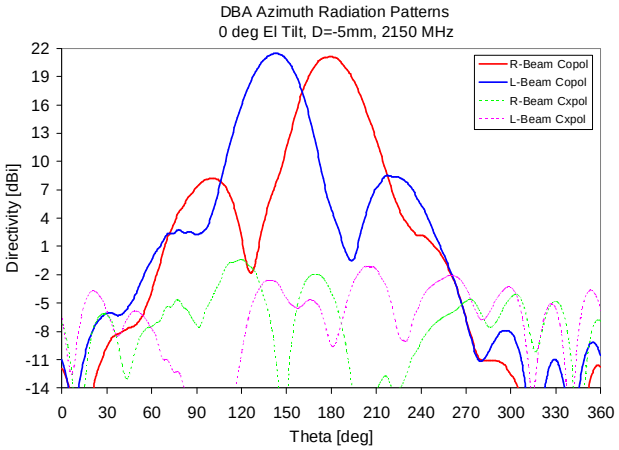



Fig.18(e) Measured DBA Az Pattern (0deg Tilt, D=-5mm, 2200MHz)

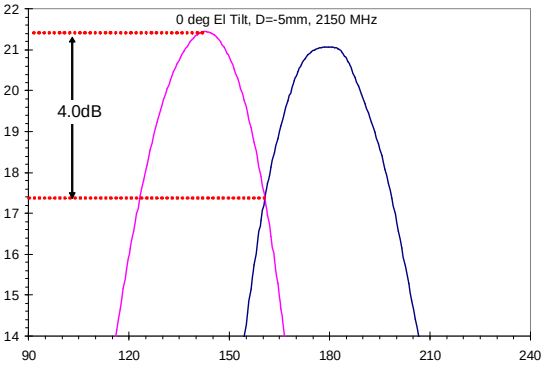



Fig.18(f) Measured cross-over loss (0deg Tilt, D=-5mm, 2200MHz)

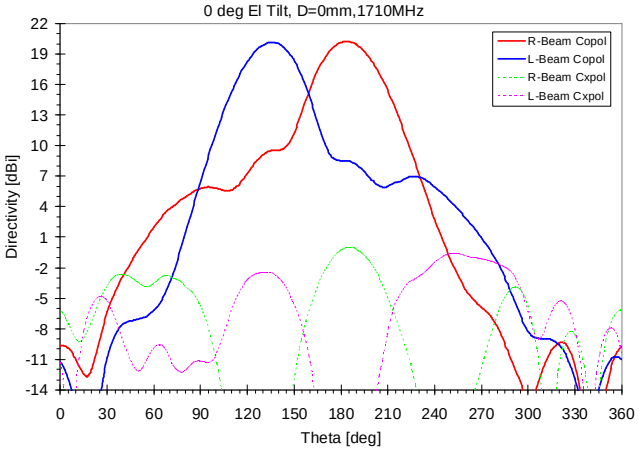



Fig.19(a) Measured DBA Az Pattern (0deg Tilt, D=0mm, 1710MHz)

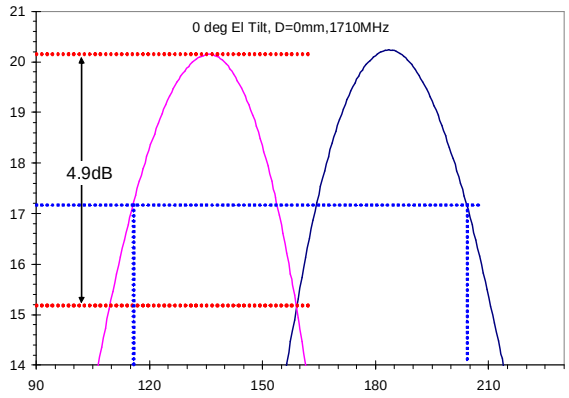



Fig.19(b) Measured cross-over loss (0deg Tilt, D=0mm, 1710MHz)

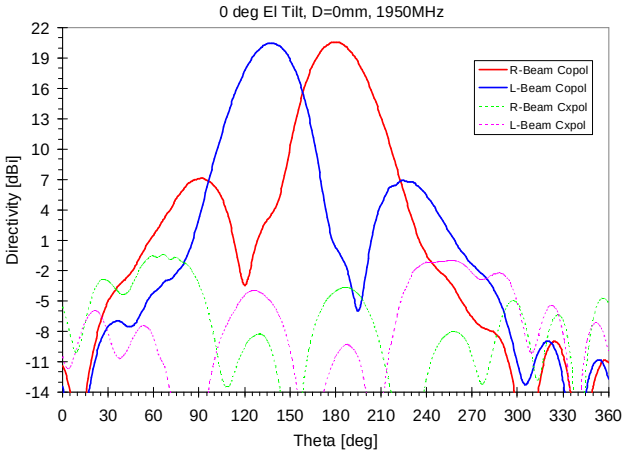



Fig.19(c) Measured DBA Az Pattern (0deg Tilt, D=0mm, 1950MHz)

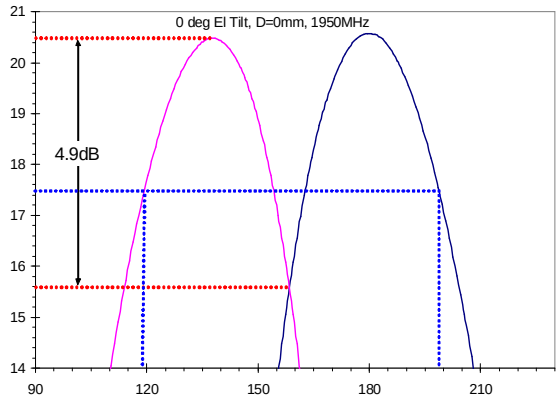



Fig.19(d) Measured cross-over loss (0deg Tilt, D=0mm, 1950MHz)

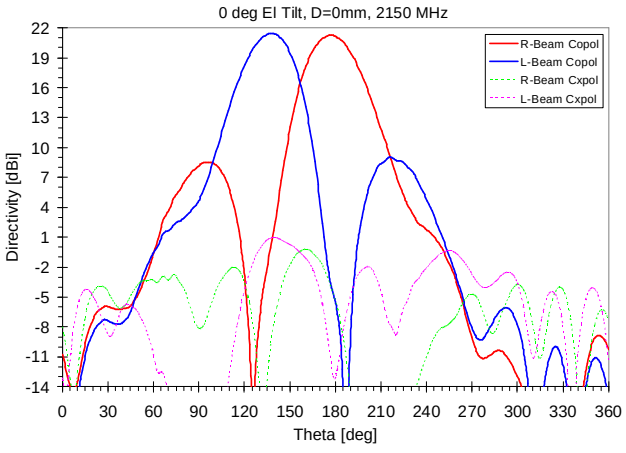



Fig.19(e) Measured DBA Az Pattern (0deg Tilt, D=0mm, 2200MHz)

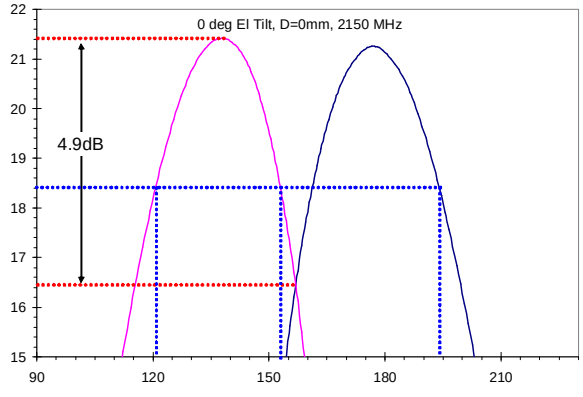



Fig.19(f) Measured cross-over loss (0deg Tilt, D=0mm, 2200MHz)

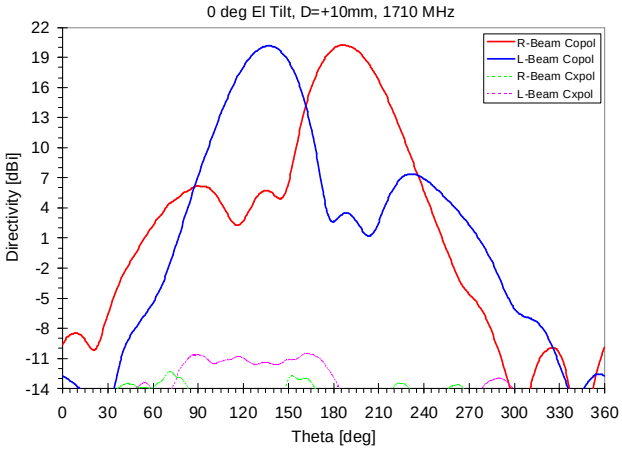



Fig.20(a) Measured DBA Az Pattern (0deg Tilt, D=+10mm, 1710MHz)

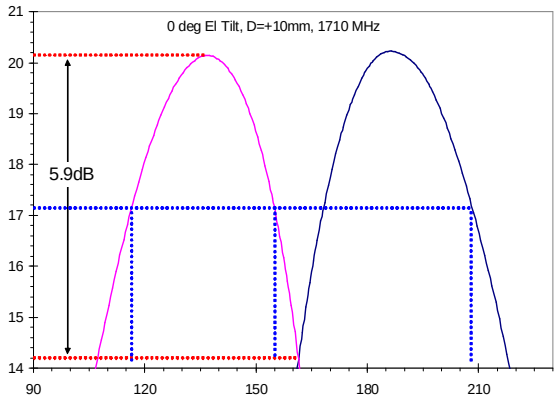



Fig.20(b) Measured cross-over loss (0deg Tilt, D=+10mm, 1710MHz)

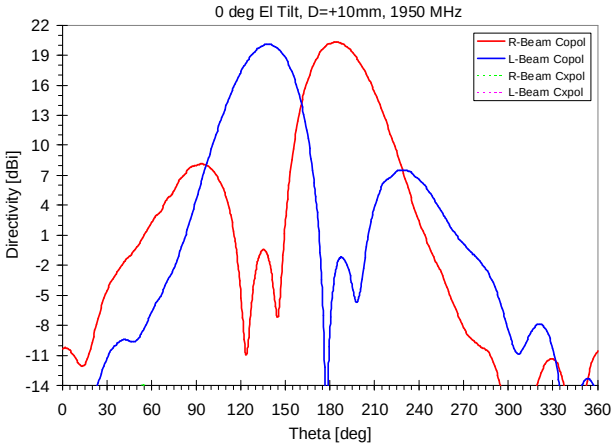



Fig.20(c) Measured DBA Az Pattern (0deg Tilt, D=+10mm, 1950MHz)

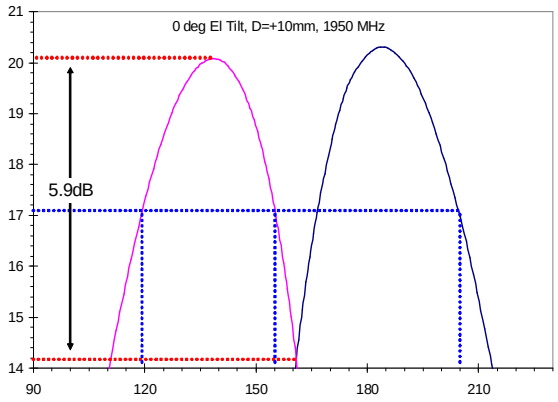



Fig.20(d) Measured cross-over loss (0deg Tilt, D=+10mm, 1950MHz)

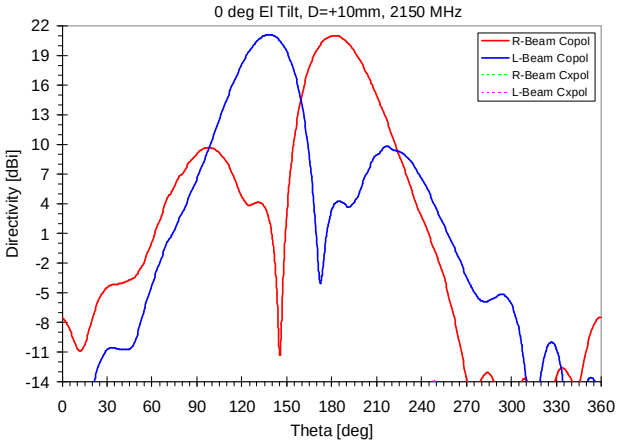



Fig.20(e) Measured DBA Az Pattern (0deg Tilt, D=+10mm, 2200MHz)

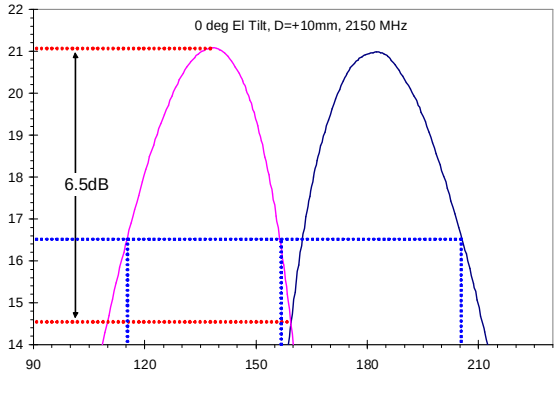



Fig.20(f) Measured cross-over loss (0deg Tilt, D=+10mm, 2200MHz)

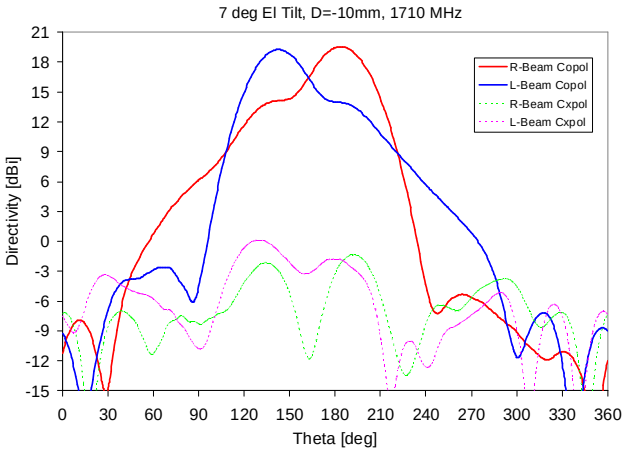



Fig.21(a) Measured DBA Az Pattern (7deg Tilt, D=-10mm, 1710MHz)

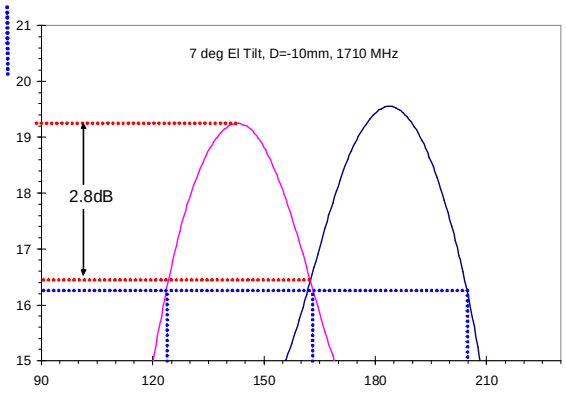



Fig.21(b) Measured cross-over loss (7deg Tilt, D=-10mm, 1710MHz)

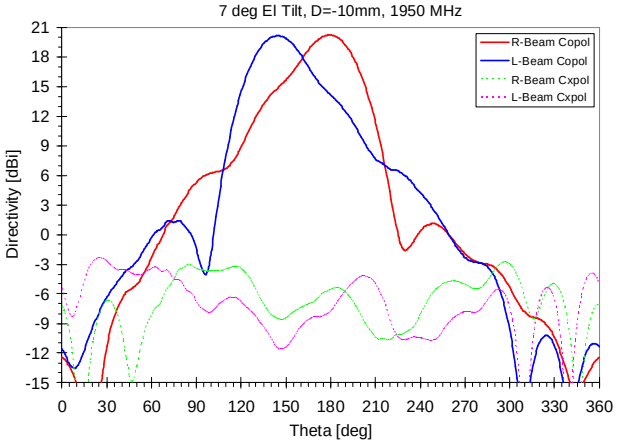



Fig.21(c) Measured DBA Az Pattern (7deg Tilt, D=-10mm, 1950MHz)

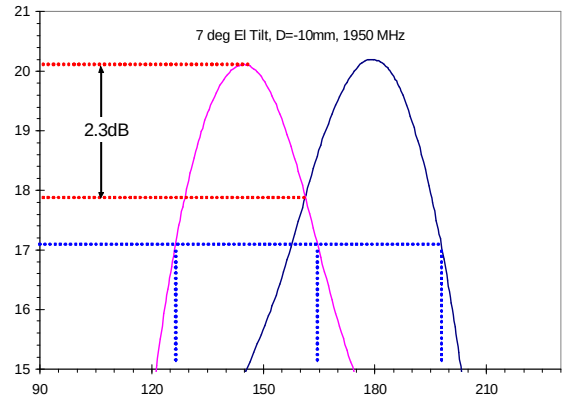



Fig.21(d) Measured cross-over loss (7deg Tilt, D=-10mm, 1950MHz)

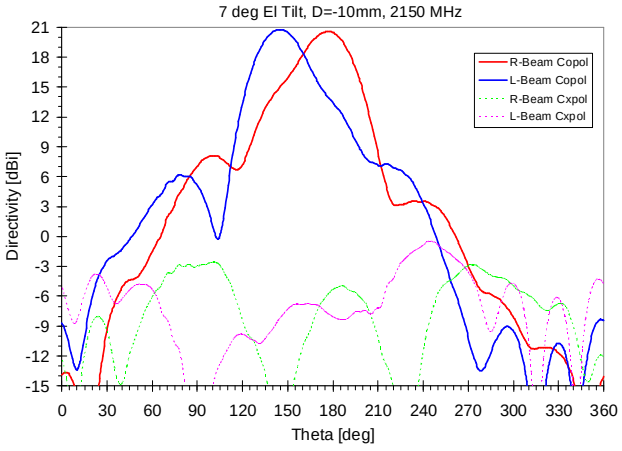



Fig.21(e) Measured DBA Az Pattern (7deg Tilt, D=-10mm, 2200MHz)

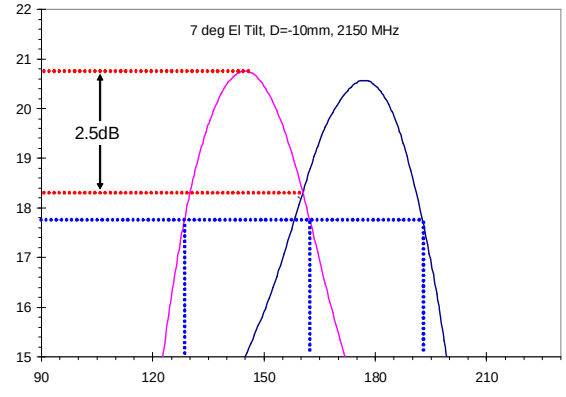



Fig.21(f) Measured cross-over loss (7deg Tilt, D=-10mm, 2200MHz)

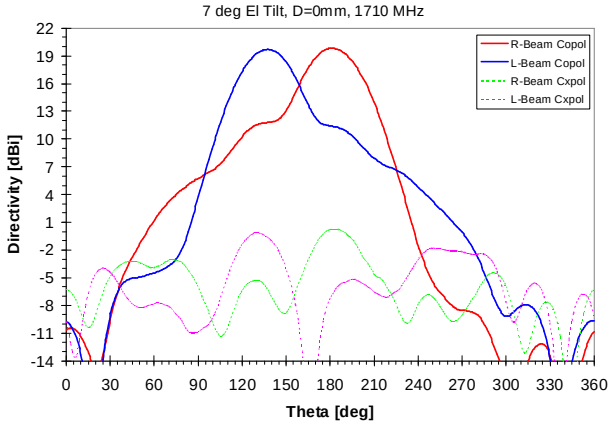



Fig.22(a) Measured DBA Az Pattern (7deg Tilt, D=0mm, 1710MHz)

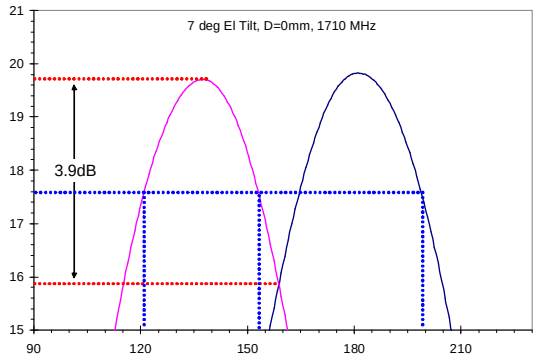



Fig.22(b) Measured cross-over loss (7deg Tilt, D=0mm, 1710MHz)

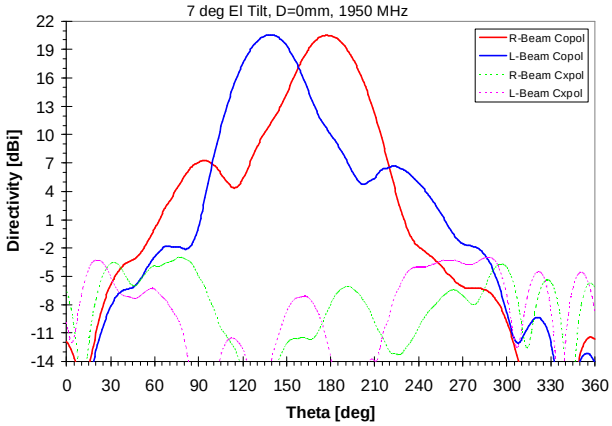



Fig.22(c) Measured DBA Az Pattern (7deg Tilt, D=0mm, 1950MHz)

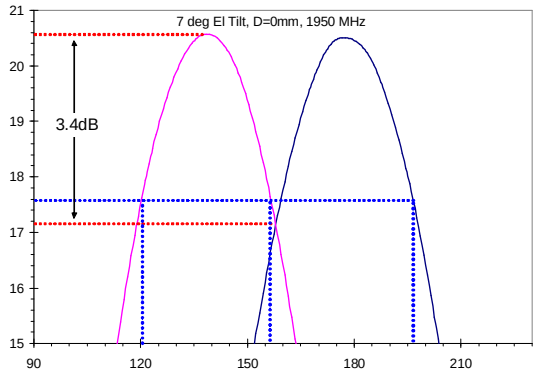



Fig.22(d) Measured cross-over loss (7deg Tilt, D=0mm, 1950MHz)

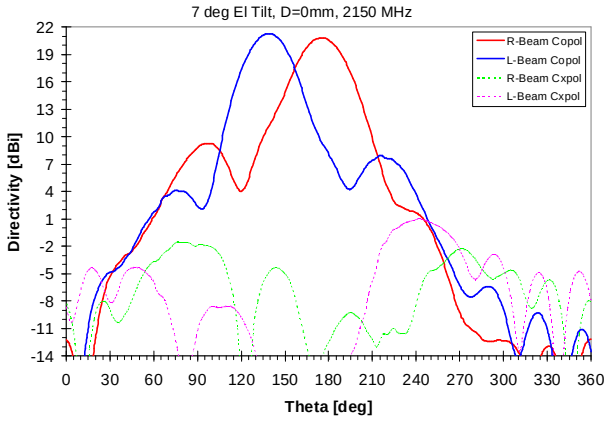



Fig.22(e) Measured DBA Az Pattern (7deg Tilt, D=0mm, 2200MHz)

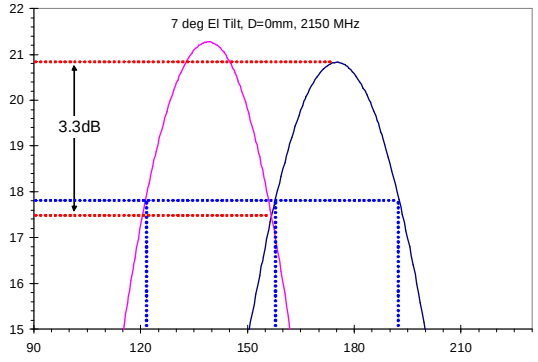



Fig.22(f) Measured cross-over loss (7deg Tilt, D=0mm, 2200MHz)

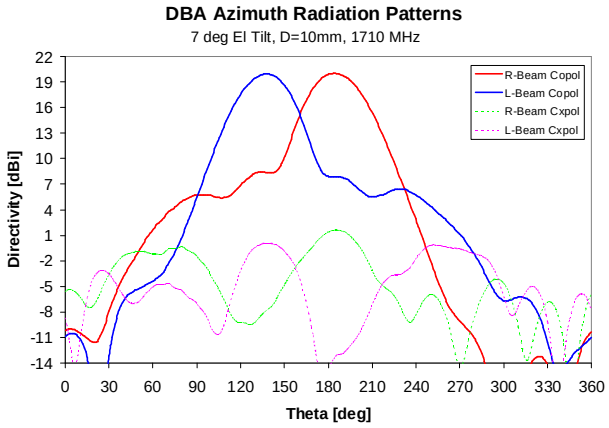



Fig.23(a) Measured DBA Az Pattern (7deg Tilt, D=+10mm, 1710MHz)

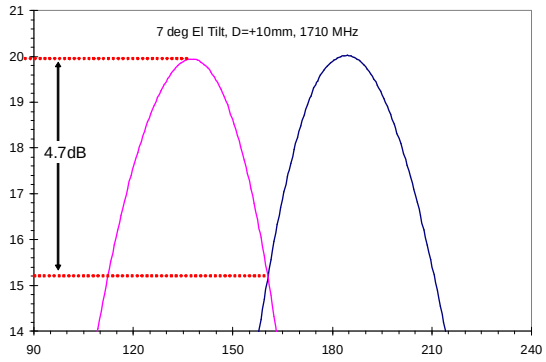



Fig.23(b) Measured cross-over loss (7deg Tilt, D=+10mm, 1710MHz)

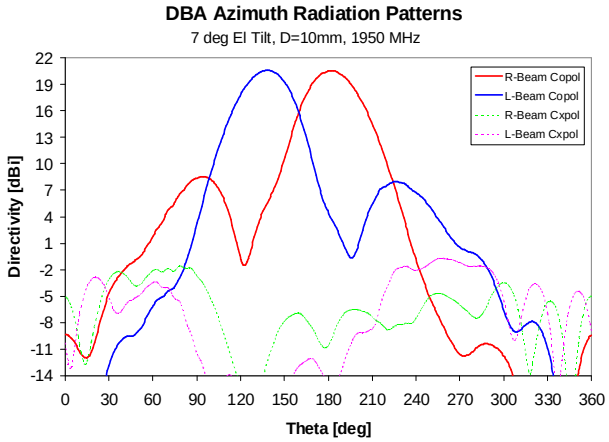



Fig.23(c) Measured DBA Az Pattern (7deg Tilt, D=+10mm, 1950MHz)

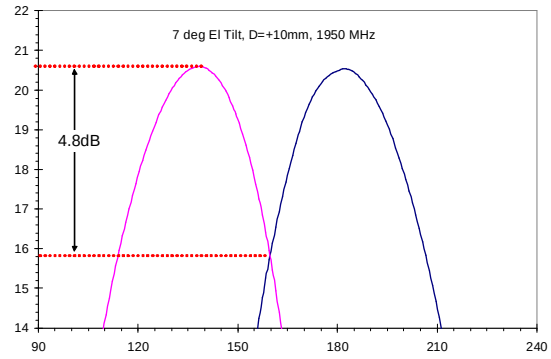



Fig.23(d) Measured cross-over loss (7deg Tilt, D=+10mm, 1950MHz)

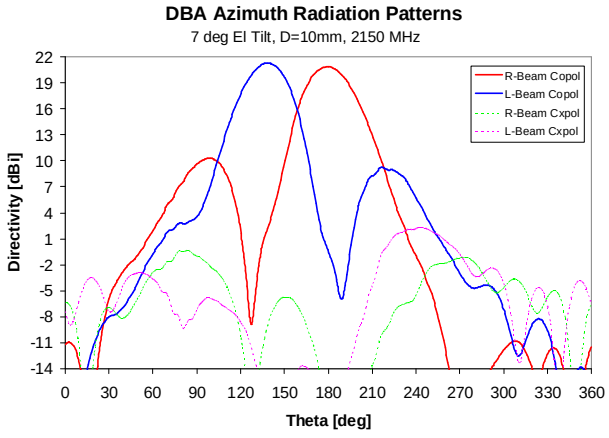



Fig.23(e) Measured DBA Az Pattern (7deg Tilt, D=+10mm, 2200MHz)

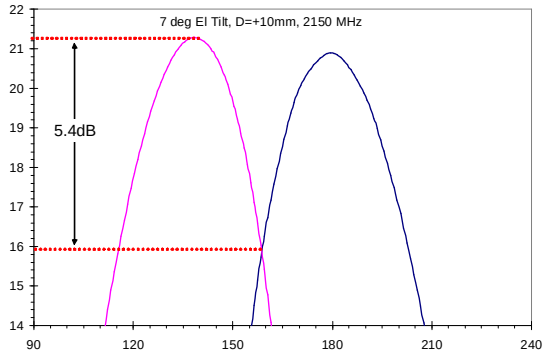



Fig.23(f) Measured cross-over loss (7deg Tilt, D=+10mm, 2200MHz)

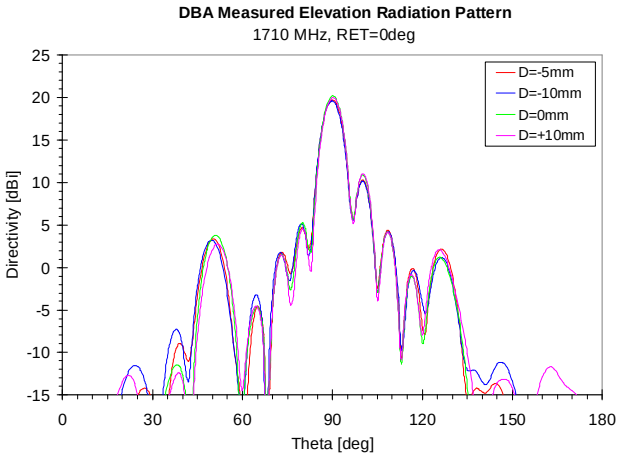



Fig.24(a) Measured DBA EL Pattern (0deg Tilt, 1710MHz)

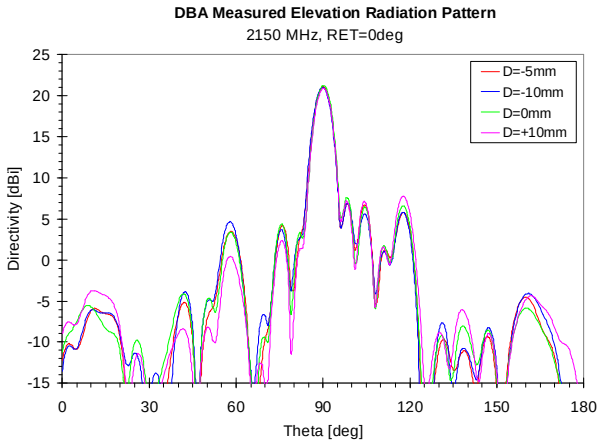



Fig.24(b) Measured DBA EL Pattern (0deg Tilt, 2150MHz)

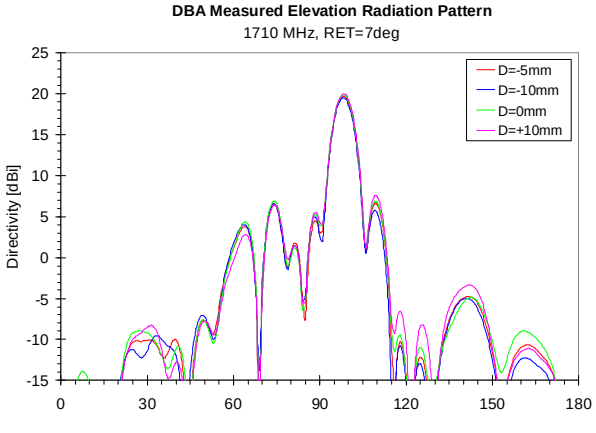



Fig.24(a) Measured DBA EL Pattern (7deg Tilt, 1710MHz)

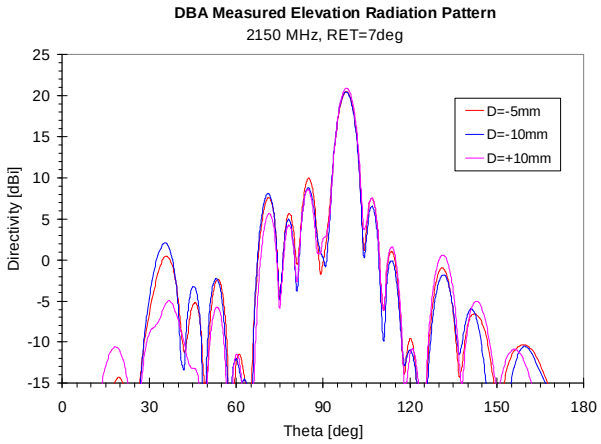



Fig.24(b) Measured EL Pattern (7deg Tilt, 2150MHz)

## VIII. DISCUSSION AND RESULTS COMPARISONS

Table 5 summarizes measured antenna parameters of the prototype DBA., including:

1) Cross-over loss
2) Discrimination at peak
3) HPBW
4) Directivity

Table 6 compares these parameters at various values of center displacement, D. HPBW of the two beams are, in general, within 31 deg and 41 deg for D within ±10mm. The measured directivities are approximately between 20 to 21dBi, relatively independent of the displacement D. The cross-over loss varies significantly as the center displacement changes. When D is set to -10mm, the cross-over loss is reduced to -2.3 dB for RET=7deg and -3.5dB when RET=7deg. However, at the same time, the pattern discrimination is also reduced to between 6 to 9dB. The pattern discrimination is improved as the center displacement increases. When D=+10mm, the pattern discrimination is over 13 dB, but the cross-over loss is over 4.7 dB.

The HPBW of the elevation patterns are generally between 6 to 8 deg. A maximum elevation tilt of 7 deg is feasible using the current Powerwave RET phase shifter. Measured elevation patterns at 7deg tilt indicate a slightly higher SLL, above 16dB. Fig. 24 and 25 compare HFSS simulations and measured patterns (1700 MHz and 2200 MHz) in the azimuth plane for D=0.0mm and -5.0mm. These results show very good correlations between the simulations and measured results. However, the HPBW is somewhat better correlated when the displacement D is above 0 mm and the cross-over loss is better predicted when D is below 0mm. Nevertheless, measured patterns and the HFSS simulated results compare relatively well.

Table 5: Summary of measured DBA parameters

| Item | Parameter | |
|---|---|---|
| 1 | Frequency | 1710 - 2150 MHz |
| 2 | Az Beamwidth | Individual : 31 to 41 deg |
| | | Combined : 67 to 92 deg |
| 3 | El Beamwidth | 6 - 8 deg |
| 4 | Directivity | 19.7 to 21.3 dBi |
| 5 | Cross Pol Level | < -20 dB |
| 5 | Polarization | ±45 deg |
| 6 | El Tilt | 6-7 deg (RET) |
| 7 | Cross-Over loss | 2.3 to 6.5 dB |

Table 6: Pattern parameters for various displacement distance D

| Distance (D) | Cross-over Loss (dB) | Discrimination At Peak (dB) | HPBW (deg) | Dmax (dBi) |
|---|---|---|---|---|
| RET=0deg | | | | |
| D=-10 mm | 3.5 - 3.9 | >9 | 32 - 39 | 19.7 – 21.3 |
| D= 0.0mm | 4.9 | >12 | 31 - 37 | 20.2 – 21.4 |
| D=+10mm | 5.9 - 6.5 | >16 | 36 - 41 | 20.1 – 21.1 |
| RET=7deg | | | | |
| D=-10 mm | 2.3 - 2.8 | >6 | 34 - 40 | 19.4 – 20.8 |
| D= 0.0mm | 3.3 – 3.9 | >9 | 32 - 36 | 19.7 – 21.3 |
| D=+10mm | 4.7 – 5.4 | >13 | 31 - 37 | 20.0 – 21.3 |

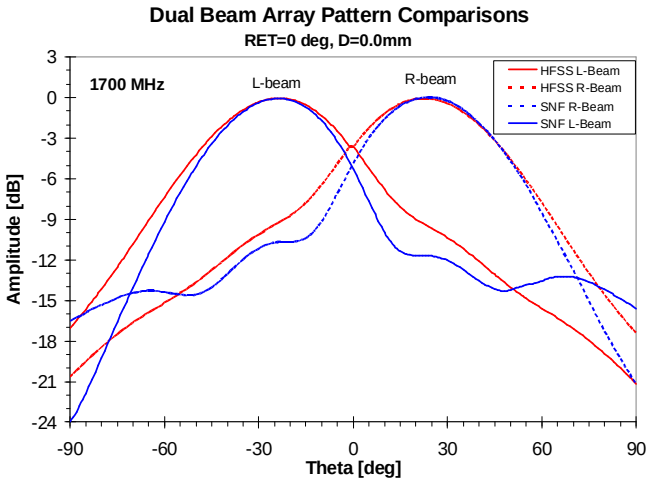



Fig.25(a) Pattern comparison at 1700 MHz, D= 0.0mm

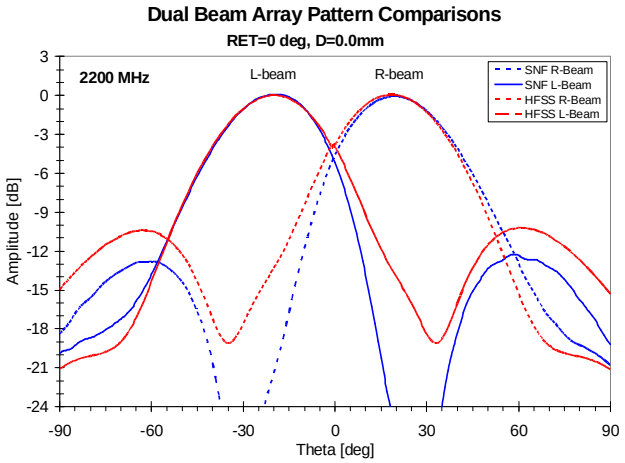



Fig.25 (b) Pattern comparison at 2200MHz, D=0.0mm

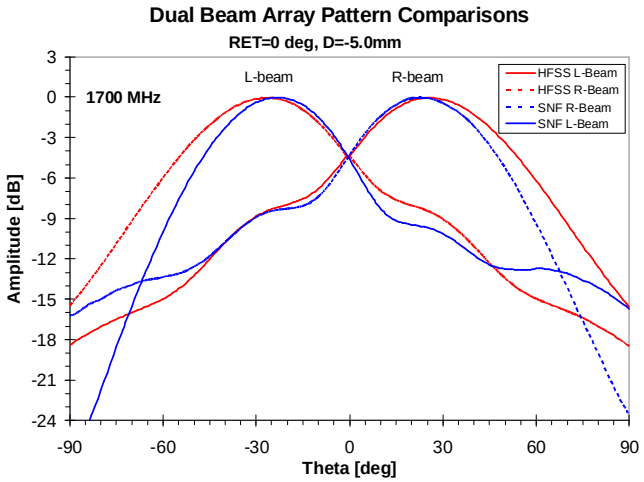



Fig.26(a) Pattern comparison at 1700 MHz, D=-5.0mm

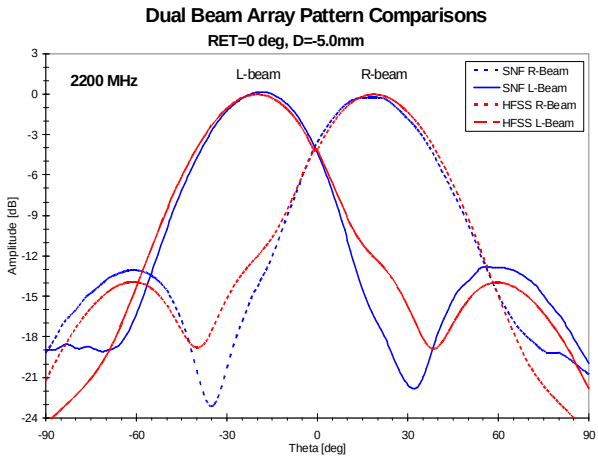



Fig.26(b) Pattern comparison at 2200 MHz, D=-5.0mm

## IX. CONCLUSION

Concept of an adjustable dual beam array for cell site enhancement of wireless base station is presented. This method allows for an increase in network capacity through higher order sectorization. The concept is realized using an adjustable three-column array and a compact, low-loss beam former. This structure produces two symmetrical narrow beams with respect to the azimuth boresight within a cellular sector. Radiation patterns of the two beams are adjustable for optimization of coverage of a cellular sector to minimize beam-split loss, or to optimize pattern discrimination and HPBW performance.

The array is capable of delivering more than two beams using a suitable BFN circuit. Theoretically, the device is capable of producing a broad beam for the full azimuth coverage and three narrow beams, simultaneously. It is therefore feasible for the array to transmit signal at one polarization using the broad beam pattern (65 degree), while receive signals at other polarization using the two narrow beams for diversity. For other applications, the dual beam can be used to transmit and to receive simultaneously. The issue with cross-over loss between beams can also be eliminated using three narrow beams, if required. This method allows an effective increase in the overall network capacity as a result of low-loss and narrow beam patterns.

A full array prototype was built and radiation patterns are also presented. The measured results are well correlated with the EM simulated results.

The adjustable DBA and the associated BFN technology under this development are patent pending (Application #12/252,334 and #12/175,725) and are strictly proprietary to Powerwave Technologies.

## REFERENCES

[1] S. Foo, B. Vassilakis, “Adjustable dual beam base station antenna,” *Wireless and Mobile Communications, 2008, ICWMC’08, The Fourth International Conference on*, July 27 2008-Aug 01 2008, pp315-320.

[2] R. J. Mailoux, *Phased Array Antenna Handbook*, second edition. Boston, MA: Artech House, 2005.

[3] R. C. Hansen, *Phased Array Antennas*. New York: John Wiley & Sons, 2005.

[4] H. J. Moody, ”The systematic design of the Butler matrix,” *IEEE Trans. On Antennas & Prog.*, vol.12, Issue 6, 1964, pp.786-788.

[5] S. Foo, B. Vassilakis, “Dielectric fortification for wide-beamwidth patch arrays,” *Antennas & Propagation. Society International Symposium 2008, AP-S 2008*, 5-11 July 2008, pp.1-4.

# Robustness in Sensor Networks: Difference Between Self-Organized Control and Centralized Control

Yuichi Kiri
Graduate School of Information Science and Technology
Osaka University
1-5, Yamadaoka, Suita-shi
565-0871 Osaka, Japan
y-kiri@ist.osaka-u.ac.jp

Masashi Sugano
School of Comprehensive Rehabilitation
Osaka Prefecture Univ.
3-7-30, Habikino, Habikino-shi
583-8555 Osaka, Japan
sugano@rehab.osakafu-u.ac.jp

Masayuki Murata
Graduate School of Information Science and Technology
Osaka University
1-5, Yamadaoka, Suita-shi
565-0871 Osaka, Japan
murata@ist.osaka-u.ac.jp

***Abstract*—Self-organized control has received significant attention in the area of networking, and one of the main factors for this attention is its robustness. However, it should be stressed that deciding whether self-organized control is robust or not is not a trivial task. Even if it is in fact robust, the factors underlying its robustness have not yet been explored in sufficient detail. In this paper, we provide the first quantitative demonstration of the superior robustness of self-organized control through comparison with centralized control in a sensor network scenario. Through simulation experiments, we show that self-organized control maintains the functionality of its data collection even in a variety of perturbations. In addition, we point out that the difference in the robustness of the abovementioned control schemes stems from the degree to which the comprehension of a given node about the state of the network depends on information obtained from other nodes.**

***Keywords*—sensor network; self-organized control; centralized control; robustness; simulation**

## I. Introduction

As networks are becoming increasingly larger and more complex, a critical issue in today's dynamically changing and uncertain environments is to maintain the functionality of networks in a manner which allows them to adapt to environmental changes. A control scheme which maintains the performance even when the network state changes dramatically or unforeseeable circumstances occur is preferable for present and future networks, even if the basic network performance in such cases is inferior to that of networks operating with other control schemes. The property which allows a system to maintain its functionality despite external and internal perturbations is called "robustness" [15]. In this age when networks play an essential role in our everyday lives, the robustness of networks is becoming increasingly important.

Distributed control has been said to be superior to centralized control with respect to robustness. Currently, a type of distributed control scheme which is beginning to attract considerable attention is one of self-organized control [10][20]. The communication networks based on such a self-organized control are considered to be suitable as a network which consists of movable nodes like many persons or cars, and a network used in the situation where environmental variation is remarkable, like disaster sites. In this control scheme, each component autonomously decides the following action on the basis of local information, and the simple microscopic actions of the components collectively provide structure and functionality at macroscopic level without any centralized coordination [19]. Such behavior is distinct from plain distributed control, where individual components act autonomously but depend on global information. Although scalability, adaptability, and fault tolerance, which are included in the concept of robustness in a broad sense, are "known" as properties inherent to self-organized control, we stress that this knowledge is certainly not trivial. Even assuming that the notion of robustness is true, to the best of our knowledge the reasons why self-organized control is robust and the factors which determine the superiority of its robustness as compared to other control schemes have not been examined with sufficient rigor.

In our previous work [13][14], we provided quantitative evidence of the robustness of self-organized control with respect to transmission errors and node failures, and concluded that the robustness of the self-organized control scheme is superior to that of other control schemes. However, since sensor networks face a wider range of perturbations, the purpose of this paper is to demonstrate the advantages of self-organized control against perturbations different from those in our previous work. Furthermore, based on the results of the evaluation, we also pose interesting questions such as why and how self-organized control is robust. In [21], from the results of the comparison, we pointed out that the difference in the robustness is derived from the degree to which the comprehension of a given node about the state of the network depends on information from other nodes. This is the key to differentiating the degrees of robustness of those two control schemes. In this paper, we describe the details of each method which were not able to be described in [21]. Furthermore, we show the characteristic of self-organized control by distribution of the number of hop of routes, and present the difference in the robustness of each control method against bit error.

The remainder of the paper is organized as follows. In Section II, earlier approaches to self-organized control are reviewed. Sections III describes the mechanisms of centralized

control and self-organized control, respectively. Section IV presents the simulation results so as to compare the robustness of both control approaches. In Section V, we discusses what brings robustness to self-organized control on the basis of these results. The paper is concluded in Section VI and discusses the generalization of our conclusions.

## II. Related work

The principle of self-organization is developed in nature [8], and we can find it everywhere. Each component autonomously decides its next action on the basis of local information, and the microscopic simple actions of the components collectively provide structure and functionality at the macroscopic level without any centralized coordination [19]. Such self-organized behavior is disparate from the distributed paradigm where individual components act autonomously while sharing global information, and many researchers have tried to derive the advantageous properties of the self-organizing system in efforts to solve scalability, reliability, availability, and robustness problems. For example, Directed Diffusion [12] is a well-known self-organization paradigm for certain novel features, including reinforcement-based adaptation of the gradient to the empirically best path. It is also known to be robust against node failures. [9] is proposed to achieve good adaptability and scalability by endowing mobile agents with simple intelligence. Some researchers further this approach and incorporate the behavior of social insects into the agents. BiSNET [4], which was shown to have strong self-healing capability for false positive data in data gathering, are examples that were inspired by the foraging principles of honey bees, while [16][5][25] are inspired by the Ant colony metaheuristic and said to be robust against node mobility. ACE [6] is an emergent algorithm that forms clusters through three rounds of feedback between nodes. Using local information alone, it efficiently covers the network with only a small amount of overhead. Ant-based clustering [11][24][22] is also a clustering method, drawing its inspiration from the behavior of ant colonies, but it is applied for data analysis. In addition, the task allocation method proposed in [17] uses the concepts of the "division of labor" of ants to achieve higher coverage in sensor network.

## III. Centralized and Self-organized control schemes in sensor networks

We provide detailed explanation of our centralized and self-organized control schemes, which are the subjects of robustness evaluation in the present study. The operations of both control schemes are based on the premise that multiple sinks are deployed in their respective monitoring regions. Using this multi-sink configuration, both control schemes take a cluster-based approach, in which the same number of node clusters and sinks is formed, and individual sensor nodes transmit their sensed data to the sink located in their cluster (Figure 1).

### A. Centralized control

Younis *et al.* [23] proposed a data-gathering scheme for sensor networks that assumes the existence of multiple

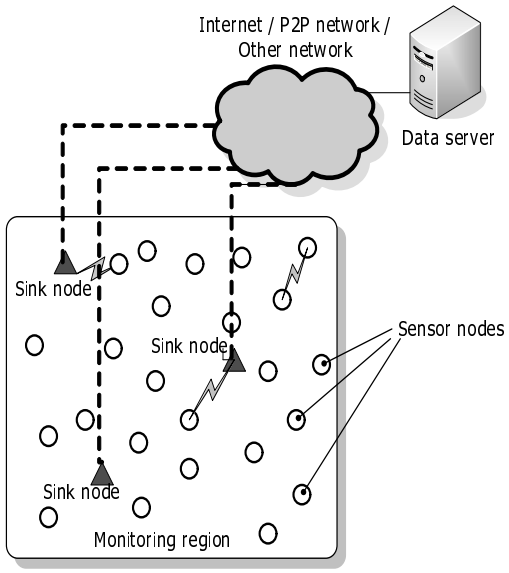



Fig. 1. Network model.

sinks (for consistency with the terminology used in our self-organized control [13], we use "sinks" here instead of the "gateway nodes" used in [23]). Sinks are significantly less energy-constrained than sensor nodes and the sensed data is gathered first in them. Sensor nodes are divided into the cluster which each sink manages, and the sinks calculate the route from each sensor node to themselves based on the residual power, state, etc. of a sensor node. They then tell their cluster members their previous- and next-hop nodes and the state they should stay in next (e.g., active or sleep state). In this data-gathering scheme the role of the clusters is almost same as that of the clusters in the scheme described in [13] — in both the cluster determines the eventual destination to which data packets are sent — so these two schemes are well-suited to be compared. Younis *et al.* [23], however, describe only the routing and node-state management and do not specify how the sensor nodes should be apportioned into clusters. In addition, some of its assumptions, for example, that each sink is located within the one-hop of all the sensor nodes in its cluster, are not appropriate for large-scale sensor networks. So we made some modifications to the proposed mechanism in order to make a convincing comparison.

We assume the existence of a control station, which is wired to all sinks. The station knows the initial power and locations of all nodes and sinks, and manages the overall network. Up-to-date residual power is reported periodically from sensor nodes, but the reporting packet is forwarded to the sink in a multi-hop fashion instead of direct communication. The station first divides the sensor nodes into as many clusters as there are sinks. The role of a cluster is to determine the destination sink for each sensor node, and we say that "sensor node $n_i$ belongs to cluster $S_j$" if $n_i$ transmits their sensing data to the destination sink $S_j$. The clustering method used is same as Voronoi tessellations using locations of sinks as basing points.

In other words, the central station splits the sensor nodes into clusters in such a way that each sensor node transmits packets to the nearest sink.

After clusters are determined, the station constructs routes for packets. As described in [23], the routes are determined by using Dijkstra's algorithm to minimize the total link cost. Link cost is assigned by the station beforehand to all the links between all node-and-node, node-and-sink pairs. Calculation of link cost is modified from [23] due to difference of assumptions, and the cost $C_{ij}$ of the link between node $n_i$ and $n_j$ is defined by residual power of the node and the distance between them:

$$C_{ij} = \begin{cases} \frac{E_{I_j}(4\pi)^2 d(n_i,n_j)^2}{E_{R_j}\lambda} & \text{if } d(n_i,n_j) \le \delta \\ \frac{E_{I_j} d(n_i,n_j)^4}{E_{R_j} h^4} & \text{if } \delta < d(n_i,n_j) \le r_{\max} \\ \infty & \text{if } r_{\max} < d(n_i,n_j) \end{cases} \quad (1)$$

where $E_{I_j}$ and $E_{R_j}$ are respectively the initial and residual powers of node $n_j$, $\lambda$ is the radio wavelength, $h$ is the height of the antenna, and $d(n_i, n_j)$ is the distance between nodes $n_i$ and $n_j$. The threshold value $\delta$ is a constant defined as $\delta = \frac{4\pi h^2}{\lambda}$, and $r_{\max}$ is the communication range of a sensor node.

After route construction is finished, the central station transmits the route information to sinks. For the sake of simplicity, packets which includes the route information are called "command packets" hereafter. The sink uses minimal transmission power when transmitting the command packet so that all the sensor nodes in its cluster can receive them. Command packet provides following information to sensor node $n_i$.

- Cluster to which $n_i$ belongs.
- The previous-hop node from which $n_i$ receives a packet and the next-hop node to which $n_i$ should transmit a packet.

The detection of node failure is based on a soft state model. Each sensor node transmits a hello message at a regular interval $t_{\text{hello}}$. On receiving the hello message from a neighboring sensor node $n_i$, sensor node $n_j$ registers entry of $n_i$ to its neighboring node table and interprets the reception as a sign that $n_i$ is working properly. Every time $n_j$ receives a hello message from $n_i$, expiry-time field in the entry is updated to the sum of $t_{\text{expire}}$ and the value of $n_j$'s internal timer. Only if $n_j$'s timer exceeds the value of expiry-time field, $n_i$ is deemed to have failed, and $n_j$ sends a failure-indication packet to its sink. This packet passes through the same route which the station calculated for data packets, and it reaches the sink. The sink passes the failure-indication packet to the station, which then recalculates new routes that circumvent the failed node. New routes are packed in a command packet and transmitted from the sink to sensor nodes.

Even when $n_i$ works normally, hello packets from $n_i$ might not arrive within the expiry time because of interference or transmission error. This possibility must be allowed for, because the accumulation of such false positives would cause a virtual connectivity problem limiting network performance. Preparing for such a false detection, node $n_j$ memorizes an ID of the failed node when detecting the failure. And if $n_j$ could receive a hello packet from $n_i$, it deems the detection of $n_i$'s failure to have been false positive, and transmits a failure-recovery packet to inform the station about that. The sink relays it to the station, and the station recomputes new routes and distributes them to sensor nodes.

In this centralized control, sink-failure can be easily detected because of the assumption that sinks and the central station are linked with wire. By keeping track of sinks' status, the station can recompute clusters and routes just after the sink failure. It does not need to take explicit measures, and all it has to do is to transmit a command packet containing new cluster organization and route information as usual. Reliable communication can be readily provided in wired networks. So we ignore the possibility of false detection of sink failure.

### B. Self-organized control

We have proposed a bio-inspired control which shows a self-organized property [13]. Our self-organized control approach is based on pheromone-mediated ant-swarm behaviors called ant colony optimization (ACO) [3] and ant-based clustering [11][24][22]. Sensor nodes are divided into as many clusters as there are sinks by using ant-based clustering with a virtual "cluster pheromone," and routing is performed in each cluster by using "routing pheromone." The detailed operation for our proposal is given in the following.

ACO is a probabilistic approach inspired by ants in their foraging activity to combinatorial optimization problems like the traveling salesman problem [7]. Ants follow efficient routes to their food by being attracted to higher concentrations of pheromones left by other ants. An ant will leave a volatile pheromone trail while carrying food back to the nest. If another ant finds the trail before it dissipates, that ant will follow it to the food and it too will leave pheromone on the way back, reinforcing the trail. If there is enough food that several workers can bring food back to the nest, a high pheromone concentration will be maintained and even more ants will be attracted. As the food supply becomes smaller, fewer ants will be attracted and the trail will gradually disappear as the pheromone evaporates. This positive-feedback trail building is the basic idea behind the ACO approach, and ACO has been applied to some of the routing problems.

We have also applied the principle of ACO to hop-by-hop routing in our proposed scheme. Each sensor node has a pheromone table, and the advantages of neighbors as a next-hop node are stored in the form of routing pheromones. When a sensor node transmits a packet to notify the sink of obtained data, it refers to its pheromone table, and stochastically selects the next-hop node leading to the sink based on the routing-pheromone value. Thus, each sensor nodes selects a next-hop node with greater probability of having more routing pheromones (sensor node with more routing pheromones means preferable next-hop node). Furthermore, if some neighboring nodes have almost the same routing-

pheromone value, they are selected as next-hop nodes with almost the same frequency, and the number of packets that must be relayed is distributed among them.

An important problem of applying ACO to routing is how to determine which route should have higher routing-pheromone value, in other words, how to define what are the "preferable routes" in a given network. We define good routes in sensor networks as follows:

- routes with a small hop count on the way to a sink.
- routes that go through sensor nodes with high residual power.

It is not necessary for each node to send packets (ants) in order to find good paths to the destination as some ant-based routing employed [5][25][2]. Such strategies could cause unnecessary power consumption and needlessly occupy wireless channels, because of ants traveling back and forth over the network. Thus, we chose sinks to flood the ants, which we call backward ants. Backward ants do not go back into the sink. As we previously pointed out, the required next-hop node is a sensor node located nearer to the sink, which has enough residual power. With that in mind, the role of backward ants is to establish a routing-pheromone distribution in which the required next-hop node has a higher routing-pheromone value. Let us introduce following terms to simplify our explanation of routing.

| | |
|---|---|
| $n_i$: | ID of sensor node. |
| $S_k$: | ID of sink. At the same time, $S_k$ also represents ID of cluster to which sink $S_k$ is dedicated. |
| $S_{n_i}$: | ID of sink that $n_i$ belongs to. |
| $Pb_{S_k}(n_i)$: | Routing-pheromone value that $n_i$ assigns to backward ant, which is transmitted by $S_k$. |
| $P_{n_i}(n_i)$ | Routing-pheromone value for $n_i$ to declare as its own pheromone. |
| $P_{n_i}(n_j, S_k)$: | Routing-pheromone value stored in $n_i$'s pheromone table that represents benefits of $n_j$ as next-hop node to transmit packet to $S_k$. |
| $C_{n_i}(S_k)$: | Cluster pheromone of $S_k$ estimated by $n_i$. |

A sink $S_a$ broadcasts backward ant $B$ with maximum routing-pheromone value $Pb_{S_a}(S_a) = P_{\max}$. On receiving $B$, sensor node $n_i$ stores routing pheromone carried by the backward ant ($Pb_{S_a}(S_a)$), its source node ($S_a$), and sensor node which relays $B$ immediately before ($S_a$) as an entry, in its own pheromone table. Thus, $n_i$ memorizes that the benefit of selecting $S_a$ as a next-hop node for transmitting packets to $S_a$ is $P_{\max}$. After that, $n_i$ relays $B$, making $B$ carry a new routing-pheromone value. This new routing-pheromone value $Pb_{n_i}(S_a)$ is calculated according to:

$$Pb_{n_i}(S_a) = \alpha\left(1 - \exp\left(-\beta\frac{E_{R_i}}{E_{I_i}}\right)\right) Pb_{S_a}(S_a) \qquad 0 < \alpha < 1, \beta > 0 \quad (2)$$

After receiving $B$, which is relayed by $n_i$, $n_j$ creates a new entry $(n_i, S_a, Pb_{n_i}(S_a))$ as in the case of $n_i$. Then, $n_j$ calculates a new routing-pheromone value according to Eq. (2), and forwards $B$ with a new routing-pheromone again. A good pheromone distribution emerges through frequent repetitions of these behaviors.

Sensor nodes periodically communicate using a hello message like that in the centralized control described in Sect. III-A. But the purpose of this hello message is not only to provide a countermeasure to node failures but also to comprehend the situation of surrounding area. The hello message transmitted from $n_i$ conveys routing-pheromone value of $n_i$ itself ($p_{n_i}$), cluster ID to which $n_i$ belongs to ($S_{n_i}$), and cluster pheromone of $S_{n_i}$ evaluated by $n_i$ ($C_{n_i}(S_{n_i})$), which is described in detail later in this section. $p_{n_i}$ is the mean routing-pheromone value for all entries in $n_i$'s routing table. After receiving the hello message, $n_j$ updates the routing-pheromone value for the $n_i$'s entry in $n_j$'s routing table following Eq. (3) with $\gamma \in [0, 1]$.

$$P_{n_j}(n_i, S_{n_j}) = \gamma P_{n_j}(n_i, S_{n_i}) + (1 - \gamma) p_{n_i} \quad (3)$$

A sensor node chooses its next-hop node stochastically using the routing-pheromone distribution, and relays packets to it. Assuming $N_{n_i}$ is a set of neighboring nodes for $n_i$, which is equivalent to candidate set of next-hop nodes, the probability of $n_i$ selecting $n_j$ as its next-hop node is represented as:

$$p_{n_i}(n_j) = \frac{P_{n_i}(n_j, S_{n_i})^2}{\sum_{k \in N_{n_i}} P_{n_i}(k, S_{n_i})^2} \quad (4)$$

This form of equation is used in some propositions using the ACO approach, e.g., [5]. Routing loop can be constructed due to the probabilistic approach, but discarding the looped packets made the data collection unreliable in our simulations. So now we avoid routing loops by appending node IDs the packet went through to the header. Sensor nodes listed in the header are excluded from the set of candidate for next-hop node. This requires only a small amount of communications overhead.

How to select a destination sink still remains a question in multi-sink sensor networks. Our clustering method, ant-based clustering, is also inspired by a swarm behavior of ants. Ant-based clustering was originally a method of swarm intelligence by ants grouping eggs or larvae according to their size. Ants repeatedly pick up and drop larvae based on their degree of similarity with neighbor eggs while wandering around. In such a behavior, larvae which differ substantially from their neighbors in size move toward similar-sized ones, and clusters of different-sized larvae emerge in a self-organized way. We substitute similarity with the advantage of belonging to a cluster, and do clustering to suit the network situation.

Each node calculates a cluster-pheromone value based on the routing-pheromone values, and uses them to determine which cluster it should belong to. Cluster $S_{n_i}$'s cluster pheromone evaluated by $n_i$ is defined as:

$$C_{n_i}(S_{n_i}) = \frac{\sum_{k \in \mathrm{blng}_{n_i}(S_{n_i})} C_k(S_{n_i}) + avg\_ph_{n_i}(S_{n_i})}{|blng_{n_i}(S_{n_i})| + 1} \quad (5)$$

where $\text{blng}_{n_i}(S_{n_i})$ represents a set of neighboring nodes of $n_i$ that participate in cluster $S_{n_i}$. This information can be recognized via hello messages, which has the cluster ID of the sender. The term $\text{avg\_ph}_{n_i}(S_{n_i})$ is the mean of routing-pheromone values in entries having destination sink $S_{n_i}$:

$$avg\_ph_{n_i}(S_{n_i}) = \frac{\sum_{k \in \text{blng}_{n_i}(S_{n_i})} P_{n_i}(k, S_{n_i})}{|\text{blng}_{n_i}(S_{n_i})|} \quad (6)$$

Cluster-pheromone value is conveyed in hello packets, so each sensor node can acquire the cluster-pheromone values of neighboring clusters. Sensor nodes regard a cluster with a higher cluster-pheromone value as a good cluster to join, and stochastically switch to it. The probability of $n_i$ changing its cluster from $S_j$ to $S_k$ is

$$P_{n_i}(S_j \to S_k) = \left( \frac{f_{n_i}(S_j, S_k)}{k_{th} + f_{n_i}(S_j, S_k)} \right)^2 \quad (7)$$

where $k_{\text{th}}$ is a constant value used to control the probability and where $f_{n_i}(S_j, S_k)$ is calculated as follows:

$$f_{n_i}(S_j, S_k) = \max\left(0, \frac{|\text{blng}_{n_i}(S_k)|}{N_{n_i}} \frac{C_{n_i}(S_k) - C_{n_i}(S_j)}{C_{n_i}(S_k)}\right) \quad (8)$$

The detection of node failures is exactly equivalent to that of centralized control described in Sect. III-A. After $t_{\text{expire}}$ passes without receiving hello packets from sensor node $n_j$, neighboring node $n_i$ detects that $n_j$ has failed. By deleting the entry for $n_j$ in its pheromone table, $n_i$ selects appropriate next-hop nodes according to Eq. (4) without any special handling.

Detecting sink failure was also based on the same soft-state model. That is, the sink periodically broadcast hello message as well as sensor nodes. Sensor nodes around the sink determine that the sink has failed if they had not received hello message from the sink for $3 \times t_{\text{expire}}$. The cluster in sink failure is no longer preferable. Thus, sensor nodes set cluster-pheromone values of all the entries stored in their neighbors table to 0 and abandon their membership. As hello messages indicating the sink failure propagated over the network after that, sensor nodes participating in the failed sink's cluster also abandoned their membership, and joined other clusters.

## IV. Evaluation and discussion

We try comparison of our self-organized and centralized controls by simulation experiments. First, we explain the simulation model which experimented, then we evaluate the robustness against various perturbations.

### A. Simulation Environment

We implemented our self-organized and centralized controls on ns-2 network simulator [1]. In the following experiments, we randomly placed 300 sensor nodes over a region monitoring a square, 100 m per side, unless otherwise stated. We assumed there were four sinks at $(25, 25)$, $(75, 25)$, $(25, 75)$, $(75, 75)$, respectively. We tested other sink positions, and obtained almost the same results.

TABLE I
Sensor node parameters

| | |
|---|---|
| Transmission power | 0 dB |
| Communication range | 10 m |
| Frequency | 2,450 MHz |
| Bit rate | 250 kbps |
| Height of antenna | 20 cm |
| Initial power | 25 J |
| Power consumption in transmission state | 40.95 mW |
| Power consumption in receiving state | 45.78 mW |

TABLE II
Simulation parameters

| | |
|---|---|
| $t_{\text{hello}}$ | 1 s |
| $t_{\text{expire}}$ | 5 s |
| $P_{\text{max}}$ | 10 |
| $\alpha$ | 0.7 |
| $\beta$ | 7 |
| $\gamma$ | 0.875 |
| $k_{\text{th}}$ | 0.5 |
| Size of a hello packet | 10 bytes |
| Size of a failure detection packet | 10 bytes |
| Size of a failure recovery packet | 10 bytes |
| Size of a data packet | 64 bytes |

We used the two-ray ground reflection model [1] as the radio propagation model, and the MAC and PHY layers follow the IEEE 802.15.4 specification. In the simulation of the centralized control, the size of command packet can easily exceeded the value specified in IEEE 802.15.4. We therefore virtually set *aMaxPHYPacketSize*, which determines the maximum length of a packet, to infinity. The size of the command packet transmitted from sink $S_j$ is calculated using the following equation:

$$\sum_i 6 \cdot e_{n_i} \cdot num_{S_j} + 7 \quad (9)$$

where $e_{n_i}$ is the number of previous- and next-hop node pairs assigned to node $n_i$ and $num_{S_j}$ is the number of sensor nodes in cluster $S_j$. We assume that 6 bytes are enough for the pair, and that 7 bytes are enough for a header. We set the parameters of sensor nodes (listed in Table. I) by referring to [18]. The simulation parameters are also listed in Table. II. We do not consider FEC to take particular note of the effect of transmission error, therefore the packet is discarded even if one bit error occurs.

In the following data-collection model we used, sensor nodes send the information they obtain to their sinks in a multi-hop way at a predefined interval $t_{\text{intval}} = 10$ s. Sensor nodes do not synchronized with each other, and the transmission time of it is independent of that of the others. One of the most important metrics for sensor networks is the reliability of which information is brought to a sink. We therefore defined a metric we call the data-collection rate. When the number of sensor nodes that work properly is $N_{\text{act}}$, the number of data packets generated in $t_{\text{intval}}$ is of course $N_{\text{act}}$. When the number of packets that reach one of the sinks is $r$, the data-collection rate is defined as $r/N_{\text{act}}$.

In the centralized control, the parameter with the greatest influence on the data-collection rate is command-packet trans-

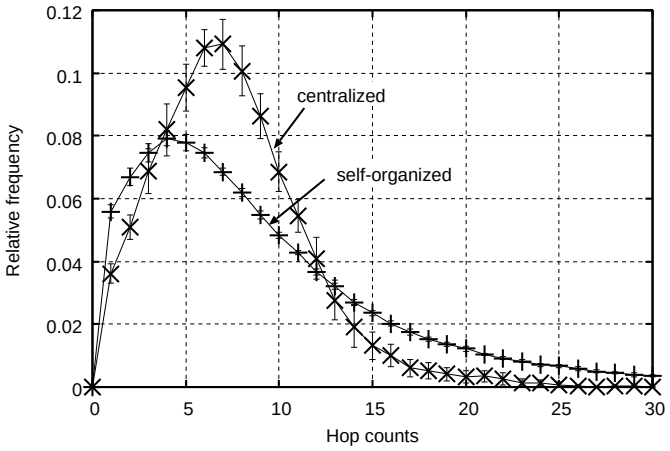



Fig. 2. Efficiency of routes generated by centralized control and of routes generated by self-organized control.

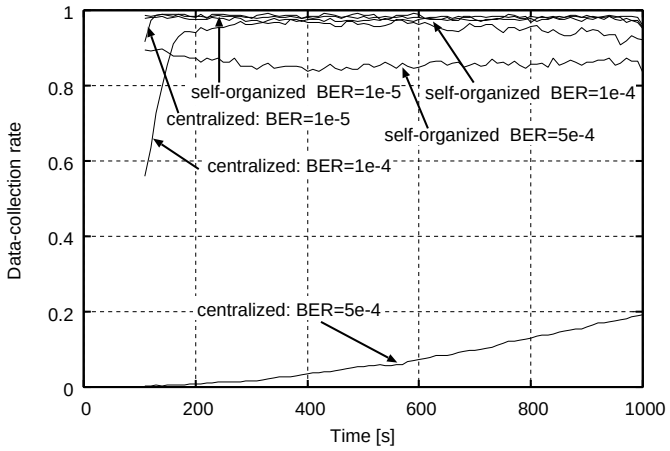



Fig. 3. Influence of BER on data-collection rate.

mission interval. If this interval is too long, sensor nodes will only slowly find out what it should do next, especially when the command packets are frequently lost. And if the interval is too short, command packets coming one after another result in severe interference problems. We conducted simulation experiments to find out whether 1 s, 10 s, 100 s, or 500 s would be the best interval and chose 10 s as the one yielding the best balance between data-collection rate and power consumption. Not only in centralized control, transmission interval of backward ants in our self-organized control also has great influence. Too short an interval causes repeated interference and too long an interval does not construct pheromone distribution enough for data gathering. We simulated transmission intervals of 10 s, 100 s, and 500 s and selected 100 s.

In the simulation experiments, each sink transmits command packets or backward ants until at least 100 s pass over. So we consider the network is in the transient state for 100 s from the start, and do not plot a graph in the time window.

### *B. Instability of Generated Routes*

We first compared the efficiency, in terms of hop counts, of the routes generated using centralized control and self-organized control. The hop counts reported here are mean values of all routes between each sensor node and its sink. The distribution of hop count is shown in Figure 2 with 95% confidence intervals. This graph is for the idealized scenario in which no node failures occur, and bit error rate is set to $10^{-5}$. Changing BER did not generate significant influence. Actually, there is only a little difference in their distribution as shown in Figure 2, and the same is true for their mean values as shown in Table. III, where statistics values of the routes are listed. However, variance of both control approaches differs substantially. These interesting results suggest that quality of generated routes can fluctuate widely, i.e., low predictability and controllability, in self-organized control. A sensor node in self-organized control decides its own action on the basis of limited, local information. Their lack of global viewpoint leads to difficulty in finding global optimum, and results in wide fluctuation.

### *C. Measures against Transmission Error*

We conducted simulation experiments to study the robustness of both control approaches against transmission error under the assumption that no node failures occur. In Figure 3, both kinds of control show about the same data-collection rate with BER $= 10^{-5}$, but that of centralized control becomes slow to rise up along with the increase in BER.

In the centralized control, the tremendous amount of information is gathered to the central station to decide a course of actions for each sensor node, and the station issues the instructions to sensor nodes. Sensor nodes completely rely on the control information from the station, and the station believes sensor nodes follow the order. With this strong dependency, what will happen when the information is beyond some sensors' reach? This situation just arises due to transmission error in this simulation experiment. In the case where some sensors can receive the instruction and others cannot, inconsistent views of the routes can be introduced among them. Such inconsistency makes sensor nodes lose their next-hop node for a received packet, and the network gets stuck in the pathological state until their views get consistent. Actually, data-collection rate increases with time in Figure 3, but this is because frequently transmitted command packets (i.e., with an interval of only 10 s) compensate discarded ones. This slow ascent means that the network does not adapt well when the route changes for any reason.

In the self-organized control, sensor nodes are not able to know global information of the network, leading to easily have inconsistent information among them. But the adverse effects of their inconsistency are localized around them, because they have its own knowledge base based on their limited view, instead of sharing global information. That results in the good robustness against transmission error as shown in Figure 3.

Differences in the behaviors of the two kinds of control also appear in Figure 4, where mean of the data-collection rate

TABLE III
STATISTICS OF ROUTES GENERATED IN CENTRALIZED CONTROL AND SELF-ORGANIZED CONTROL. 95% CONFIDENCE INTERVALS ARE ALSO SHOWN.

| | Centralized control | Self-organized control |
|---|---|---|
| Average hop count | $7.47 \pm 0.36$ | $9.08 \pm 0.34$ |
| Average delay | $0.156 \pm 8.62 \times 10^{-3}$ | $0.226 \pm 1.56 \times 10^{-2}$ |

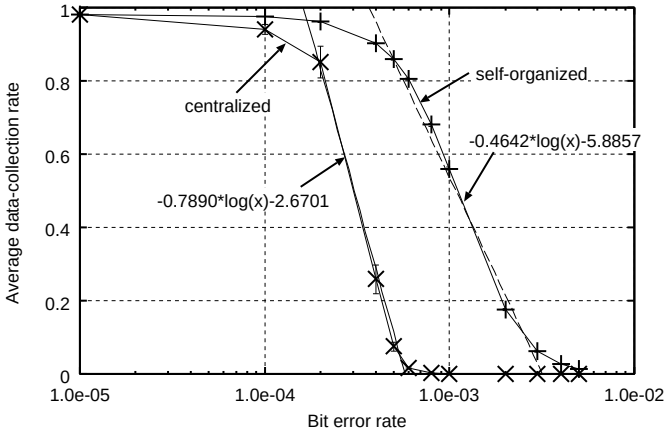



Fig. 4. Data-collection rate versus BER.

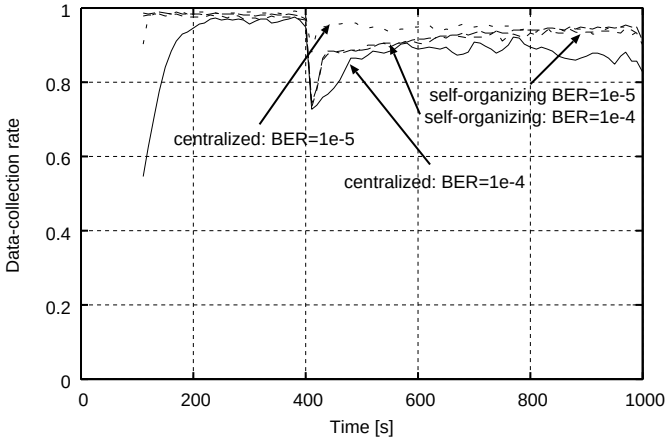



Fig. 5. Features of the process of recovery from sink failure.

are plotted against BER. Logarithmic approximation lines for their decays are also shown. Self-organized control keeps data-collection rate above 80% about 3 times longer. In addition, the gradient of the self-organized control is only 58% of the centralized control. When the gradient is steep, the network function might deteriorate markedly in response to even a small change of BER. When the gradient is gentle, however, data collection is not affected significantly if the BER changes. For that reason, self-organized control is more robust against transmission error. As mentioned above, such robustness of the self-organized control originates the fact that each node operates based on only its local information. On the other hand, the lack of the correspondence of routing information by the packet loss causes the performance deterioration at the centralized control.

### *D. Measures against sink failure*

Figure 5 presents the results for the case in which a sink located at $(25, 25)$ fails at 400 s. After the sink failure, the data collection rate drops sharply to about 75%, except in the case of centralized control with $10^{-5}$ BER (Bit Error Rate), where the rate drops to only 90%. A rate of 75% means that one cluster suffered catastrophic damage (the ratio of data packets gathered within a cluster is about 25%). Not only is the drop in the data collection rate in the case of centralized control and low BER small, but also the recovery is almost immediate. The control station which is wired to the sinks becomes aware of the failure within a short amount of time (in our simulations, it is set to 0 s), after which the clusters are reconstructed and the routes are recomputed upon receiving the command packet, in order to adapt the whole network to the failure. Sensor nodes immediately modify their cluster membership and routing table according to the instructions contained in the command packet, and the data collection rate is restored soon after that. Indeed, in cases where the channel quality is poor, the data collection rate in the centralized control scheme is unable to recover within the simulation time shown in Figure 5, since centralized control is weak with respect to transmission errors, as indicated in [14].

In contrast to the centralized control scheme, the self-organized control scheme needs more time for the distant sensor nodes to adapt to the sink failure. In addition, since the network has no supervisor and no explicit instructions, some nodes might be prone to taking contradicting actions based on the possibility of receiving inaccurate information about the condition of the network. For these reasons, in low BER environments, the self-organized control scheme exhibits worse recovery than the centralized one. In high BER environments, however, the relationship between self-organized control and centralized control is reversed, since the self-organized control scheme inherently does not have critically important information whose loss can bring serious and adverse influence to the network.

### *E. Measures against node failure*

We already demonstrated the robustness against node failure in our previous work [14]. Moreover, we showed that although most of the sensor nodes other than the failed ones exhibit data collection rates of about 100% in the self-organized control scheme, failures in the case of the centralized control scheme have considerable influence on the data collection rates at the cluster level, where many sensor nodes are unable to transmit packets to their sinks, and this influence is especially notable when concentrated and simultaneous failures occur. However, when we tested random failures in a 100 m $\times$100 m

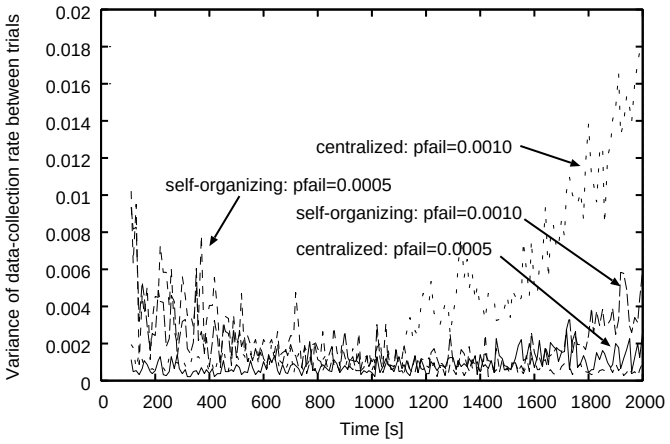
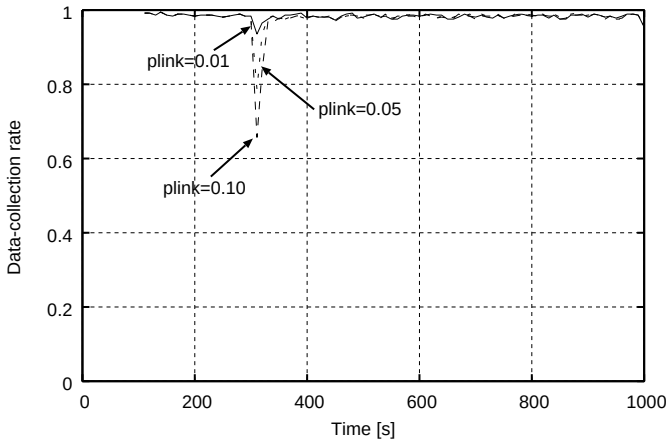
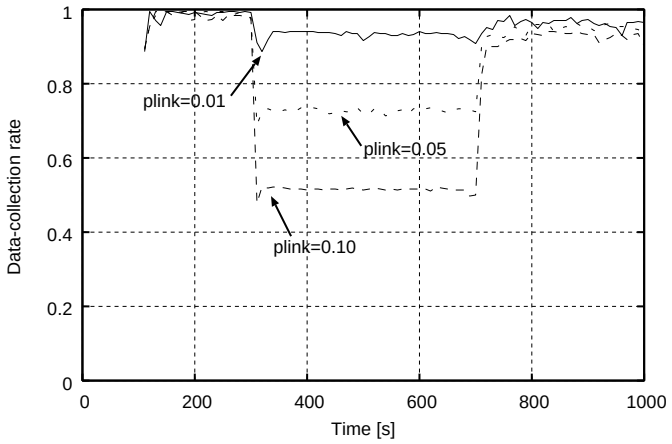



Fig. 6. Variances of the data collection rates among trials.




(a) Self-organized control.




(b) Centralized control.

Fig. 7. Influence of link disconnections on the data collection rate.

monitoring region containing 300 nodes, the difference in the robustness of the self-organized and the centralized control schemes was not clear due to the connectivity degradation caused by the continual node failures. Therefore, here we temporarily used a narrower monitoring region of 50 m $\times$ 50 m while keeping the number of nodes and sinks, and defined $p_{\text{fail}}$ as the failure rate per second for each sensor node.

The variances of the data collection rates of both control schemes among trials are shown in Figure 6. The variance in the self-organized control scheme is small and not as sensitive to the failure rates. However, in the centralized control scheme, the data collection rates in some trials experience sudden drops, which lead to the higher variance of the data collection rates, as shown in Figure 6. The high variance in the case of centralized control indicates the difficulty of predicting the data gathering capability in harsh environments, although all of the plots are prepared using the same parameters.

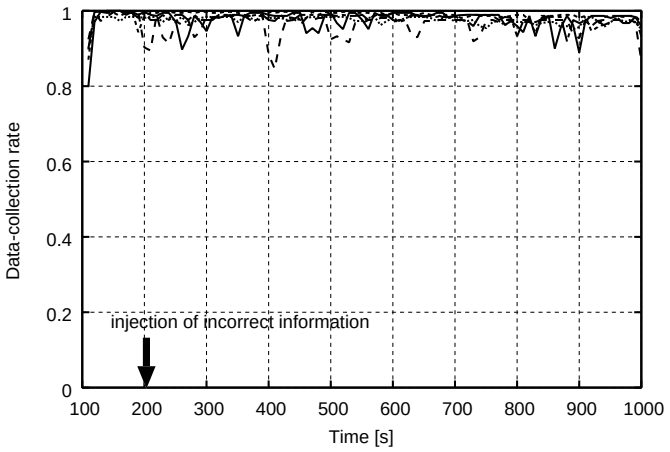
### *F. Measures against link disconnection*

As links can become disconnected intermittently in wireless networks, in the case where the link between nodes $n_i$ and $n_j$ is disconnected but the link between $n_i$ and $n_k$ is still connected, there is a possibility that the status of $n_i$ as seen from the perspective of $n_j$ and $n_k$ is inconsistent. Therefore, in order to study the differences in the robustness of the two schemes, we randomly disconnected a percentage of the links. We assume that each node is linked to an arbitrary neighboring node, and each link is disconnected with probability $p_{\text{link}}$ in both directions. This disconnection process was conducted for all nodes, and the duration of the disconnection was 400 s, from $t$=300 s to $t$=700 s.

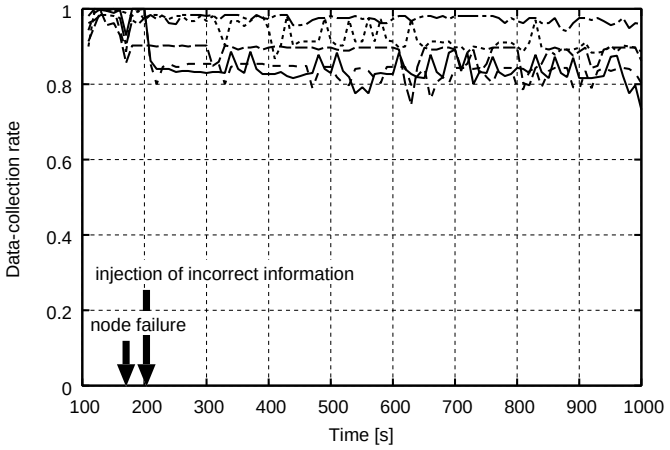
In the results shown in Figure 7, the data collection rate in the self-organized control scheme immediately recovers to the rate before the disconnection, although it experiences a declination for a short amount of time. The centralized control scheme, on the other hand, suffers greatly from the disconnections, where detection of massive node failures occurs since neighboring nodes regard disconnected nodes as failed due to their inability to transmit hello messages. In other words, sensor nodes cannot distinguish failures from link disconnections in our centralized control scheme. Furthermore, after the detection of a missing link, the neighboring nodes transmit failure-indication packets, which are in fact false-positive detection packets, to the control station. As a result, the control station does not provide routes to the node which is considered as failed, and the packets from the disconnected node are discarded, which is the main reason for the decay of the data detection rate in Figure 7(b).

## V. Dependence on control information

Next, we consider the factor which affects the difference in robustness and perform the evaluation by additional simulations.

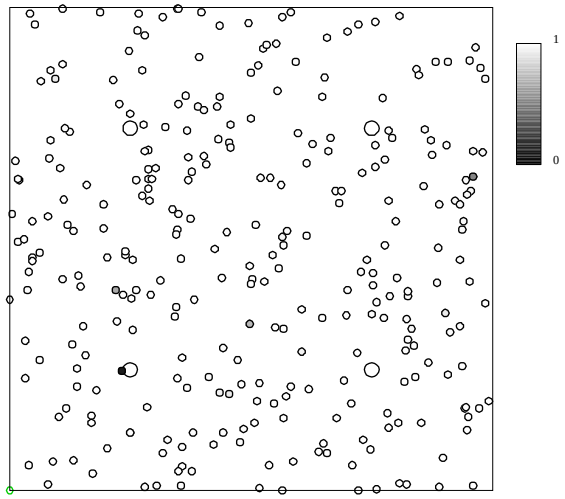



(a) False-positive failure detection.

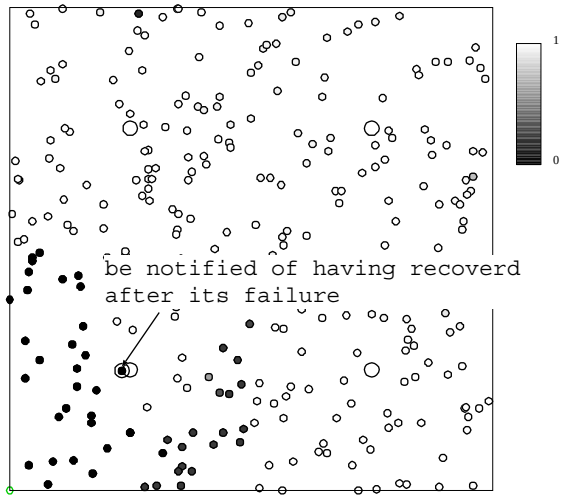



(b) False-recovery indication.

Fig. 8. Results of injecting incorrect information.



(a) From $t$=160 s to $t$=200 s




(b) From $t$=200 s to $t$=1000 s

Fig. 9. State of the network after injecting false-recovery information.

### A. Factors influencing the difference in robustness

In the evaluation presented in Section IV and in previous works, there was a significant difference between the robustness of self-organized control and centralized control. We are inclined to explain this trend in terms of "dependence on control information". In this case, "dependence" has almost the same meaning as that used in fault management. The dependence is a relation in which an error or failure in an object may cause an error or failure in another object. We define control information as the information exchanged between entities of a given network which coordinates their joint operation.

In Sections IV-E and IV-F, even the control station itself did not comprehend the correct state of the network. This is caused by the fact that the control station also depends on control information received from the nodes in the network. The control station constructs a precise view of the whole network by integrating each piece of information about the state of the network. In other words, the problem of the dependence is that the control information from potentially unreliable nodes in environments where reliable communication is not guaranteed plays a critical role in generating the control scheme at the control station. In Section IV-E, failure indication packets, which notify the command node about the correct state of the network, did not reach the control station, resulting in a sudden drop of the data collection capability of the clusters. In Section IV-F, one node considers a neighboring node to be operating correctly, while another node considers the same neighboring node as faulty, resulting in the transmission of failure indication packets even though no nodes have failed.

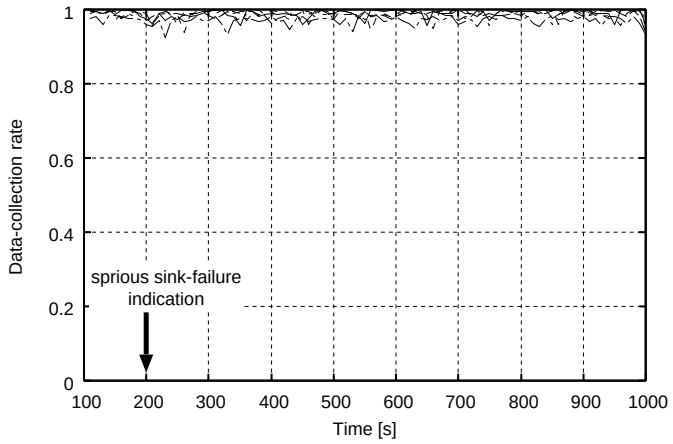



Fig. 10. Influence of erroneous sink failure indication.

In this way, information which does not reflect the correct state of the network brings vulnerability to the centralized control scheme. However, since optimization of the whole network cannot be performed if the dependency on the control information is reduced, general network performance like delay or a throughput degrades. For example, as shown in Figure 2, in self-organization control, distribution of the number of hop from a node to the sink is large, and there are nodes to whom delay becomes large very much.

Of course, at the node level, self-organized control is identical to centralized control, meaning that individual nodes potentially have an erroneous understanding about the state of the network. However, individual nodes affect only their surrounding environment or neighboring nodes since all nodes have only partial view of the network, and do not transmit or receive explicit control information. Due to this behavior, the influence of individual nodes on the global state of the network is much smaller than in the centralized control scheme. In this regard, since we have not yet clarified the influence of erroneous information received from individual nodes, in the next section we verify our idea by deliberately injecting incorrect information into the network.

### *B. Influence of incorrect information*

The purpose of this demonstration is to determine how strong the influence of information received from individual nodes is, as well as how potentially unreliable nodes affect the behavior of the whole network. Therefore, in this section, we deliberately inject spurious information in order to show unambiguously the influence of information received from individual nodes on the functionality of the network. At first, in the centralized control scheme, we considered two scenarios: 1) we injected false-positive failure detection packets, which convey the misinformation that a properly working node is detected as failed, and 2) false-recovery packets, which inform the surrounding nodes that a node which has failed is detected as recovered.

Although we deliberately injected incorrect information at $t$=200 s that the node nearest to the coordinate $(25, 25)$ had failed, there was no fluctuation or drop in the data collection rate due to the injection, as seen from the results shown in Figure 8(a). In fact, the node which was wrongly detected as failed was not able to send its packets to the sink as the control station did not consider the failed node as a member of the data collection cluster. However, routing information was supplied to the other sensor nodes correctly, and thus the influence of the erroneous information was limited.

Next, we tested the scenario where incorrect information about the recovery of a node is injected into the network. At first, we made the node nearest to the coordinate $(25, 25)$ fail at $t$=160 s, followed by the injection of information that the node has recovered at $t$=200 s. Figure 8(b) shows the results of five trials, and it is clear that the behavior of the data collection rates are different among them, i.e., they are different depending on the node deployment. There is a clear drop in two of the plotted lines just after the injection of erroneous information at $t$=200 s. Given this factor, focusing on one of those lines, in Figure 9 we visualized the data collection rate of the individual nodes from the time when node fails ($t$=160 s) until the injection of misinformation ($t$=200 s), and from the injection ($t$=200 s) to the end of the simulation ($t$=1000 s), respectively. As shown in Figure 9(a), the influence of the node failure can be limited. However, after the injection, data collection in the larger part of the respective cluster becomes impossible.

Self-organized control does not have any means for explicit indication of failure or failure recovery. Therefore, it was impossible to compare it directly with the centralized control in terms of the influence of erroneous information. Instead, we used the indication of sink failure, which is a message which explicitly conveys information about the failure of a sink to the neighboring nodes by using a hello message. Furthermore, we made the sensor node nearest to the coordinate $(25, 25)$ transmit the information about the sink failure. This indication is spread over the respective cluster through forwarding by nodes which receive the indication.

As a result, although spurious sink failure indication was injected into the network at $t = 200$ s, there was no clear difference in the data collection rate before and after the injection, as seen from the data collection rates from five trials presented in Figure 10. In our self-organized control scheme, sensor nodes invalidate their membership to the respective cluster upon receiving the sink failure indication, and negative influence was expected due to the dynamic change of cluster membership. However, contrary to our expectation, the cluster memberships were restored to those before the injection. In other words, correct information from other nodes naturally adjusts the situation caused by erroneous information, and this fact contributes to the robustness of self-organized control.

## VI. Conclusion

In spite of growing interest, there are many points regarding self-organization which remain insufficiently understood. In this paper, we studied the robustness of self-organized

control against a wide range of perturbations by comparing it with centralized control, and we attempted to answer some important questions. One such question is whether self-organized control is in fact robust, and we quantitatively demonstrated the affirmative answer by examining various scenarios. Although this result is not surprising, it was found that self-organized control has the obvious benefit of superior robustness, especially if applied to systems in dynamically changing environments, although at the cost of reduced system predictability. Furthermore, the questions about why self-organized control is robust and what factors determine the robustness of self-organized control were also addressed, and based on the results obtained from the simulation experiments, we arrived at the conclusion that the dependence on the control information in the system plays a critical role in determining whether or not the robustness is sufficient. In a network which is composed of potentially unreliable nodes and is located in a harsh environment, decreasing the dependence on the control information received from the nodes is critical to yielding sufficient robustness, and self-organized control inherently possesses such properties.

## Acknowledgements

This research was partially supported by the "Global COE (Center of Excellence) Program" and the Grant-in Aid for Scientific Research (C) 19500060 of the Ministry of Education, Culture, Sports, Science and Technology, Japan.

## References

[1] ns-2 – the network simulator. online available at http://www.isi.edu/nsnam/ns.

[2] B. Barń and R. Sosa. AntNet: Routing algorithm for data networks based on mobile agents. *Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial*, 12:75–84, 2001.

[3] E. Bonabeau, G. Theraulaz, and M. Dorigo. *Swarm Intelligence: From Natural to Artificial Systems (Santa Fe Institute Studies in the Sciences of Complexity Proceedings)*. Oxford Univ Pr, Oct. 1999.

[4] P. Boonma, P. Champrasert, and J. Suzuki. BiSNET: A biologically-inspired architecture for wireless sensor networks. In *Proceedings of The Second IARIA International Conference on Autonomic and Autonomous Systems*, Published by the IEEE Computer Press, July 2006.

[5] G. D. Caro, F. Ducatelle, and L. M. Gambardella. AntHocNet: An ant-based hybrid routing algorithm for mobile ad hoc networks. *European Transactions on Telecommunications*, 16:443–455, Oct. 2005.

[6] H. Chan and A. Perrig. ACE: An emergent algorithm for highly uniform cluster formation. In *Proceedings of the First European Workshop on Wireless Sensor Networks*, pages 154–171, Jan. 2004.

[7] M. Dorigo, V. Maniezzo, and A. Colorni. The ant system: Optimization by a colony of cooperating agents. *IEEE Trans. Systems, Man, and Cybernetics*, 26(2):29–41, 1996.

[8] F. Dressler. Self-organization in ad hoc networks: Overview and classification. Technical report, University of Erlangen, Department of Computer Science 7, Mar. 2006.

[9] L. Gan, J. Liu, and X. Jin. Agent-based, energy efficient routing in sensor networks. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 472–479, Aug. 2004.

[10] C. Gershenson and F. Heylighen. When can we call a system self-organizing? In *Proc. 7th European Conference on Advances in Artificial Life*, pages 604–614, Sept. 2003.

[11] J. Handl, J. Knowles, and M. Dorigo. Strategies for the increased robustness of ant-based clustering. *Engineering Self-Organising Systems: Nature-Inspired Approaches to Software Engineering*, 2977:90–104, 2004.

[12] C. Intanagonwiwat, R. Govindan, and D. Estrin. Directed diffusion: A scalable and robust communication paradigm for sensor networks. In *Proceedings of the Sixth Annual International Conference on Mobile Computing and Networks*, pages 56–67, Aug. 2000.

[13] Y. Kiri, M. Sugano, and M. Murata. Self-organized data-gathering scheme for multi-sink sensor networks inspired by swarm intelligence. In *Proc. 1st IEEE Intl. Conf. on Self-Adaptive and Self-Organizing Systems*, July 2007.

[14] Y. Kiri, M. Sugano, and M. Murata. Robustness differences between bio-inspired control and centralized control. In *Proc. of Biological Approaches for Engineering Conference*, Mar. 2008.

[15] H. Kitano. Biological robustness. *Nature Review Genetics*, 5(11):826–837, Nov. 2004.

[16] Z. Liu, M. Z. Kwiatkowska, and C. Constantinou. A biologically inspired optimization to AODV routing protocol. In *Proceedings of the 3rd Workshop on the Internet, Telecommunications and Signal Processing*, pages 106–111, Dec. 2004.

[17] K. H. Low, W. K. Leow, and J. Marcelo H. Ang. Task allocation via self-organizing swarm coalitions in distributed mobile sensor network. In *Proceedings of 19th National Conference on Artificial Intelligence*, pages 28–33, July 2004.

[18] Moteiv Corporation. *Telos (Rev B): PRELIMINARY Datasheet*, May 2004.

[19] C. Prehofer and C. Bettstetter. Self-organization in communication networks: Principles and design paradigms. *IEEE Communications Magazine, Feature Topic on Advances in Self-Organizing Networks*, 43(7):78–85, July 2005.

[20] T. D. Seeley. When is self-organization used in biological systems? *Biological Bulletin*, 202:314–318, June 2002.

[21] M. Sugano, Y. Kiri, and M. Murata. Differences in robustness of self-organized control and centralized control in sensor networks caused by differences in control dependence. In *Proceedings of The Third IARIA International Conference on Systems and Networks Communications (ICSNC 2008)*, Oct. 2008.

[22] A. L. Vizine, L. N. de Castro, E. R. Hruschka, and R. R. Gudwin. Towards improving clustering ants: An adaptive ant clustering algorithm. *Informatica Journal*, 29(2):143–154, July 2005.

[23] M. Younis, M. Youssef, and K. Arisha. Energy-aware routing in cluster-based sensor networks. In *Proc. 10th IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunications Systems*, Oct. 2002.

[24] D. Zaharie and F. Zamfirache. Dealing with noise in ant-based clustering. *IEEE Trans. Evolutionary Computation*, pages 2395–2402, Sept. 2005.

[25] Y. Zhang, L. D. Kuhn, and M. P. Fromherz. Improvements on ant routing for sensor networks. In *Proceedings of the Fourth International Workshop on Ant Colony Optimization and Swarm Intelligence*, pages 154–165, Sept. 2004.

# Selection of Computing Elements for Energy Efficiency in Wireless Sensor Networks using a Statistical Estimation Method

Steven Corroy[▷] Jan Beiten[▷◦] Junaid Ansari[◦] Heribert Baldus[▷] Petri Mähönen[◦]
[▷]Philips Research, Distributed Sensor Systems
HTC 37-51, NL-5656AE, Eindhoven, The Netherlands, steven.corroy@philips.com
[◦]Department of Wireless Networks, RWTH Aachen University
Kackertstrasse 9, D-52072, Aachen, Germany, jan@mobnets.rwth-aachen.de

## Abstract

*A wide range of wireless sensor network applications are characterized by local processing of the sensed data and only meager data communication requirements. Since sensor nodes are battery powered and wireless communication bears a high energy cost, data transmission can be traded for on-node-computing in order to extend the lifetime of a node as well as the network. Furthermore, the energy consumption can be reduced significantly by selecting and realizing the application on an appropriate processing element. In this article, we propose a new statistical technique for energy consumption estimation for a specific application on various platforms. We have empirically verified the methodology on various classes of embedded processors commonly used for sensor nodes. The methodology is also applicable to multiprocessor platforms. Our solution is not only capable of achieving high degree of accuracy but also facilitates the application developer to evaluate different platforms without actually implementing the application on each of these platforms. Our experimental evaluation results on different platforms will help understand the implications of using different processing elements and their effects on the lifetime of a wireless sensor network.*

*Keywords: Wireless sensor networks; energy efficiency; energy consumption estimation.*

## 1 Introduction

Wireless Sensor Networks (WSNs) have a wide range of applications with long operational lifetime requirements. Since WSN nodes are generally battery operated, achieving long lifetime just by changing batteries is either too cumbersome or impossible. Therefore, efficient use of the available battery source becomes an important issue in WSNs. A sensor node essentially includes a microcontroller, a radio transceiver and a few sensors. The usage of all the components needs to be optimized for battery conservation. Radio communication exhibits the highest energy consumption budget in many applications while others are dominated by computations. Low power MAC protocols are designed to optimize the use of radio resource by periodically turning on/off the radio in order to save energy, while the radio is inactive. One class of MAC protocols is the common schedule based protocols like S-MAC [17] and its variants. These protocols coordinate the active periods of the nodes so that messages are sent only when all the nodes are active at a common wake-up period. Another common class of MAC protocols is the preamble sampling MAC protocols like B-MAC [13] and its variants, where nodes sleep and wake-up asynchronously and rely on long preambles to signal the upcoming data.

While MAC protocols play an important role in applications where data communication is dominant in terms of energy consumption, these have little influence in computationally intensive applications with meager communication requirements. In order to meet the computing requirements, many types of microcontrollers with different characteristics are used. These include ASIC (Application Specific Integrated Circuit), ASIP (Application Specific Integrated Processor), RISC (Reduced Instruction Set Computer) and CISC (Complex Instruction Set Computer). A DSP (Digital Signal Processor) is an ASIP specialized in digital signal processing. RISCs and CISCs are both GPPs (General Purpose Processors). Each of these different classes of processing elements suits better to different application demands. For instance, a DSP is more suitable for performing signal processing, e.g. computing Fast Fourier Transformation (FFT), while it is inefficient for controlling Serial Peripheral Interface (SPI) communication. In order to select the most power efficient processor, it is necessary to determine

the anticipated energy consumption of a processing element executing the application.

Code execution depends on several factors such as acquiring sensor data or other external inputs. Thus the energy consumed can only be measured at runtime. For a runtime measurement, the application needs to be implemented on the WSN platform. The implementation efforts are relatively high due to different instruction sets and processing element specifics. It is neither cost nor time efficient to implement an application on all the available WSN platforms in order to determine the optimal one. The number of implementations required can be lowered with the experience of a code developer, but still the number of possible platforms remains relatively high. Therefore, a technique to determine power consumption estimation without actual implementation effort is desirable.

Simulators such as a circuit simulator cannot produce accurate results for energy consumption, since they cannot completely simulate the wireless sensor node environment. Furthermore, simulators are hardware specific and are not available for each platform. The effort required to implement an application on a simulator is very similar to the effort required to implement the application on a real hardware platform. Therefore, it is easier to estimate energy consumption on a higher level independent from the actual implementation. The next higher level of abstraction that allows energy estimation is the code level. We describe in the following a methodology for estimating energy consumption of a specific application on a specific WSN platform at the code level. Our method is also applicable to platforms with multiple processors as we describe in the later sections. This article is an extended version of earlier paper [1], published in the International Conference on Sensor Technologies and Applications, SENSORCOMM 2008.

The rest of the article is organized as follows: Section 2 presents the related work in this area, Section 3 and 4 detail our solution, Section 5 presents the results that we obtained by applying our methodology on a single processing element on two different classes of application requirements. Section 6, describe the methodology applicable on multiprocessor architecture. Finally, in Section 7 we conclude the article.

## 2 Related Work

The energy consumption of a specific application depends on the hardware platform. The suitability of the hardware to the required software functionality allows to achieve energy efficiency. The energy consumption estimation for computing elements has been an on-going research issue [3, 5, 9]. However these studies concentrate on determining the energy consumption on one specific platform for a specific application.

Feinstein *et al.* [5] have developed a method to measure power consumption using a single measurement point. Their method assigns a unique ID to each process and logs in the corresponding real time power consumption and the execution time of the task. Although this approach enables a precise estimation of the energy consumption at the task-level, it requires a sensor node, for carrying out the measurements and an implementation of the application on each of the target platforms.

Bircher *et al.* [3] proposes to use event counters (e.g., DMA accesses or interrupts) for modeling the power consumption of a whole computing platform. From specific events to the microcontroller, their approach derives values for the rest of the platform e.g. I/O power consumption. Similar to the method by Feinstein *et al.* [5], this approach also requires the implementation of the application on each target platform.

PowerTOSSIM [14] is a tool for estimating the power consumption of applications developed in the TinyOS operating system environment for the supported platforms. Since it is a high level power consumption estimator, the accuracy is not high enough. PowerTOSSIM is based on the TinyOS simulator TOSSIM, which lacks the ability to model the simulations accurately owing to its inability to handle preemption of tasks and lackness of precise timings the execution of different functions [7], PowerTOSSIM also remains inaccurate.

A. Dunkels *et al.* [4] developed a software based technique for measuring online power consumption of a certain application developed for Contiki operating system running on Moteiv Inc.'s TmoteSky sensor node platform. Since the power consumption of the node in each state is known beforehand, a sensor node can estimate its own energy consumption by time-stamping each of its states. This technique however requires a full implementation of the application on the supported platform and hence makes it a time-taking and tedious job for the code developer.

Circuit simulators for energy consumption can give cycle accurate results but require long execution time. Niar *et al.* [9] proposes to combine statistical simulation to circuit simulation. It generates a short synthetic program representing the original application using statistical values, e.g., instruction distribution or cache miss rate. It then simulates the short synthetic program, giving substantial speed gains with only an average error of 3.8%.

The work by H. Joe *et al.* [6] aims at designing an accurate instruction level power estimator for sensor networks. They developed a power estimation tool using a machine instruction level simulator and correspondingly an energy consumption estimation module. The energy consumption estimation module is instructed by the instruction level simulator for profiling the energy consumption during the run time of the application. A post-processing module handles

the adjustments for the function call/returns and the I/O accesses to estimate the per node energy consumption in function calls and hardware components. Again this provides a very low level power consumption estimation which requires a pre-implementation of the application. Furthermore, for large simulation of sensor network applications, this approach is very slow.

In contrast to the approaches described above for power consumption estimation, we aim at estimating and evaluating the power consumption of a particular application on different platforms without implementing it. We trade-off estimation precision for ease of use. In order to evaluate and verify our work, we used three different microcontrollers in our experiments which represent the three different classes of computing elements. This includes the 8051 architecture [8], the MSP430 architecture [16] and the Coolflux architecture [10].

The 8051 is an 8-bit CISC type processor with Harward architecture as part of the CC2430 [15]. It was one of the first embedded processors that included CPU, RAM, ROM, I/O, interrupt logic and timers in a single package. Despite its design age, 8051 is still in use for many embedded system applications. The MSP430 is a 16-bit RISC with von Neumann processor architecture. Its design was specifically developed as a RISC architecture with very low power consumption. There is a wide selection of MSP430s in the market, providing different combinations of peripherals.

The NXP CoolFlux is a 24-bit DSP with features such as pipelining, MMU, register file structure and a very efficient Multiplication/Accumulation (MAC) implementation.

# 3 Methodologies for Power Consumption Estimation

Calculating the required number of instructions for a specific application and analyzing the required energy consumption enables to compare the energy efficiency of different computing elements. While considering different processing platforms, the operating clock frequencies vary significantly from one platform to another. It implies that the possible number of executable instructions within a certain time-frame varies correspondingly. For all the processing elements, Million Instructions Per Second (MIPS) can be calculated and is used as the common performance reference. However, the computational power consumption of a single instruction differs much from one platform to another. Certain instructions may require only one cycle on a RISC platform and take several hundred cycles on a CISC platform. Therefore, MIPS is not a reliable means to derive energy comparison. Other benchmarks are not widely available for embedded systems and are bound to only a small set of applications due to a different combination of instructions. So a new approach needs to be devised in order to analyze the energy consumption with a wider focus than conventional benchmarking. In the following, we present a method for estimating power consumption of an application utilizing the code level. It consists of two steps. First the code of a new application is divided in short blocks of code. We call *weight* a physical measure applying to a block of code, e.g., run time or energy consumption. In the following we explain the methodology restraining the meaning of *weight* to energy consumption for clarity purpose. Second, the empirical weight of the blocks of code is multiplied with their number of occurrence which gives an estimate of the overall power consumption of the application. Next we explain how to choose the relevant blocks for splitting the application and how to calculate their weights.

## 3.1 Blocks of Code Granularity

A high level programming language is composed of different instructions. An application is a sequence of instructions. The shortest atomic element of an application is thus an instruction. But an application can also be described as a sequence of different combination of instructions. For estimating an application we split it up into a large number of blocks of code. A block of code is a combination of instruction. The size of a block of code is determined by the number of instructions it contains. When splitting an application all blocks of code have the same size. The first step of our approach is to identify the relevant blocks independently. The granularity of the blocks is decisive for the accuracy and the complexity of the estimation. Long blocks of code give more accurate results because they contain more dynamic effects, like for instance compiler optimization. On the other side, increasing the length of the blocks of code increases quadratically the complexity of the method. Assume $l$ is the length of the blocks of code and $k$ the number of possible blocks (all different combination of instructions of size $l$), increasing to $2l$ the length of the blocks increases to $k^2$ the number of possible blocks. Our experiments show that single instructions ($l = 1$) such as additions, jumps and explicit memory accesses are flexible and efficient processing blocks for multiple platforms.

## 3.2 Blocks of Code Weight

The second step is to determine the weights assigned to blocks of code. A simple approach is to measure the energy consumption of all single blocks on each platform. We implement a program for each block of code (e.g. an addition or jump instruction) containing a sequence of the block in a row. We measure the energy expended by this program and derive the energy consumption of the block of code. Applying the calculated weights to real world WSN applications show that the results have a 95% confidence interval within

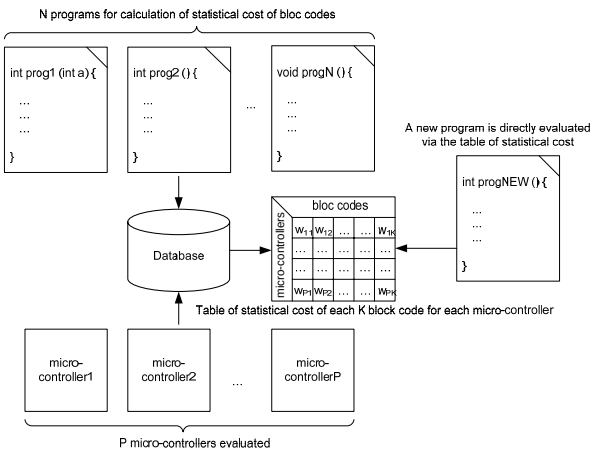



**Figure 1. Overall system for optimal computing element selection.**

[-27% 23%]. The imprecision of the results is caused by the compiler optimizations and runtime effects. In order to improve the accuracy of the results, a statistical approach is introduced which decreases the influence of the compiler optimizations.

### 3.2.1 System Description

It is assumed that the effects of the compiler optimizations in reducing the energy consumption are statistically foreseeable. Instead of using the measured energy consumption of constant-sized-blocks-of-code to determine the weights of the blocks, we measure the energy consumption of several representative WSN applications and derive statistically the cost of each block of code. This approach has the advantage of taking into account much of the dynamic effects occurring due to compiler optimizations while keeping shorter blocks of code for fast and flexible power estimation. Our overall system is illustrated in Figure 1. We implement on $P$ platforms $N$ applications {prog1,...,progN} such as simple instructions concatenation and also more sophisticated algorithms such as FFT or field calculations. The power consumption and the runtime of those applications are empirically measured. We maintain a database with tables $T_1$ and $T_2$ for storing:

1. The runtime $t_{np}$ and the energy consumption $e_{np}$ of each application for each platform with $n \in [1, N]$ and $p \in [1, P]$.

$$T_1 = \begin{pmatrix} (t_{11}, e_{11}) & (t_{12}, e_{12}) & ... & (t_{1P}, e_{1P}) \\ (t_{21}, e_{21}) & (t_{22}, e_{22}) & ... & (t_{2P}, e_{2P}) \\ ... & ... & ... & ... \\ (t_{N1}, e_{N1}) & (t_{N2}, e_{N2}) & ... & (t_{NP}, e_{NP}) \end{pmatrix}$$

2. The number of occurrence $c_{kn}$ of each blocks of code in each application (assuming $K$ possible blocks of code) with $k \in [1, K]$ and $n \in [1, N]$.

$$T_2 = \begin{pmatrix} c_{11} & c_{12} & ... & c_{1K} \\ c_{21} & c_{22} & ... & c_{2K} \\ ... & ... & ... & ... \\ c_{N1} & c_{N2} & ... & c_{NK} \end{pmatrix}$$

We use a linear programming solver to find the weights $w_{kp}$ of each block of the code on each platform with $k \in [1, K]$ and $p \in [1, P]$. The problem to solve has the following form for energy consumption:

$$\text{for each } n \in [1, N],\ p \in [1, P],\ \sum_{k=1}^{K} c_{kn}.w_{kp}^{e} = e'_{np} \quad (1)$$

and for run time

$$\text{for each } n \in [1, N],\ p \in [1, P],\ \sum_{k=1}^{K} c_{kn}.w_{kp}^{t} = t'_{np} \quad (2)$$

where $e'_{np}$ is the statistical energy consumption and $t'_{np}$ the statistical run time of a specific program on a specific platform ($e'_{np} \geq e_{np}$ because compiler optimizations tend to remove or regroup instructions, thus making the code more efficient and more compact than originally written). We put the following constraints which represent the fact that the energy consumption of one operation must be positive:

$$\text{for each } k \in [1, K],\ \ p \in [1, P],\ w_{kp}^{e} > 0, w_{kp}^{t} > 0 \quad (3)$$

Finally, we minimize the sum of the squared relative error between $e'_{np}$ and $e_{np}$ among all the applications:

$$\text{for each } p \in [1, P],\ \sum_{n=1}^{N} \left( \frac{e'_{np}}{e_{np}} - 1 \right)^2 \quad (4)$$

Respectively, we minimize the error in terms of time

$$\text{for each } p \in [1, P],\ \sum_{n=1}^{N} \left( \frac{t'_{np}}{t_{np}} - 1 \right)^2 \quad (5)$$

From now on, if one wants to estimate the power consumption (respectively the run time) of a new application, one just needs to count the number of occurrences of each block of the code in the program (e.g., with a parser) and to multiply them with their statistical cost. We maintain a database table $T_3$ for the weights of the form

$$T_3 = \begin{pmatrix} (w_{11}^{t}, w_{11}^{e}) & (w_{12}^{t}, w_{12}^{e}) & ... & (w_{1P}^{t}, w_{1P}^{e}) \\ (w_{21}^{t}, w_{21}^{e}) & (w_{22}^{t}, w_{22}^{e}) & ... & (w_{2P}^{t}, w_{2P}^{e}) \\ ... & ... & ... & ... \\ (w_{K1}^{t}, w_{K1}^{e}) & (w_{K2}^{t}, w_{K2}^{e}) & ... & (w_{KP}^{t}, w_{KP}^{e}) \end{pmatrix}$$

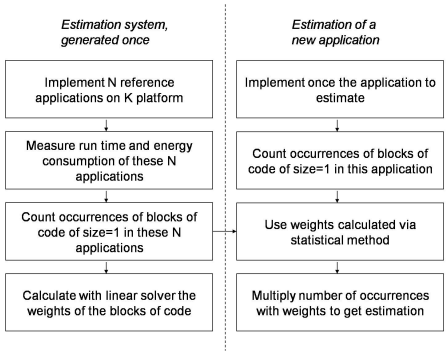



**Figure 2. Overview of estimation methodology.**

| *Instruction* | *8051 8 bit* | *8051 16 bit* | *8051 float* | *CF 24 bit* |
|---|---|---|---|---|
| Summation | 16.17 | 53.79 | 158.07 | 0.01 |
| Subtraction | 21.12 | 49.5 | 816.42 | 0.10 |
| Multiplication | 31.68 | 86.46 | 611.49 | 0.85 |
| Division | 76.23 | 127.05 | 712.80 | 2.27 |
| IF Then | 4.95 | 42.24 | 273.57 | 0.62 |
| IF Then ELSE | 138.93 | 111.21 | 2749.89 | 0.00 |
| Loop Repetition | 5.61 | 4.95 | 28.38 | 0.00 |
| Array access | 9.57 | 0.00 | 106.26 | 0.10 |
| Assignment | 20.13 | 29.70 | 130.68 | 0.00 |
| Modulo | 667.92 | 714.78 | 113.85 | 0.61 |

**Table 1. Statistical energy consumption of processing elements in nano Joules.**

We illustrate a global overview of the methodology in Figure 2 The left part of the schema illustrates how the estimation system is built. This computation work is processed only once to generate all the statistical weights. The right part illustrates the work which take place when estimating a new application, this is the part that needs to be fast and simple.

#### 3.2.2 Results

The results for finding the weight of blocks of code are shown in Table 1 (note that only a subset of all possible blocks of code is presented). Using those weights, we estimated the energy consumption of several other applications (two of them are presented in more detail in Section 5) and verified the results with measurements. Our method shows significantly better results for all the platforms as shown in Table 2. The algorithm is reliable and accurate for both computational time and computational energy consumption.

#### 3.2.3 Multiprocessor Approach

The evaluation method that we developed is not only capable to forecast energy consumption for a single computing element but also for a multi-processor platform. We assume a proper mapping of the application on the different processors. We estimate the energy consumption of the code running on each processing element separately. Then, we calculate the power required for communication between all processors (more details in Section 4) and add it to the previously estimated processor consumptions. This approach enables to find the optimal WSN platform for a specific application when it requires more than one processor (the case study in Section 5.2 highlights it). For example combining a CISC for controlling the peripherals and a DSP for computing may be beneficial as long as the overhead in terms of internal communication and double-powering does not counter-balance the gains in terms of processing efficiency.

| *Processing element* | *Word-Width* | *95% Confidence interval* |
|---|---|---|
| 8051 | 8 | [-13 %;13 %] |
| 8051 | 16 | [-14 %;14 %] |
| MSP430 | 16 | [-13 %;13 %] |
| Coolflux | 24 | [-8,3 %; 8,3 %] |

**Table 2. Confidence intervals for energy estimation on 8051, MSP430 and Coolflux.**

## 4 WSN platform Power Consumption

In Section 3, we presented a method to determine the power consumption of a specific application on a specific computing element. Because our main interest is in WSN applications, we propose a method to estimate the energy consumption of the whole WSN platform using the forecast for the computing element. We first indicate the dominant factors for WSN platforms with respect to power consumption.

### 4.1 Power Consumption Factors in WSN Platforms

*Data Acquisition* is the sampling of data from a sensor of the processing element. It contains the power consumption of the sensor (taken from the data sheet or measured once) and the energy needed by the processing element to control the communication interface. This last value can be measured one time for each platform and reuse for any new application to be benchmarked.

*Computation* is evaluated as described in Section 3.

*Power Mode Change* is the action for the processing element to go from active mode to sleep mode (and vice-versa). It is specific for each platform, but only have to be measured once for each platform.

*Power Down Mode* or Sleep Mode represents the energy consumed by the processing element while sleeping. Usually processing elements provide a set of power down modes, diverging with available peripherals, timers and memory. Each power mode was measured once per platform.

| *Task* | *Coolflux* | *8051* | *MSP 430* |
|---|---|---|---|
| Data acquisition | 654 | 306 | 170 |
| Computation | 0.05 | 291 | 46 |
| Power mode changes | 0.00 | 585 | 158 |
| Idle power | 1211 | 565 | 794 |
| Overall | 1866 | 1747 | 1167 |

**Table 3. Power consumption of the first case (in $\mu$J) on the evaluated platforms.**

*Wireless Communication* is predicted using methods described in [2] [11].

### 4.2 Methodology

In order to evaluate the total power consumption of a WSN platform, we add the individual power consumption of all those factors together. In order to predict the time that an application spends in sleep mode or to foresee the number of power mode changes, we use a periodic model which is used in most of the WSN applications and consists of six phases: 1) Power-up mode, 2) Sample data from the sensor, 3) Process data, 4) Transmit data, 5) Power down mode and 6) Sleep mode. The total period of one cycle is chosen by the programmer and the sleep time is determined by subtracting the foreseen time for data sampling, computation, transmission and power mode change.

## 5 Case Studies

In this section, we describe the performance results obtained from our implementation. We carried out the implementation and evaluation of two very typical sensor network applications and discuss their energy efficiency on different processing elements. We consider single processor as well as multiprocessor solutions in this regard.

### 5.1 Applications with Moderate Computational Requirements

In order to provide a concrete example on how to apply our method, the first application that we consider is acceleration sensing. An accelerometer acquires 16 bit values at a frequency of 50Hz. If the values are greater than a specific threshold, an alert is transmitted to a hub. The computing element is shut-down between each sample. Table 3 shows the break-down of power consumption of the application on various platforms (over 1s). In the following, we describe the various sources of energy consumption in the sample application.

#### 5.1.1 Data Acquisition

Out of the previously described classes of processing elements, GPPs are the most efficient for data acquisition. Since GPP platforms have integrated peripheral interfaces such as UART and SPI, these can communicate with the sensors very efficiently. Both MSP430 and 8051 have an integrated SPI. The activation of the interface consumes only 60 $\mu$W of power, which is insignificant compared to the active power consumption of the two processors. In the sample application, the processing elements cannot utilize the idle time during the data acquisition for computations as there is only a single task running. Furthermore, energy savings cannot be obtained just by switching into the sleep mode as the time interval is too short.

Coolflux platform requires a significant amount of energy for data acquisition owing to the absence of a dedicated serial interface and can only emulate the serial protocol in the software (bit-banging). This causes very high acquisition time and current consumption on Coolflux. Therefore, a Coolflux is the least efficient processing element during data acquisition.

#### 5.1.2 Computation

From the computational point of view, Coolflux is the most energy and time efficient processor element in our study. Coolflux contains two Arithmetic Logical Units (ALUs) and can therefore execute multiple instructions in a single cycle. In the 16-bit data processing application, Coolflux can work on its 24-bit native word-width. 8051 and MSP430 require a comparable run-time for computations. 8051 has an 8 bit ALU. Therefore, it requires additional cycles for each 16 bit operation. MSP430 as a RISC processor, is more cycle effective and additionally can work at its native word-width. Therefore, the power consumed by MSP430 is significantly lower for this application. Overall, Coolflux is magnitude times more energy and time effective than the CC2430 and the MSP430.

#### 5.1.3 Power Mode Changes

8051 requires a significant amount of energy for a single power mode change. For a sampling rate of 50 Hz, the energy spent in mode transitions is more than the computation itself. 8051 spends about 94 % of its time in power down mode and 4 % of time in power mode changes, but through the low energy consumption in power down mode, most of the energy is spent in the mode changes. MSP430 requires about the same time to switch to sleep mode, but through its lower current, the energy consumption for that is significantly lower. Contrary to the GPPs, Coolflux platform decouples the clock network and does it within a few cycles. The energy consumed for the power mode change is only 0.16 nW.

#### 5.1.4 Energy Consumption in Sleep Mode

Coolflux can change its power mode very fast. On the other hand, the power down mode itself is less effective. Coolflux consumes about 25 % of its active power in sleep mode. By spending a long time in sleep mode, the overall power spent in sleep mode exceeds the power spent in all the other operations in the sample application.

The power down modes of MSP430 and 8051 consume about the same amount of energy. 8051 also supports more effective power modes than the chosen power mode in our application, but only supports half of its memory. MSP430 also supports other power down modes but since it has a relatively large switching time, these are not used in our sample application.

#### 5.1.5 Efficiency depending on the Sampling Rate

We change the accelerometer sampling rate to understand its impact on the overall energy efficiency. Results are shown in Figure 3. Note that CC2430 relates to a platform containing a 8051. If the sample rate is very low, 8051 turns out to be the most energy efficient processing element, because of its low power consumption in power down mode. MSP430 is more effective in data acquisition and power mode changes. Therefore, starting at a sample rate of 18 Hz, MSP430 becomes more efficient for our sample application.

It may be noted that Coolflux becomes more power efficient than 8051 beyond a certain sampling rate. Although the Coolflux is less effective in data acquisition, its ability for computation and energy effective power mode changes compensates it. Overall, the MSP430 is relatively the most preferred processing element for high sampling rates.

The maximum supported sampling rates are 1310 Hz for Coolflux, 2564 Hz for 8051 and 3322 Hz for MSP430. A lower maximum sampling frequency of Coolflux is caused by its inability for higher data acquisition rate. The difference between 8051 and MSP430 is caused by the computational ability of MSP430.

#### 5.1.6 Efficiency depending on the Computation

Depending upon the computational power, the efficiency of the processing elements is also evaluated. We modify the number of calculations in the application. Figure 4 shows the energy consumption depending on the proportion of computations compared to the initial application. It allows estimating, how much computation would make it worth to implement the algorithm on another processing element. 8051 becomes less effective than Coolflux for the application if the number of computations is doubled. If the amount of computation is increased by a factor of 20, Coolflux becomes more efficient than MSP430.

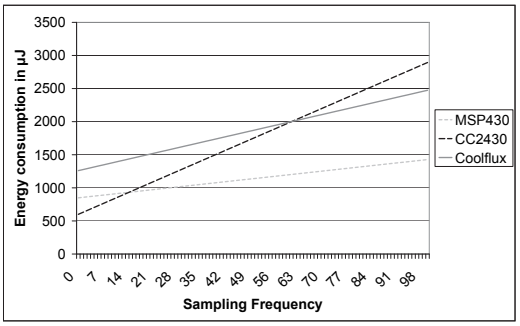



**Figure 3. Influence of the sensor sampling rate on the energy consumption of the 8051 (inside CC2430), MSP430 and Coolflux.**

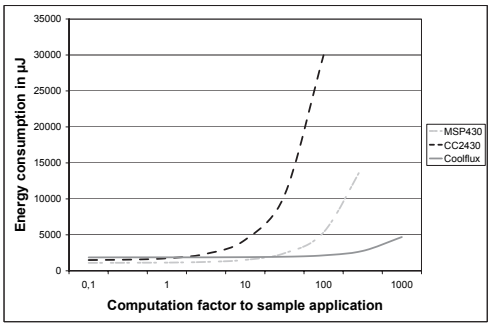



**Figure 4. Influence of the computational load on the energy consumption of the 8051 (inside CC2430), MSP430 and Coolflux.**

In conclusion, the presented example highlights the attributes of different classes of processing elements. As a DSP is more effective for calculations, GPPs are more effective during data acquisition. The sample application considered above is very simple and requires only a very limited number of computations. Also, the number of computations required are clearly dominated by conditional jumps and not by arithmetic operations. Therefore, a simple processor efficiently supporting fast switching modes and low energy consumption in power down mode is the best choice for this kind of application. Multiprocessor concepts are not suitable, since the complexity of the algorithm is low and the energy spent in power down mode exceeds all the possible energy improvements.

### 5.2 Applications with High Computational Requirements

A second study case is the recognition of heart rate from ECG data using a real-time algorithm. The heart rate must

| *Task* | *8051* | *MSP430* | *MSP430+Coolflux* |
|---|---|---|---|
| Data acquisition | 7.5 | 3.0 | 3.0 |
| Computation | 6 | 3.9 | 0.1 |
| Wireless communication | 1.2 | 1.3 | 1.2 |
| Serial communication | | | 0.1 |
| Power mode changes | 5.9 | 2.4 | 2.4 |
| Idle power | 0.0 | 0.2 | 1.3 |
| Overall | 20.6 | 10.8 | 8.1 |

**Table 4. Power consumption of the second case (in mJ) on the evaluated platforms.**

be sampled at high frequency to obtain reliable results. The amount of computations is high because of the real-time nature of the algorithm. The heart rate calculation is sent regularly to a hub. We evaluated an in-house developed heart rate analysis code using a sampling rate of 500 Hz. It calculates the heart beat using an algorithm described in [12]. This algorithm does not store all the heart beat samples and therefore requires little memory. The algorithm is dominated by comparison operations and 32 bit calculations. The application is divided into data acquisition, computation and wireless communication.

A comparison of a platform combining MSP430 and Coolflux is shown in Table 4 (over 1s). A stand alone Coolflux solution is not appropriate, because of its ineffective data acquisition. On the combined platform, data acquisition is done on MSP430 because there is no ADC on Coolflux. Due to the high sampling rate, it takes a significant amount of time and energy. The multiprocessor solution has the advantage to allow higher sampling rates than the single processor solution. The computations are very effective on Coolflux DSP. Indeed the 12-bit data gets double after multiplication and fits very well to the 24-bit architecture of Coolflux unlike MSP430 or 8051. Therefore, the Coolflux is magnitude times more effective than MSP430 or 8051. Altogether with the multiprocessor approach, the power consumption is reduced by a factor of 28%. The proposed multiprocessor solution benefits from the strenghs of both classes of processing elements, namely the computational ability of DSPs and the fast data acquisition of GPPs, but this at the expense of size and complexity.

# 6 Multi-processor Scheduling

It is evident from the previous section that in some cases, an application can be implemented more energy efficiently on multiple specialized processing elements rather than on a single multi-purpose processing element. This leads to finding the best combination of the selected hardware. Simultaneously, it is required to split-up the application into tasks that are can be assigned to the selected hardware. The hardware selection and the correspondingly task scheduling is done mostly manually, which remains inefficient. With a wider range of energy efficient processing elements and furthermore their distributed nature such as in wireless sensor networks, the task becomes increasingly complex and unmanageable.

In Section 3, we have developed an approach that can be used to support power efficient implementation of a given application on multi-processor hardware. Tasks such as data acquisition, performing calculations, data communication, power mode switching and energy consumption in idle mode are the fundamental blocks of the overall energy consumption on a sensor node. In our approach, after the system designer specifies an application, the code is broken down to granular blocks and weights are calculated for each of the block on all the processing elements under consideration. In addition to the application related facts, the designer also specifies timing and data dependency of the application tasks. The hardware and software requirements are formulated as conditions. Each combination of hardware with the regarding scheduling has to meet all the conditions. The sum of the weights of each block shows the total energy consumption of a working solution. The minimum of this sum shows the most energy efficient solution of the evaluated hardware.

The solution space for the described conditions grows exponentially with the number of considered processing elements and tasks. In order to calculate the energy consumption of a wide range of possible solutions, NP solvers are used. Integer Linear Programming (ILP) solvers can directly be used to calculate the most energy efficient solution. In the following, we describe the detailed formulation of the multi-processing scheduling problem to be solved through ILP.

## 6.1 Attributes of the Tasks

Each task requires a set of constraints in order to describe their behaviour and to allow the formulation of equations for their scheduling.

| | |
|---|---|
| $E_{\beta,\gamma}$ | Energy consumed by task $\beta$ on hardware $\gamma$ |
| $T_{\beta,\gamma}$ | Average time required per second for task $\beta$ on node $\gamma$ |
| $T_{\beta}$ | Average maximum execution time required for task $\beta$ |
| $C_{\beta 1,\beta 2}$ | Number of exchangeable bits from task $\beta 1$ to $\beta 2$ |
| $\Xi$ | Number of all tasks to be scheduled |
| $O_{\beta}$ | Average number occurrence of task $\beta$ |

$E_{\beta,\gamma}$ describes the energy consumed by task $\beta$ on hardware $\gamma$. Also the run-time of a task $T_{\beta,\gamma}$ on a certain hardware has to be ascertained. It is referenced against the

scaled real-time constraint $T_\beta$. These values are scaled to the required time per second. This allows to consider the number of occurrences and the elapsed time on the hardware simultaneously. If the number of exchangeable bits $C_{\beta 1,\beta 2}$ for a task is greater than zero, both parts are dependent on each other. Additionally, this value is used to calculate the energy consumption cost of the communication task.

### 6.2 Attributes of the Networked Nodes

The networked nodes require a set of attributes in order to distribute the hardware according to the specification.

| | |
|---|---|
| $S_{\alpha,\beta,\gamma,\delta}$ | Task $\beta$ is scheduled on node $\alpha$ on hardware $\gamma$ with ID $\delta$ |
| $\Upsilon$ | Available sensor nodes |
| $\Delta$ | Available hardware IDs |

The variable $S_{\alpha,\beta,\gamma,\delta}$ describes the scheduling of task $\beta$ on hardware $\gamma$ with hardware ID $\delta$ to the node $\alpha$. Furthermore, the scheduling implicitly contains a hardware selection $\gamma$. It is possible to have multiple instances of $\gamma$ in the network node and these instances are numbered with $\delta$.

### 6.3 Attributes of the Hardware

The hardware requires attributes in order to describe the hardware for the solver and checking the suitability of specific tasks. The hardware attributes optimize the hardware for the constraints given by the tasks and the network nodes' attributes.

| | |
|---|---|
| $EI_\gamma$ | Energy consumed by hardware $\gamma$ in idle mode |
| $CC_{\alpha 1,\alpha 2,\gamma 1,\gamma 2,\delta 1,\delta 2}$ | Communication cost between hardware $\gamma 1$ and $\gamma 2$ |
| $\Gamma$ | Available hardware |
| $EC_\gamma$ | Energy consumption of power mode change on hardware $\gamma$ |

The idle state energy consumption $EI_\gamma$ becomes significant when a processing element is used very rarely. Each hardware has certain available peripherals $AP_{\gamma,\epsilon}$. The communication cost of two hardware elements $CC_{\alpha 1,\alpha 2,\gamma 1,\gamma 2,\delta 1,\delta 2}$ depends on hardware location. $EC_\gamma$ describes the energy required to change power mode on hardware $\gamma$.

### 6.4 Computing the Overall Energy Consumption

Integer Linear Programming requires an objective function that is either maximized or minimized. The term to be minimized in our case is the energy consumption of the overall network. It consists of the energy consumption of all the tasks running on different nodes. The run-time and real-time constraints are applied to the summed energy consumption formulation in order to obtain the optimum.

$S_{\alpha,\beta 1,\gamma,\delta 1}$ is used as Boolean including the scheduled elements. The first sum describes the energy consumption of the tasks.

$$\sum_{tasks} = \sum_{\alpha_\epsilon \Upsilon, \beta_\epsilon \Xi, \gamma_\epsilon \Gamma, \delta_\epsilon \Delta} S_{\alpha,\beta,\gamma,\delta} E_{\beta,\gamma}. \quad (6)$$

The second sum contains the energy consumed in idle mode. Therefore the average run-time in idle mode is calculated by subtracting the time spent in active mode. This is multiplied by the energy of the elements in idle mode

$$\sum_{idle} = \sum_{\alpha_\epsilon \Upsilon, \gamma_\epsilon \Gamma, \delta_\epsilon \Delta} (1 - \sum_{\beta_\epsilon \Xi} S_{\alpha,\beta,\gamma,\delta} T_{\beta,\gamma}) EI_\gamma. \quad (7)$$

The next sum is the communication costs between the tasks. Again $S_{\alpha,\beta 1,\gamma,\delta 1}$ is used as Boolean to exclude not scheduled tasks, whereas $C_{\beta 1,\beta 2}$ describes the amount of communication and $CC_{\alpha 1,\alpha 2,\gamma 1,\gamma 2,\delta 1,\delta 2}$ the communication costs

$$\sum_{com} = \sum_{\alpha_\epsilon \Upsilon, \beta 1_\epsilon \Xi, \beta 2_\epsilon \Xi, \gamma_\epsilon \Gamma, \delta 1_\epsilon \Delta, \delta 2_\epsilon \Delta} C_{\beta 1,\beta 2} S_{\alpha,\beta 1,\gamma,\delta 1} * S_{\alpha,\beta 2,\gamma,\delta 2} CC_{\alpha 1,\alpha 2,\gamma 1,\gamma 2,\delta 1,\delta 2}. \quad (8)$$

The last sum describes the energy spent on task mode changes. The assumption of diverging sample rates of the different applications does not allow to combine tasks to save energy through a decrease of power mode changes.

$$\sum_{changes} = \sum_{\alpha_\epsilon \Upsilon, \beta_\epsilon \Xi, \gamma_\epsilon \Gamma, \delta_\epsilon \Delta} S_{\alpha,\beta 2,\gamma,\delta 2} O_\beta EC_\gamma. \quad (9)$$

All considered energy consumptions can be summed together in

$$\sum_{total} = \sum_{tasks} + \sum_{idle} + \sum_{com} + \sum_{changes} \quad (10)$$

The implementations constraints need to be formulated. For example equation 11 describes the real-time constraints of the problem. It is necessary to be sure, that the run-time of all scheduled tasks on the hardware is smaller than the real-time constraint of the task.;

$$\sum_{\alpha_\epsilon \Upsilon, \gamma_\epsilon \Gamma, \delta_\epsilon \Delta} S_{\alpha,\beta,\gamma,\delta} T_{\beta,\gamma} \leq T_\beta. \quad (11)$$

Additional constraints are required for computational restrictions, fixed hardware assignments, identification of hardware in a specific node and hardware restrictions. These are formulated analogously.

This formulation as ILP of minimization of the global energy consumption of the network, although not allready solved, provides some understanding of the problem. It highlights the trade-off between computation, communication, power mode changes and idle mode that need to be take in account to reach an optimal solution.

## 7 Conclusions and Future Work

In this article, we described a new method for estimating the power consumption of a particular application on different wireless sensor node platforms. The method involves slicing down a whole application into smaller granular blocks of code. We use a linear programming solver to determine the weights associated for each of the code block on each platform. Through a detailed case study, we analyzed the trade-offs among CISC, RISC and DSP approaches for WSN nodes and showed empirically that our method is accurate. We carried out the evaluation of our methodology on 8051, MSP430 and Coolflux processors, representing the three processor classes. Our method provides easy way to estimate the power consumption but trades-off the accuracy. However, it is worth noticing that our method achieved a worst-case accuracy of 86%. We evaluated an application with meager computing requirements as well as a computationally intensive application. Our scheme requires just a single code implementation of the application for determining the most power efficient computing element, which will help code developers easily select the most suitable platform. The presented scheme also allows efficient benchmarking of the processing elements and determining the most energy efficient element, thereby saving costs and implementation efforts. We have also presented an extension of the scheme to multi-processor architectures. We have applied linear programming approach to efficiently schedule different tasks on a multi-processor platform. Real world applications are typically realized on a number of network nodes rather than just on a single node. Therefore, a local energy optimal solution may not necessary be the global energy optimal solution. In order to find a globally optimized estimate, a system wide hardware selection and scheduling is required. It is possible to build up a system of linear conditions, which represent the restrictions for a practical WSN. Using linear programming, the global energy optimal implementation of a network can be calculated based on the local estimation provided by our methodology.

## Acknowledgments

We would like to thank the financial support from E.U. (project IST-034963-WASP), Philips Research, Deutsche Forschungsgemeinschaft through the UMIC-excellence cluster and RWTH Aachen University.

## References

[1] S. Corroy, J. Beiten, J. Ansari, H. Baldus, and P. Mähönen. Energy efficient selection of computing elements in wireless sensor networks. In *International Conference on Sensor Technologies and Applications (SENSORCOMM 2008)*, pages 312–318, 2008.

[2] M. Achir and L. Ouvry. QoS and energy consumption in wireless sensor networks using CSMA/CA. Technical report, Electronics and Information Technology Laboratory Atomic Energy Commission, 2005.

[3] W. Bircher and L. John. Complete system power estimation: A trickle-down approach based on performance events. In *IEEE International Symposium on Performance Analysis of Systems & Software*, April 2007.

[4] A. Dunkels, F. Österlind, N. Tsiftes, and Z. He. Software-based sensor node energy estimation. In *Proceedings of the 5th international conference on Embedded networked sensor systems*, pages 409–410, 2007.

[5] D. Feinstein, M. Thornton, and F. Kocan. System-on-chip power consumption refinement and analysis. In *6th IEEE Dallas Circuits and Systems Workshop on SoC*, 2007.

[6] H. Joe, J. Park, C. Lim, D. Woo, and H. Kim. Instruction-level power estimator for sensor networks. *ETRI Journal*, 30(1):47–58, February 2008.

[7] O. Landsiedel, H. Alizai, and K. Wehrle. When timing matters: Enabling time accurate and scalable simulation of sensor network applications. In *Proceedings of the 7th international conference on Information processing in sensor networks*, pages 344–355, 2008.

[8] I. S. MacKenzie. *The 8051 Microcontroller*, volume 4th Edition. Prentice Hall, 2001.

[9] S. Niar and N. Inglart. Rapid performance and power consumption estimation methods for embedded system design. In *7th IEEE Int. Workshop on Rapid System Prototyping*, pages 47–53, June 2006.

[10] NXP, http://www.coolfluxdsp.com. *Coolflux DSP*, 2004-05.

[11] L. Ouvry and M. Achir. Probabilistic model for energy estimation in wireless sensor networks. *Lecture Notes in Computer Science*, 3121/2004:157–170, 2004.

[12] J. Pan and W. Tompkins. A real time QRS detection algorithm. *IEEE Trans. On Biomedical Engineering*, 32, 1985.

[13] J. Polastre, J. Hill, and D. Culler. Versatile low power media access for wireless sensor networks. In *Proc. of SenSys*, pages 95–107, 2004.

[14] V. Shnayder, M. Hempstead, B. rong Chen, G. W. Allen, and M. Welsh. Simulating the power consumption of large-scale sensor network applications. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 188–200, 2004.

[15] Texas Instrument, http://focus.ti.com/docs/prod/folders/print/cc2430.html. *CC2430*, 1995-2007.

[16] Texas Instruments, http://focus.ti.com/docs/prod/folders/print/msp430f1611.html. *TI MSP430x1611 - Mixed Signal Microcontroller*, 2005.

[17] Y. Wei, J. Heidemann, and D. Estrin. Medium access control with coordinated adaptive sleeping for wireless sensor networks. *IEEE Trans. on Net.*, 12(3):493–506, June 2004.

# Context Modeling for Cross-layer Context Aware Adaptations

Ruwini Kodikara
Centre for Distributed
Systems & Software Engineering
Monash University
900, Dandenong Road, Caulfield
East Vic 3145, Australia

Arkady Zaslavsky
Department of Information &
Communication Technology
Luleå University
of Technology
931 87 Luleå,Sweden

Christer Åhlund
Division of Mobile
Networking & Computing
Luleå University
of Technology
931 87 Skellefteå,Sweden

## Abstract

*Demand for real-time services over the Internet while moving, is growing rapidly. This necessitates efficient delivery of wireless real-time traffic. Limitations of existing layered protocol stack for wireless networks lead to the proposal of cross-layer interactions as an alternative solution. At the same time, next generation ubiquitous computing drives wireless applications and protocols to be context aware. A generic context aware architecture with context modeling can aid increasingly demanding real-time applications over highly dynamic wireless networks to be cross-layer context aware and adaptive. Moreover, a generic architecture can make lower layer protocols to be context aware and adaptive to various situations dynamically. This article discusses the adaptive approach supported by proposed cross layer context aware architecture called CA3RM-Com. The scope of this article is to discuss context modeling specifically and address the issue of context representation of multi-layer context for various adaptive situations. Various single layer and multi-layer cross-layer adaptations and the representation of context parameters with respect to each layer of the protocol stack is discussed. We discuss how these adaptations can be operated in the proposed CA3RM-Com architecture. Context aware adaptive multi-homed Mobile IP is discussed as an example adaptation that the architecture can support. Moreover, the extended simulation of context aware adaptive multi-homed Mobile IP is discussed.*

***Index Terms***—Context Awareness; Real-time Communications; Mobile IP

## 1. Introduction

Rapid growth of Internet, increasing demand to stay connected while on the move are the driving forces of evolving wireless technologies. Demand for real time wireless traffic such as voice, multimedia teleconferencing, mobile PC, mobile TV, mobile games and video conferencing is increasing day by day. Moreover, next generation computing is becoming ubiquitous necessitating wireless technologies to accomplish context aware adaptations [1].

Unlike in static wired networks, wireless networks require adaptations to various situations for efficient real-time communication. The reasons for such adaptations are the challenges arise due to nature of wireless networks, wireless devices in use and characteristics of real-time media. Wireless networks have inherent limitations of radio links such as noise, shadowing, channel fading and interference; network topology is unpredictable due to highly dynamic nature; network management and routing is complex compared to static wired networks. Moreover, there are constraints in wireless network devices in terms of energy and computational capabilities. On the other hand, real-time applications are bandwidth intensive, latency sensitive and loss tolerable in nature [2]. Real-time applications demand for high data rates and are coupled with stringent delay constraints. These data packets should be delivered with tolerable delays and loss rates to avoid decode errors and to guaranty the Quality of Service (QoS) levels. Even though the technology is growing and trying to satisfy the growing demands of applications, there exists an impediment in wireless networks due to the dynamic nature and limited resources of the underlying network compared to wired networks [1]. Because of this, the next generation wireless technologies should facilitate adaptations to various situations dynamically. Challenge in facilitating dynamic adaptations is the awareness of the situations.

Modern communication systems use layered protocol stack for inter-networking due to several reasons. The initial purpose of layering [3] was to ensure modularity. In a modular system, each module has clearly defined functions, procedures, specified and controlled interactions among modular component to enable layer independence. Because of the abstractions, the overall system is easy to understand.

Hence, layered approach reduces system design complexity. So, layering ensures easy implementation and maintainability. Moreover, the layered approach assures the interoperability between different systems. Standardized abstractions allow designers of various subsystems to focus on their particular subsystem without bothering about the entire system interoperability.

Though strict layered approach serves as an elegant solution for inter-networking static wired networks, it is argued in the literature that the layered protocol stack is not adequate for efficient functionality of wireless networks [4],[5]. The layered protocol stack is insufficient to cater for the adaptations of demanding applications and complex networking conditions. Further, the system performance improvements are restricted in layered approach, because most of the time performance improvement is achieved through multi-layer joint interactions.

The rest of the article is organized as follows. Section 2 discusses the background and related work. Section 3 presents motivation of the context modeling, classification of context parameters related to adaptive situations and context representation mechanism we propose. Section 4 discusses the simulation of context aware adaptive handover. Finally, Section 5 concludes the article.

## 2. Background and Related Work

### 2.1 Cross Layer Design

Cross-layer interaction was proposed as a solution to overcome limitations of layered protocol stack when applied in wireless networks [6]. Cross-layer design is defined as ***Protocol design by the violation of a reference layered communication architecture with respect to that layered architecture*** *[7].*

According to aforementioned definition, designing protocol by violating the reference architecture, by allowing direct communication between protocols at nonadjacent layers or sharing variables between layers is cross-layer design with respect to the reference architecture. It is argued that violation of layered architecture includes creating new interfaces between layers, redefining the layer boundaries, designing protocol at a layer based on the details of how another layer is designed, joint tuning of parameters across layers and giving up the luxury of designing protocols at different layered independently [7].

A number of cross layer designs are proposed in the literature including [8], [9], [10], [11], [12]. They can be classified as specific solutions and generic architectures.

Specific solutions are not based on the objective of providing a generic framework and are tailored towards a specific adaptation/requirement. Joint adaptations with multi-layer interactions were proposed with various performance objectives. Joint adaptations of adaptive routing and rate/channel adaptations are proposed in [13]. Joint congestion Control, rate control, adaptive routing and channel scheduling is proposed in [14]. Cross-layer mechanism of joint channel scheduling and rate/channel adaptation is presented [15]. Congestion control together with channel scheduling is proposed in [16]. Adaptive routing with joint link rate adaptation is discussed in [17]. Link rate adaptation ([12], [18]) and joint power control is studied in [19]. Joint application layer and lower layer interactions are also presented. Packetization with joint link rate adaptation ([11],[20]), Packetization with joint adaptive routing [21], Packetization with adaptive routing and adaptive modulation are studied in [22]. Energy optimizes routing was proposed in [23]. QoS Control with joint adaptive modulation and power Control at physical layer is studies in [2]. Study related to QoS control with joint channel scheduling is presented [24]. Moreover joint rate control, adaptive routing, channel scheduling and link rate adaptation are studied in [25] and [26].

The frameworks which are based on generic cross layer design are considered in detail for further analysis as bellow.

**Cross-Layer Signaling Shortcuts-CLASS.** The cross-layer signaling design framework suggests in [4] is called Cross-Layer Signaling Shortcuts (CLASS). CLASS proposes direct signaling between non-neighboring layers. Internal message format of CLASS is defined with the objective of supporting local adaptations. External information flow is based on standard ICMP and TCP/IP headers.

The direct signaling across the layers proposed by CLASS inherently has a very low latency. The mechanism is highly flexible, because any protocol or application at any layer can exchange context. So, wide range of adaptations can be supported. Internal signals are light weighted but the external messages are wrapped in either ICMP or TCP/IP headers so, it introduces some overhead. Hence, average signaling overhead of internal and external context exchange is moderate. However, direct interaction among the protocols introduce high design complexity and hence, maintenance difficulty. Moreover, CLASS proposal violates the concept of layered protocol stack by direct signaling among layers for performance objectives.

**Cross-layer Coordination Planes.** A framework based on cross-layer coordination planes for wireless terminals is proposed [27]. Cross-layer coordination model composed of four coordination planes where each of them is a cross-section of layered-protocol stack. The planes are classified according to the functionality as security, QoS, mobility, and wireless link. Internal details of signaling and interactions are not available.

Coordination planes separate the wireless networking problems from the existing functionality of the layered

stack hence ensures uninterrupted operation to existing stack. So this concept has high degree of coexistence in the existing layered stack. Due to the cross-section views introduced as coordination planes the modularity of the proposal is low. Implementation and changes to the existing protocols and proving operations of planes is complex. Similarly maintenance is also difficult due to the complexity in changing and evolving the protocols. Flexibility is low since the adaptations are limited to the once defined in coordination planes. Moreover, the system cannot support adaptations which may involve interaction among the planes and scalability is low.

**WIreless DEployable Network System-WIDENS.** The WIreless DEployable Network System (WIDENS) [28, 29], is a ad-hoc communication system specifically designed for public safety or emergency applications. WIDENS architecture supports combination of several joint optimizations such as secured QoS extension for route optimization, mobility management, resource allocation at the MAC layer with hard QoS support, combine opportunistic scheduling and channel coding, slotted multiuser/stream capability.

WIDENS cross-layer architecture preserves modularity to a great extent, by allowing layer by layer interaction. The cross-layer interaction is separated from non-cross layer information flow, so the solution can coexist with the existing layered protocol stack. However, providing mapping function with the separated standard protocol functionality is complex and demands synchronization mechanisms. Further, to support wide range of adaptations it demands complex and considerable amount of changes to the protocol stack. So, design complexity is high introducing difficult maintainability. The processing overhead of context passes to the next layer is very high due to mapping of state information and parameters of adjacent layers. In addition to that, latency of layer-by-layer traversal and processing at each layer is very high. However, unnecessary and unintended cross-layer operations are avoided by controlling information flow through the translation at each layer. Flexibility of the architecture to support range of adaptation is low. Each newly added adaptation requisite changes to whole protocol stack that the packets flow through.

**ECLAIR Cross-layer Architecture.** ECLAIR architecture proposed [8], is based on the fact that protocol behavior is determined by the protocol data-structure. ECLAIR provides an interface to read and update the protocol data-structures through the interface called a Tuning Layer (TL) for each layer. TL is further divided in to generic tuning layer nad implementation specific tuning layer for portability objectives of implementation. Cross layer feedback algorithms and data structures are added in to Protocol Optimizers (PO). The collection of POs forms the Optimizing SubSystem (OSS).

ECLAIR cross-layer architecture is separated and can be easily enabled/ disabled it facilitates the uninterrupted operation to the layered protocol stack. Modularity of the system is high because it allows layer separation and preserves the modular functionality. Cross-layer interaction can be facilitated at any layer, and the solution can be extended to range of adaptations and optimizations through OSS. So, the scalability of the architecture is high. However, the design involves changes to almost every protocol that uses context as well as providing the context. So, maintenance and management of product is difficult and it hindered the evolution. In addition to that, there exists an extra complexity in implementing TLs and POs. Further, TLs and POs add extra signaling overhead and latency.

**Cross-Layer Decision Support Based on Global Knowledge-CrossTalk.** A cross layer architecture called CrossTalk for decision support based on global knowledge is proposed [9]. CrossTalk enables mobile devices to establish the state of the mobile node as a *local view* and relative status called *global view* compared to global network conditions. Local view is represented as the sum of local parameters such as battery level, SNR, location information, transmit power, etc. Global view is based on the metrics such as energy level, communication load or neighbor degree. The CrossTalk architecture consists of two data management entities to manage aforementioned two views. CrossTalk proposes local adaptations of the mobile device based on the global status. Global view is encouraged to use whenever possible to have network wide accurate decisions.

CrossTalk proposes a comprehensive network wide decision mechanism. The architecture can coexist with the layered architecture with uninterrupted operation to the stack. However, CrossTalk does not address the local view in detail for example how the local parameters are acquired by the local view management entity and how they are exchanged to the interested protocols. Further, establishing a global view and data dissemination is costly and complex. Solution is less flexible in local adaptations and performance improvements because of the lack of support for local adaptations. Latency and overhead is high due to complex network wide data dissemination procedure. Local data accessibility and dissemination procedure is not addressed and information about modularity of signaling mechanism is not available.

**Local Server based Cross-Layer Coordination Framework.** A cross-layer coordination framework which consists of a local cross-layer coordination server and clients at each layer is suggested in [30]. Non-adjacent layer interaction is done through the cross-layer server. Context delivery is performed in a way that, when an initiating layer wants to send a certain event to another target layer, the client of the initiating layer first sends event to the server, and then the server forwards it to the target layer. How the interested cross-layer protocols

and applications can express interest for context is not addressed. A parameter repository is maintained at the server.

The framework preserves the modularity while maintaining a higher degree of flexibility by allowing interaction among non-adjacent layers. Since the cross-layer interactions are separated from the standard operational protocol stack, coexistence of the framework with the layered stack is high. However, since the layers that support the parameters also need to be changed and all the adaptations are maintained at the layer client itself, the design complexity of the framework is high. Since the parameters traverse through server and client are kept in a repository rather than notifying the interested layers as an when the event occurs, there is a latency of the signaling. Signaling overhead is low because the event structure composed of few fields.

### 2.2 Context Aware Adaptations

A general definition of context for context aware computing domain is presented in [31]. According to the definition provided in [31] context is: ***any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves.***

Challenges in context aware computing, include uncertainty, diversity and complexity of context information. Research had been tried to investigate important aspects of context aware computing such as context discovery, context presentation, and context execution including reasoning to address aforementioned challenges [32, 33, 34, 35].

**Context Modeling.** Context representation and modeling has been addressed various domain specific requirements to describe context. Mark-up based models extend existing web standards such as XML [1] and RDF [2] to represent contextual information. Composite Capability/Preference Profiles (CC/PP) [3] is a data representation mark-up format based on RDF, which is proposed to describe user agent and proxy capabilities and preferences. The Comprehensive Structured Context Profiles (CSCP) context model which overcomes structural shortcoming of CC/PP is proposed in [36]. XML based context representation is proposed in [37]. XML is used to encode context configurations and values, and XML associated tree structure and XML schema are used to represent richer data and meta-data. A context modeling approach called Context Spaces which describe context and situations with spatial metaphors of state and space is presented in [38]. Context Space explicitly models context and situations in general rather representing specific contextual information.

[1] urlhttp://www.w3.org/XML/

[2] urlhttp://www.w3.org/TR/1999/REC-rdf-syntax-19990222/

[3] urlhttp://www.w3.org/TR/2001/WD-CCPP-struct-vocab-20010315/

## 3 Context Representation and Support for Adaptations

### 3.1 Motivation

A number of research studies has been done in specific cross-layer adaptations as discussed in Section 2. Almost none of studies address the context awareness in cross-layer signaling in great detail. None of the studies present a clear, generic way to model context parameters. None analyses the context at various layers of the protocol stack relation to each adaptation.

The generic architectures address the context exchange and signaling mechanisms and very few studies try to address context acquisition including [4, 39]. To facilitate context awareness it is necessary to investigate other aspects of context awareness mentioned in 2.2 in addition to the signaling mechanisms (context exchange) addressed in the cross-layer architectures. Generic mechanisms for context presentation and context acquisition are key aspects which are missing in the generic cross-layer architectures proposed in literature. In addition to that, clear definition of context parameters used in specific adaptations and applications is necessary in order to facilitate control over multiple adaptations to avoid performance degradation of wholes system.

### 3.2 Classification of context parameters

A detail classification of context parameters used in various cross-layer context aware adaptations discussed in Section 2 is presented in this section. Table 1 shows the identified layer parameters related to adaptations at each layer. This parameter identification is based on the review of the specific adaptations found in the related work.

### 3.3 Context Aware Architecture

We propose the CA3RM-Com architecture [44] as a generic cross-layer context aware framework. The architectural details, design principles are discussed in [44]. The CA3RM-Com architecture and its modular components are illustrated in Figure 1. The CA3RM-Com architecture composed of several components to facilitate aspects of context awareness. The components are Context Exchange Module (CEM), Context Acquisition Module (CAM), Context Representation Module (CRM) and a Context Management Module (CMM). Context exchange across the protocol stack and across the network is carried out through the CEM which is called ConEx [45]. ConEx is an event driven

**Table 1. Context Parameters of Layer Adaptations**

| Adaptation | Layer | Parameter |
|---|---|---|
| **Application Layer Adaptations** | | |
| QoS Control ( [25], [2], [26], [24] ) | Application Layer | user application priorities, user QoS preferences (delay sensitivity, loss tolerance), Source distortion, packet loss rate, video source coding, user satisfaction (MOS), PSNR |
| | Link Layer | packet Size |
| | Physical Layer | modulation, code rate, symbol rate |
| Packetization ( [21], [11], [20], [40] ) | Application Layer | QoS (delay sensitivity) |
| | Data Link Layer | channel rate, multiple access, network access delay, error detection (retry limits, frame length, BER) |
| | Transport Layer | hybrid ARQ error control |
| | Physical Layer | modulation, antenna diversity, power delay profile, time delay profile, time delay speed, transmitter signal power, maximum adaptation frequency, battery power |
| **Transport Layer Adaptations** | | |
| Mobility Adaptations ( [41] ) | Network Layer | hand over notifications |
| Congestion Control/Rate Control/ Error corrections ( [14], [10], [16], [25], [26], [40] ) | Application Layer | service quality, source bit rate |
| | Transport Layer | Congestion distortion, receiver window, timeout clock, congestion window, TCP/UDP header checksum, TCP/UDP header options, serial number of corrupted packet |
| | Network Layer | route data( route failures, route changes) |
| | Data Link Layer | SNR, BER, error coding, channel conditions (channel access delay, congestion) |
| **Network Layer Adaptations** | | |
| Adaptive routing ( [14], [13], [10], [25], [26], [22], [21], [23], [17], [19], [12], [18] ) | Application Layer | traffic type, delay bound, transmission delay jitter bound |
| | Network Layer | routing metrics, route outage probability, number of nodes in routes, network packet size(routing protocol), bit rate |
| | Data Link Layer | link outage probability, network congestion, packet delay, link state routing, average SNR, SNR threshold |
| | Physical Layer | battery power, min transmission power, path loss exponent, transmission range |
| Mobility Management ( [42], [43] ) | Application Layer | Application/User QoS requirements |
| | Data Link Layer | Link layer hand over triggers |
| **Data Link Layer Adaptations** | | |
| Scheduling and Adaptive Error Control ( [14], [10], [15], [16], [25], [26], [24], [40] ) | Application Layer | service quality |
| | Network Layer | routing data (route failures, changes) |
| | Data Link Layer | SNR, link transmission rate, packet size/length, symbol rate, constellation size, error control system, channel conditions (packet loss, sequence number of packets), network delay, congestion(queue length, average link layer utilization), link BW, PER,RTT, Time slots, queue of packets per user, partial checksum |
| | Physical Layer | channel conditions (equalizer information -fading.), battery power |
| Channel/Rate adaptation ( [25], [20], [23], [17], [12], [18] ) | Application Layer | transmission rate |
| | Data Link Layer | SNR, BER, error detection (retry limits, frame control), BW, link capacity, outage probability of links, link transmission rates |
| | Physical Layer | interference, SNR, noise, fading |
| **Physical Layer Adaptations** | | |
| Adaptive Modulation/Transmission mode ( [25], [26], [22], [2] ) | Application Layer | service quality |
| | Network Layer | Routing data/traffic, network data rate |
| | Data Link Layer | SNR, payload data, |
| | physical layer | mode, Channel fading, channel code rate, modulation, bytes per symbol, BS-user gain, transmit power, SINR |
| Congestion Recognition | Physical Layer | load estimation intra-cell interference, Base station transmit power |
| Power Control ( [10], [2], [11], [23], [19] ) | Data Link Layer | angle of arrival (AOA) of RTS, CTS, transmission rate |
| | Physical Layer | energy usage (CPU, network ) |

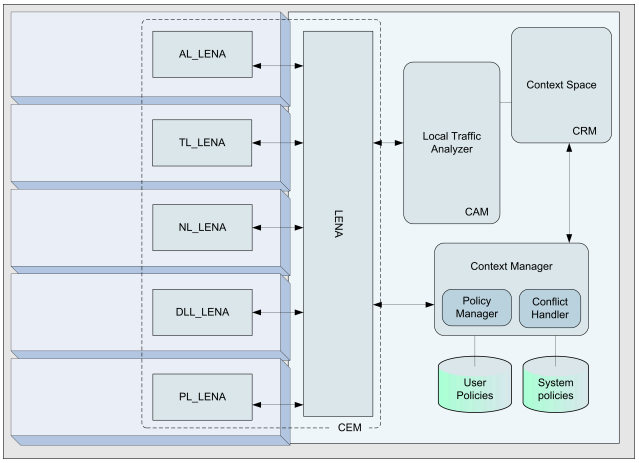



**Figure 1. Cross-layer Context Aware Architecture**

context exchange framework in which context delivery is based on subscriptions. Details about ConEx, the algorithms and message formats are presented in [45]. Modeling of ConEx is presented in [46]. Light weighted message format of ConEx ensures low overhead of context exchange mechanism. Event driven context exchange through subscriptions and notifications facilitated by ConEx ensures low latency in context delivery. ConEx preserves modularity of the protocol stack by enabling cross-layer signaling through layer agents and by restricting the direct interaction across protocols at non adjacent layers. Context acquisition is accomplished through the Local Traffic Analyzer CAM, which sniffs the packets flow through the protocol stack. Local packet analyzer is utilized in context acquisition to minimize changes to the existing protocols during the process of acquiring the context and to introduce the uninterrupted functioning of non-cross layering protocols in the existing stack. CAM exchanges context via ConEx. Context is represented using context space which is a generic representation of situations which consists of context parameters at each layer and performance parameters. Moreover, CA3RM-com supports local and global context awareness through it ConEx module. Context Manager enables policy based system driven adaptations and controls adaptations to avoid conflicts which would consequence performance degradations. The architecture is flexible and can support adaptations ranging from application adaptations to channel adaptations. Architecture can be easily enabled and disabled hence ensures higher degree of coexistence with the existing layered protocol stack.

### 3.4 Context Representation Module

Context representation is a key aspect in any context aware system. CA3RM-com architecture exploits extended multi-layer version of Context Space [38] to represent context parameters and performance optimizing metrics used in cross-layer adaptations.

Figure 2 illustrates the representation of context parameters and performance parameters in Euclidean vector space for a given problem domain. These set of parameters represent a situation. Combination of context and performance parameters (could be static or dynamic) form the context vector of a particular situation.

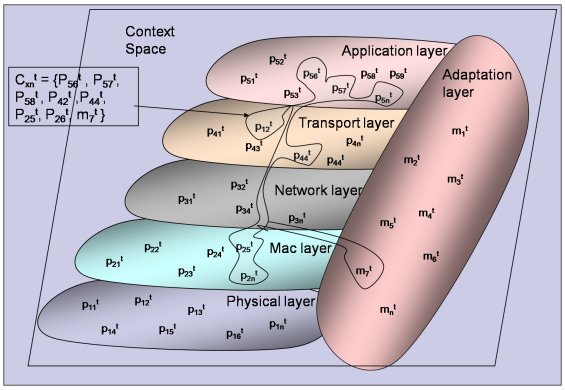



**Figure 2. Multi-layer Context Representation**

Context vector corresponding to a given situation at time t $v_t$, can be represented as a vector consists of a set of context parameters (cp) and set of performance parameters (pp) as shown in Equation 1.

$$V_t = \sum_{i=1}^{n} a_i cp_{xit} + \sum_{j=1}^{m} b_i pp_{it} \quad (1)$$

Where, $a_i$ , $b_i$ are scalars.

x indicates the layer number 1 to 5, which represent the physical, mac, network, transport, and application layers of the practical protocol stack.

$cp_{xnt}$ is the $n^{th}$ adaptation parameter at layer x at time t.

$pp_{nt}$ is the $n^{th}$ performance parameter at time t.

So, context vector at time t, can be written as shown in Equation 2

$$V_t = a_1.cp_{11t}... + a_n.cp_{5nt} + b_1.pp_{1t}... + b_m.pp_{mt} \quad (2)$$

Further, granularity is introduced to dynamic parameter ranges in order to provide reasoning and more reliable decision making about the situation as shown in Table 2. For example, parameter $cp_{11t}$ context space values range from $r1_l cp_{11t}$ to $rn_u cp_{11t}$. where, $r1_l$ represents lower bound and

$rn_u$ represents the upper bound. The context value range is subdivided in to n number of ranges.

**Table 2. Context parameter value ranges**

| Range | Parameter values |
|---|---|
| $range_1$ | $r1_l cp_{11t}$ - $r1_u cp_{11t}$ |
| $range_2$ | $r2_l cp_{11t}$ - $r2_u cp_{11t}$ |
| . | . |
| . | . |
| . | . |
| $range_n$ | $rn_l cp_{11t}$ - $rn_u cp_{11t}$ |

### 3.5 Context Management Module

Context Management Module (CMM) in CA3RM-Com architecture executes two major tasks. Firstly, Context Manager (CM) ensures that the context aware adaptations are based on predefined user and system policies through the Policy Manager (PM). Secondly, CMM controls context aware adaptations to avoid unintended conflicts that may arise by uncontrolled adaptations. This is done through the Conflict Handler. Two main categories of adaptation are considered in the proposed CA3RM-com architecture as illustrated in Figure 3.

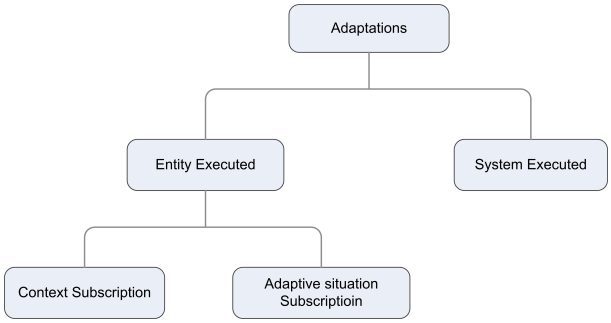



**Figure 3. Categorization of Adaptations**

The two categories of adaptations, *entity-executed* adaptations and *system-executed* are supported in the architecture. In *entity-executed* adaptations the *entity* (the term entity is used to represent any protocol or application at any layer of the protocol stack) which is executing the adaptation subscribes to the architecture to acquire the context.

Entity-executed adaptation can be achieved based on two types of subscriptions. In one type of entity-executed adaptations the entity requests particular context parameters in order to make the adaptation decision based on its own rules or policies and conditions. In the other type of entity-executed adaptations, entitys subscription is to an adaptive situation, where the policy manager executes the policies related to the adaptation and notifies the adaptation decision to the relevant entity [1]. In system-executed adaptations, the entity is not involved in subscriptions but the context manager forces the adaptation to the entity, based on system and user defined policies which enables control and administration of the system [1].

## 4 Simulation

### 4.1 Adaptive Handover

We have evaluated the context aware adaptive Multi-homed Mobile IP [47] handover mechanism based on the proposed CA3RM-Com architecture in simulation. The detail discussion of algorithm and simulation of handover scenario is out of scope of this article. In this section we discuss the simulation in brief and the extended experiments with multiple candidate networks to provide a validated example of adaptation that CA3RM-Com can support. In brief, context aware adaptive Multi-homed Mobile IP handover is an adaptive mechanism is an entity-executed adaptation (as discussed above), where the adaptation decision is done by the MIP protocol itself based on the subscribed context parameters. We show overall handover delay can be minimized by fast agent discovery and fast move detection and context aware decision for adaptation to mobility can be made. Hence increased throughput and minimized packet loss is achieved through fast handover. In context aware adaptive handover, fast movement detection was done using SNR based movement prediction without waiting for the conventional unreachability detection. The least congested GW is selected based on RNL metric [48].

The extended simulation presented here is based on the topology shown in Figure 4 with multiple candidate networks, where more than one possible approaching gateways are available for the MN for handover. Results are presented as mean value of multiple simulations with different seeds to use normal distribution. Results are presented with 90% confidence level.

Context space discussed in section previously is used as shown in Figure 5. The context vector for adaptive handover is shown in Equation 3 and the range of values of the parameters in Table 3. Granularity of SNR and RNL is shown in Table 4.

$$\begin{aligned} V_{ah} := a_1.snr + a_2.snr_{th} + a_3.snr_{cth} + a_4.ctp \\ +a.5.cps + a_6.cbf + a_7.cdr + a_8.rnl \\ +a_9.aaf + b_1.pl + b_2.thr \end{aligned} \quad (3)$$

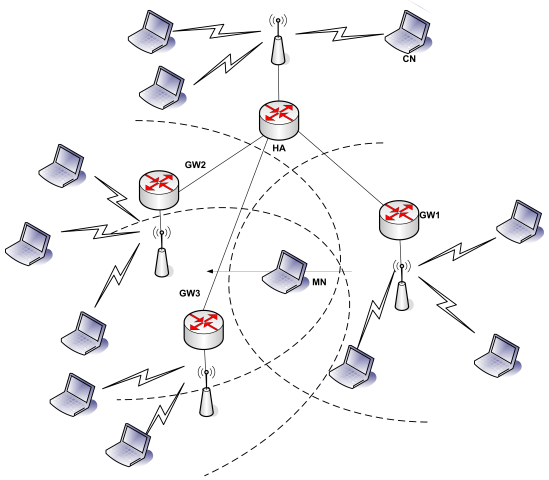



**Figure 4. Simulation Network Topology**

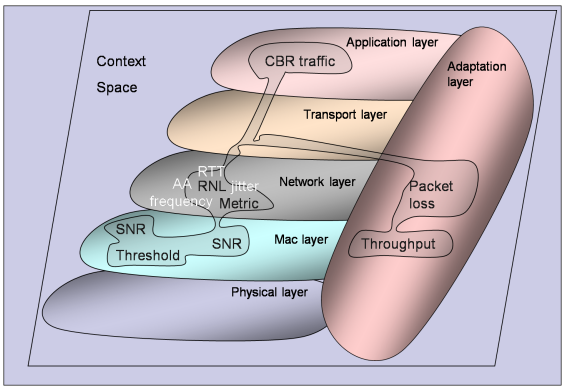



**Figure 5. Context Space for context aware handover**

**Table 3. Context Parameters of Adaptive Handover simulation**

| Symbol | Parameter | Range of Values/Value |
|---|---|---|
| $V_{ah}$ : Adaptive Handover | | |
| snr | SNR of agent advertisements | >10dB |
| $snr_{th}$ | radio receiver SNR threshold | 10dB |
| $snr_{cth}$ | CBR traffic SNR threshold | 15dB |
| ctp | CBR traffic priority | 0/1 |
| cps | CBR packet size | ( 1460 ) Bytes |
| cbf | CBR packet interval | (0.003-0.011) second |
| cdr | CBR data rate | (4-4.2)Mbps |
| rnl | RNL metric | 0-1 |
| aaf | agent advertisements frequency | 2 per second |
| pl | CBR pakcet loss rate | (0 - 1)% [4] |
| thr | CBR throughput | (3.96 - 4.22) bits per second |

**Table 4. Parameter ranges of SNR and RNL**

| SNR | |
|---|---|
| **SNR Range (dB)** | **Relationship to channel quality** |
| <15 | very poor |
| 15 to 20 | poor |
| 20 to 30 | good |
| >30 | excellent |
| RNL | |
| **RNL** | **Relationship to congestion** |
| 0.1 to 1 | poor |
| .01 to 0.1 | good |
| 0 to 0.01 | excellent |

SNR is Signal to Noise Ratio of received agent advertisement in multi-homed MIPV4 or binding updates of Multi-homed MIPV6. Context parameters such as RTT, jitter and frequency of agent advertisement are used to calculate RNL metric. Simulation of the proposed solution was carried out using the network simulator Glomosim. Time out for bindings used is three times the agent advertisement time. Simulation was carried out for 200 seconds. Constant Bit Rate (CBR) traffic flows were sent from MN to CN every 3MS. Results of different data rates with different packet sizes were simulated. Agent advertisements in the MIP were sent every half a second and MN registered every third advertisement with the HA.

Pure M-MIP approach, which does not use context exchange for handover decision is represented as *WithOut ConEx* (WOConEx) approach. Adaptive handover decision and is based on ConEx architecture is referred to as the *ConEx* (ConEx) approach.

Figure 6 illustrates the packet loss rate of CBR traffic with variable data rates. Due to the delay of move detection in *WOConEx* approach a considerable packet loss is noticed. In *ConEx* approach the move detection delay is zero with the proactive move detection technique. So the packet loss rate is zero in *ConEx* approach. The graph in Figure 7 illustrates the throughput results of CBR traffic for variable data rates. In the *WOConEx* approach, there exists a delay for move detection, which causes the packet loss. Due to this packet loss during the handover in this approach, the throughput is decreased. In *ConEx* approach the move detection is done proactive without a delay. Hence to-

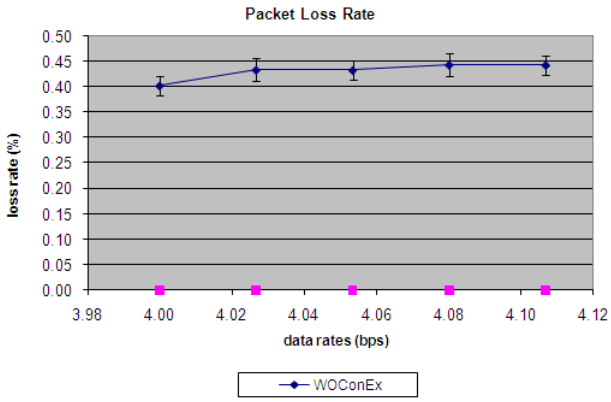



**Figure 6. Packet loss rates of CBR traffic**

tal handover delay is reduced and maximum throughput is available in *ConEx* approach compared to *WOConEx* approach.

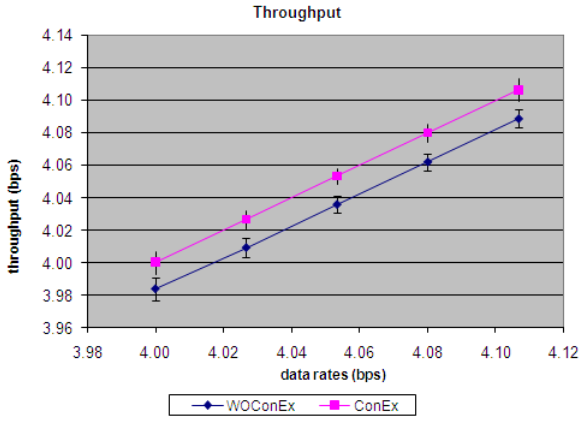



**Figure 7. Throughput results of CBR traffic**

## 4.2 Context Representation Extensibility

The proposed CA3RM-Com architecture is extensible to support various single layer and multi layer context aware adaptations. These adaptation techniques can be entity-executed or system-executed based on the application and the policy managers system policies.

**User Perceived Quality Maximization.** CA3RM-Com architecture can be used in user perceived quality maximization discussed in [24]. The context vector corresponds to the adaptive quality maximization of video applications is shown in Equation 4. Static parameter values and ranges of parameter values for dynamic parameters of context space are shown in Table 5. MOS values between 0 to 4.4 are used to represent user satisfaction.

$$V2 : upqm = a1.m + a2.cr + a3.smr + a4.snr + b1.mos + b2.us + b2.abs + b1.psnr + b2.sl + b3.us \quad (4)$$

**Table 5. User Perceived Quality Maximization**

| Symbol | Parameter | Range of Values/Value |
|---|---|---|
| V2upqm - User Perceived Quality Maximization of video | | |
| m | Modulation scheme | DBPSK, BPSK, QPSK, 16-QAM, 64-QAM, DQPSK |
| cr | Channel Code rates | 1/2,4/3,2/3 |
| smr | Symbol/modulation rate | (500, 700,900)kSymbolds/s |
| snr | Signal to Noise Ration (SNR) | (7-25) dB |
| mos | Mean Opinion Score (MOS) | 0-4.4 |
| psnr | percieved SNR (PSNR) | (50-200)kbits/s |
| sl | slice losses | 0-15% |
| us | user satisfaction | very satisfied, satisfied, some users dissatisfied, Many users dissatisfied, Nearly All users dissatisfied, Not recommended. |

**TCP congestion & flow control.** TCP congestion and flow control mechanism is proposed in [49]. The context vector representation of this adaptation in CA3RM-Com architecture we propose is shown in Equation 5. Static parameter values and ranges of parameter values for dynamic parameters of context space are shown in Table 6.

$$V3 : tcp = a1.up + a2.bw + a3.rtt + a4.aw + b1.thr \quad (5)$$

**Energy Optimized Routing.** Context modeling proposed in this article can be extended in energy optimized routing as an adaptation [23, 19]. The context vector representation of this adaptation is shown in Equation 6. Static parameter values and ranges of parameter values for dynamic parameters of context space are shown in Table 7.

$$V4 : eor = a1.ps + a2.nn + a3.br + a4.es + a5.tr + a6.ple + a7.mtp + b1.thr \quad (6)$$

**Optimal Transmission Mode.** Link adaptation/optimal transmission mode for IEEE802.11a is proposed in [15].

**Table 6. TCP congestion and flow control**

| Symbol | Parameter | Range of Values/Value |
|---|---|---|
| V3:tcp - TCP congestion and flow control | | |
| up | User priorities | 1,2,3 |
| bw | Network BW | 100Mbps |
| rtt | Round Trip Time | about 5ms |
| aw | Advertised window | 8 KB(for transmission $<$ 1 Mbps), 17 KB(for transmission 1-100 Mbps), 64 KB (for transmission $>$ 100 Mbps) |
| thr | throughput | (500-1060) kbps |

**Table 7. Energy Optimized routing**

| Symbol | Parameter | Range of Values/Value |
|---|---|---|
| V4:eor - Energy Optimized routing | | |
| ps | Network packet size | 512B |
| nn | number of nodes | 10-25/25 |
| br | bit rate | 2Mbps |
| es | Energy Saving | (0-100)% |
| tr | Transmission range | 150-250m |
| mtp | Min transmit power | 280mW |
| ple | Path loss exponent | 4 |
| thr | throughput | (0-200)kbps |

The context vector representation of adaptive transmission mode is shown in Equation 7. Static parameter values and ranges of parameter values for dynamic parameters of context space are shown in Table 8.

**Table 8. Optimal transmission mode**

| Symbol | Parameter | Range of Values/Value |
|---|---|---|
| V5:otm - Optimal transmission mode | | |
| tm | Transmission mode | 1, 2, 3 ,4 ,5, 6, 7, 8 |
| snr | SNR region (dB) | 0-5(Not used) , 5-9, 9-11, 11-15 , 15-20, 20-21, 21-30 |
| cr | Code rate (FEC) | 1/2, 3/4, 1/2, 3/4, 1/2, 3/4, 2/3, 3/4 |
| bps | bytes per OFDM symbol | 3, 4.5,6,9,12,18,24,27 |
| dr | Network Data rate (Mbps) | 6, 9, 12, 18, 24, 36, 48, 54 |
| m | modulation | BPSK, BPSK, QPSK, QPSK, 16-QAM, 16-QAM, 64-QAM, 64-QAM |
| pl | payload data | 16000 bits |
| thr | throughput | (0-200)kbps |

$$V5 : otm = a1.tm + a2.snr + a3.cr + a4.bps + a5.dr + a6.pl + a7.m + b1.thr \quad (7)$$

## 5 Conclusion

Cross-layer adaptations proposed in the literature are classified based on the layer in which the adaptation is executed. The context parameter set for the adaptations, relevant to each layer is identified. We have presented context modeling mechanism for cross-layer context aware adaptations in proposed CA3RM-Com architecture. CA3RM-Com is the generic architecture proposed to support adaptation through the multilayer context exchange based on interest, maintaining the system modularity. Various adaptations and relevant context representation that CA3RM-Com can support are discussed. Context-aware adaptive handover is used to illustrate the context modeling. The extended simulation of context aware adaptive Multi-homed Mobile IP handover and the performance improvements of the simulation results are discussed. Identification of dependency relationships and conflicts among adaptation parameters with the objective of system stability is a open research area to be addressed in future work.

## References

[1] R. Kodikara, A. Zaslavsky, and C. Åhlund, "Towards Context Aware Adaptation in Wireless Networks," in *The Second International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2008)*. Valencia, Spain: IEEE Computer Society, Oct. 2008, pp. 245 – 250.

[2] M. Van Der Schaar and N. Sai Shankar, "Cross-layer Wireless Multimedia Transmission: Challenges, Principles, and New Paradigms," *IEEE Wireless Communications, Special issue Advances in Wireless Video*, vol. 12, no. 4, pp. 50–58, Aug. 2005.

[3] H. Zimmermann, "OSI Reference Model–The ISO Model of Architecture for Open Systems Interconnection," *IEEE Transactions on Communications, Special issue on Computer Network Architectures and Protocols*, vol. 28, no. 4, pp. 425 – 432, April 1980.

[4] Q. Wang and A. Abu-Rgheff, M, "Cross-Layer Signalling for Next Generation Wireless Systems," in *The 2003 IEEE Wireless Communications and Networking (WCNC 2003)*, vol. 2, Louisiana, USA, March 2003, pp. 1084 – 1089.

[5] J. Stine, "Cross-Layer Design of MANETs: The Only Option," in *The 2006 Military Communications Con-*

*ference (MILCOM 2006)*, Washington, DC, USA, Oct. 2006, pp. 1–7.

[6] Z. Haas, "Design Methodologies for Adaptive and Multimedia Networks," *IEEE Communications Magazine*, vol. 39, no. 11, pp. 106–107, Nov. 2001.

[7] V. Srivastava and M. Motani, "Cross-Layer Design: A Survey and the Road Ahead," *IEEE Communications Magazine*, vol. 43, no. 12, pp. 112 – 119, Dec. 2005.

[8] V. Raisinghani and S. Iyer, "Cross Layer Feedback Architecture for Mobile Device Protocol Stacks," *IEEE Communications Magazine*, vol. 44, no. 1, pp. 85 – 92, Jan. 2006.

[9] R. Winter, J. Schiller, N. Nikaein, and C. Bonnet, "CrossTalk: Cross-Layer Decision Support Based on Global Knowledge," *IEEE Communications Magazine*, vol. 44, no. 1, pp. 93 – 99, January 2006.

[10] X. Lin, N. Shroff, and R. Srikant, " A Tutorial on Cross-Layer Optimization in Wireless Networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, pp. 1452 – 1463, Aug. 2006.

[11] W. Wang, D. Peng, H. Wang, and H. Sharif, "A Cross Layer Resource Allocation Scheme for Secure Image Delivery in Wireless Sensor Networks," in *The 2007 International conference on Wireless communications And Mobile Computing (IWCMC 2007)*. Honolulu, Hawaii, USA: ACM, Aug. 2007, pp. 152–157.

[12] M. Park, J. Andrews, and S. Nettles, "Wireless Channel-Aware Ad Hoc Cross-Layer Protocol With Multi-Route Path Selection Diversity," in *The IEEE 58th Vehicular Technology Conference (VTC 2003)*, vol. 4. Orlando, Florida, USA: IEEE, Oct. 2003, pp. 2197 – 2201.

[13] N. Yang, R. Sankar, and J. Lee, "Improving Ad Hoc Network Performance Using Cross-Layer Information Processing," in *The 2005 IEEE International Conference on Communications (ICC 2005)*, vol. 4, Seoul, Korea, May 2005, pp. 2764 – 2768.

[14] M. Conti, G. Maselli, and S. Giordano, "Cross-Layering in Mobile Ad Hoc Network Design," *Computer*, vol. 37, no. 2, pp. 48 – 51, Feb. 2004.

[15] M. Realp, A. Perez-Neira, and C. Mecklenbrauker, "A Cross-Layer Approach to Multi-User Diversity in Heterogeneous Wireless Systems," in *The 2005 IEEE International Conference on Communications (ICC 2005)*, vol. 4, Seoul, Korea, May 2005, pp. 2791 – 2796.

[16] S. Shakkottai, T. Rappaport, and P. Karlsson, "Cross-layer Design for Wireless Networks," *IEEE Communications Magazine*, vol. 41, no. 10, pp. 74–80, Oct. 2003.

[17] W. Yuen, H. Lee, and T. Andersen, "A Simple and Effective Cross Layer Networking System for Mobile Ad Hoc Networks," in *The 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 2002)*, vol. 4, Lisboa, Portugal, September 2002, pp. 1952 – 1956.

[18] B. Chen, M. Lee, and S. Yi, "A Framework for Cross-layer Optimization from Physical Layer to Routing Layer on Wireless Ad Hoc Networks," in *The 50th annual IEEE Global Communications Conference (IEEE GLOBECOM 2007)*, Washington, DC, USA, Nov. 2007, pp. 3678 – 3683.

[19] V. Bhuvaneshwar, M. Krunz, and A. Muqattash, "CONSET: A Cross-Layer Power Aware Protocol for Mobile Ad Hoc Networks," in *The 2004 IEEE International Conference on Communications (ICC 2005)*, vol. 7, Paris, France, June 2004, pp. 4067–4071.

[20] B. D. Noble, M. Satyanarayanan, D. Narayanan, J. E. Tilton, J. Flinn, and K. R. Walker, "Agile Application-Aware Adaptation for Mobility," in *Proceedings of the sixteenth ACM symposium on Operating systems principles (SOSP '97)*. Saint Malo, France: ACM, 1997, pp. 276–287.

[21] B. Rivera, M. Humphrey, and C. Todd, "Asynchronous Feedback of Network Capacity for Application Layer Tuning," in *The 1998 IEEE Military Communications Conference (MILCOM 98)*, vol. 3. Boston, Massachusetts, USA: IEEE, Oct. 1998, pp. 946 – 950.

[22] S. Merigeault and C. Lamy, "Concepts for Exchanging Extra Information Between Protocol. Layers Transparently for the Standard Protocol Stack," in *The 10th International Conference on Telecommunications (ICT'2003)*, vol. 2, Papeete, French Polynesia, February-March 2003, pp. 981 – 985.

[23] S. Doshi, S. Bhandare, and T. Brown, "An on-demand minimum energy routing protocol for a wireless ad hoc network," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 5, no. 1, pp. 50–66, July 2001.

[24] S. Khan, S. Duhovnikov, E. Steinbach, M. Sgroi, and K. W., "Application-driven Cross-layer Optimization for Mobile Multimedia Communication using a Common Application Layer Quality Metric," in *The 2006 international conference on Wireless communications and mobile computing (IWCMC 2006)*. Vancouver, Canada: ACM, July 2006, pp. 213 – 218.

[25] E. Setton, T. Y., X. Z., A. Goldsmith, and B. Girod, "Cross-layer Design of Ad Hoc Networks for Real-Time Video Streaming," *IEEE Wireless Communications*, vol. 12, no. 4, pp. 59–65, Aug. 2005.

[26] S. Ci, H. Wang, and D. Wu, "From Heuristic to Theoretical: a New Design. Methodology for Cross-Layer Optimized. Wireless Multimedia Streaming," in *The 16th IEEE International Packet Video Workshop*, Lausanne, Switzerland, Nov. 2007, pp. 228 – 233.

[27] G. Carneiro, J. Ruela, and M. Ricardo, "Cross-Layer Design in 4G Wireless Terminals," *IEEE Wireless Communications*, vol. 11, no. 2, pp. 7–13, April 2004.

[28] R. Knopp, N. Nikaein, C. Bonnet, H. Aïache, V. Conan, S. Masson, G. Guibé, and C. L. Martret, "Overview of the Widens Architecture, A Wireless Ad Hoc Network for Public Safety," in *The 1st IEEE International Conference on Sensor and Ad Hoc Communications and Networks (IEEE SECON 2004)*, Santa Clara, USA, Oct. 2004.

[29] H. Aïache, V. Conan, G. Guibé, J. Leguay, C. Le Martret, J. M. Barcelo, L. Cerdà, J. García, R. Knopp, N. Nikaein, X. Gonzalez, A. Zeini, O. Apilo, A. Boukalov, J. Karvo, H. Koskinen, L. Bergonzi, J. Diaz, J. Meessen, C. Blondia, P. Decleyn, E. Van de Velde, and M. Voorhaen, "WIDENS: Advanced Wireless Ad-Hoc Networks for Public Safety," in *The 14th IST Mobile & Wireless Communications Summit*, Dresden, Germany, June 2005.

[30] K. M. El Defrawy, Z. M. S., and M. M. Khairy, "Proposal for a Cross-Layer Coordination Framework for Next Generation Wireless Systems," in *The 2006 International conference on Wireless communications and mobile computing (IWCMC 2006)*, Vancouver, British Columbia, Canada, July 2006, pp. 141 – 146.

[31] A. K. Dey, "Understanding and Using Context," *Personal Ubiquitous Computing.*, vol. 5, no. 1, pp. 4–7, 2001.

[32] J. Pascoe, "Adding Generic Contextual Capabilities to Wearable Computers," in *The second International Symposium on Wearable Computers (ISWC 1998)*, Pittsburgh, Pennsylvania, USA, Oct. 1998, pp. 92–99.

[33] F. Hohl, U. Kubach, A. Leonhardi, K. Rothermel, and S. M., "Next Century Challenges: Nexus - An Open Global Infrastructure for Spatial - Aware Applications," in *The 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM '99)*, Seattle, Washington, USA, Aug. 1999, pp. 249–255.

[34] M. Khedr and A. Karmouch, "ACAI: Agent-Based Context-aware Infrastructure for Spontaneous Applications," *Journal of Network and Computer Applications*, vol. 28, no. 1, pp. 19–44, 2005.

[35] A. Padovitz, S. W. Loke, and A. Zaslavsky, "The ECORA framework: A hybrid architecture for context-oriented pervasive computing," *Pervasive and Mobile Computing*, vol. 4, no. 2, pp. 182–215, April 2008.

[36] A. Held, S. Buchholz, and A. Schill, "Modeling of Context Information for Pervasive Computing Applications," in *The 6th World Multi Conference On Systemics, Cybernetics And Informatics, Symposium on Wearable Computers (ISWC 2002)*, Orlando, Florida, USA, July 2002, pp. 92–99.

[37] L. Capra, W. Emmerich, and C. Mascolo, "Reflective Middleware Solutions for Context-Aware Applications," in *The Third International Conference on Metalevel Architectures and Separation of Crosscutting Concerns*, vol. 2192, Kyoto, Japan, Sept 2001, pp. 126–133.

[38] A. Padovitz, S. W. Loke, and A. Zaslavsky, "Towards a Theory of Context Spaces," in *The Workshop on Context Modeling and Reasoning (CoMoRea), at 2nd IEEE International Conference on Pervasive Computing and Communication (PerCom ).* Orlando, Florida, USA: IEEE Computer Society, March 2004, pp. 38 – 42.

[39] V. Raisinghani and S. Iyer, "Architecting Protocol Stack Optimizations on Mobile Devices," in *The First International Conference on Communication System Software and Middleware (COMSWARE)*, vol. 4, New Delhi, India, Jan. 2006, pp. 1–10.

[40] L. Larzon, M. Degermark, and S. Pink, "UDP Lite for Real Time Multimedia Applications," 1999, preprint (1999), available at http://www.hpl.hp.com/techreports/1999/HPL-IRI-1999-001.pdf.

[41] R. Càceres and L. Iftode, "Improving the Performance of Reliable Transport. Protocols in Mobile Computing Environments," *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 5, pp. 850 – 857, June 1995.

[42] R. Brännström, R. E. Kodikara, C. Åhlund, and A. Zaslavsky, "Mobility Management for multiple diverse applications in heterogeneous wireless networks," in *The 3rd IEEE Consumer Communications and Networking Conference (CCNC 2006)*, vol. 2, Las Vegas, USA, Jan. 2006, pp. 818–822.

[43] L. Le and G. Li, "Cross-Layer Mobility Management based on Mobile IP and SIP in IMS," in *The 2007 International Conference on Wireless Communications, Networking and Mobile Computing (WiMob 2007)*, Shanghai, China, Sept. 2007, pp. 803 – 806.

[44] R. Kodikara, A. Zaslavsky, and C. Åhlund, *Wireless and Mobile Networking*. Boston: Springer, August 2008, vol. 284/2008, ch. Supporting Adaptive Real-time Mobile Communication with Multilayer Context Awareness, pp. 435–446.

[45] R. E. Kodikara, C. Åhlund, and A. Zaslavsky, "ConEx: Context Exchange in MANETs for Real time multimedia," in *The Fifth International Conference on Networking and the International Conference on Systems and International Conference on Mobile Communications and Learning Technologies (ICN/ICONS/MCL 2006)*. Mauritius: IEEE Computer Society, April 2006, pp. 70–78.

[46] R. Kodikara, S. Ling, and A. Zaslavsky, "Evaluating Cross-layer Context Exchange in Mobile Adhoc Networks with Colored Petri Nets," in *The 2007 IEEE International Conference on Pervasive Services (ICPS'07)*. Istanbul, Turkey: IEEE Computer Society, July. 2007, pp. 173–176.

[47] C. Åhlund, R. Brännström, and A. Zaslavsky, *Lecture Notes in Computer Science*. Berlin / Heidelberg: Springer, April 2005, vol. 3420/2005, ch. M-MIP: Extended Mobile IP to Maintain Multiple Connections to Overlapping Wireless Access Networks, pp. 204 – 213.

[48] C. Åhlund, R. Bränström, and A. Zaslavsky, "Running Variance Metric for evaluating performance of wireless IP networks in the MobileCty testbed," in *The First International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (Tridentcom)*, Italy, Feb. 2005, pp. 120–127.

[49] V. Raisinghani, A. Singh, and S. Iyer, "Improving TCP Performance over Mobile Wireless Environments using Cross Layer Feedback," in *The 2002 IEEE International Conference on Personal Wireless Communications (ICPWC-2002)*, New Delhi, India, Dec. 2002, pp. 81–85.

# Dynamic End-to-End QoS Provisioning and Service Composition over Heterogeneous Networks

N. Van Wambeke[1,2], F. Racaru[1,2], C. Chassot[1,2], and M. Diaz[1,2]
[1]*CNRS ; LAAS ; 7 avenue du colonel Roche, F-31077 Toulouse, France*
[2]*Université de Toulouse ; UPS, INSA, INP, ISAE ; LAAS ; F-31077 Toulouse, France*
*{van.wambeke, racaru, chassot, diaz}@laas.fr*

## Abstract

*This paper presents an approach that aims to provide Quality of Services (QoS) over heterogeneous Internet domains. The goal is to install and to manage QoS inside each domain and to ensure it on the end to end data path. QoS is a key requirement representing a significant infrastructure upgrade for future networks. Our proposal relies on the multi-service, multi-technology model based on Bandwidth Broker, entity in charge of controlling a given domain. It introduces a concept able to provide QoS in a set of domains using an inter domain signaling protocol as well as dynamic provisioning schemes that optimize resource usage. The approach is independent of the intra-domain routing protocol. Moreover it is independent of the underlying technology and imposes minimal constraints leaving a maximum degree of freedom for users and domains providers to implement specific internal solutions. The efficiency of the solution is shown through extensive simulations.*

***Keywords:*** *QoS, CAC algorithm, adaptive provisioning, resource optimization*

## 1. Introduction

The progress of new technologies during past years contributed to a development of new types of various applications. These applications, simultaneously multimedia and/or multi users, cover a large spectrum such as IP telephony, video on demand, streaming, tele-engineering, interactive games, peer to peer. Such applications need a special handling for their packets in order to work properly, thus they require new services other than those supported by the actual Internet. At the same time, current (and future) networks are deeply heterogeneous: from IP wired networks, to WI-FI, satellite, and UMTS. Moreover, each domain (that we assimilate later on to Autonomous System - AS) has independent policies in terms of services, security, admission control etc. As a result, the problem of end to end QoS must be addressed. QoS is targeted to support new applications and mechanisms. Therefore, QoS allows the deployment of richer services, potentially chargeable per user, thus it is a source of revenue, both for operators and for service providers. This analysis leads us to answer the next challenge: how to offer and control in the network the QoS required by applications (especially multimedia and real-time ones), taking into account the strong heterogeneity of the multi domain context Internet.

The convergence of services over the same IP infrastructure and the growth of the number of users lead the Internet community to research on standards and prototypes in order to implement these new technologies.

Actually, all packets receive the same treatment in network devices, without any differentiation. The need to offer new services demands a reconsideration of actual Internet packet treatment:

- new functionalities are needed in network devices (routers) in order to support service differentiation;
- new provisioning, resource management and admission control mechanisms are required on the data path;
- communication and cooperation between network equipments are required; these exchanges represent the signaling necessary to install and manage the quality of service on the data path.

In the following of this section, extending the work presented in [1], we present several proposals that aimed to deal with the QoS at different level on Internet architecture over IP networks.

### 1.1. Previous work

First IntServ [2]-[4] and then DiffServ [5]-[7] were proposed by the Internet Engineering Task Force (IETF - www.ietf.org) working groups to add QoS to best-effort Internet. Note that several contributions proposed a combined inter functioning architecture taking into account the advantages of each IntServ and DiffServ [8].

IntServ and Diffserv models have been used as a basis for several American and European projects such as QBone (http://qbone.internet2.edu) in USA or AQUILA (http://www-st.inf.tu-dresden.de/aquila), TEQUILA (www.ist-tequila.org), CADENUS (http://wwwcadenus.fokus.fraunhofer.de), MESCAL (www.mescal.org) and more recently EUQOS (www.euqos.eu) in Europe.

None of the above proposals answer to the QoS problem completely. The difficulty comes, on the one hand, from the variety of requirements of applications, and on the other hand, from the Internet structure, nowadays heterogeneous at several levels. By definition multi technology, Internet is now multi domain, each domain (AS) managing independently its resources and its services. Several issues need to be taken into account in this context:

- *provisioning*: the goal of service provisioning is to properly dimension of network resources accordingly to signed contracts (SLA) with different clients (individuals, companies, other providers);
- *admission control*: it aims to accept new traffic in the network without harming previous flows. Controlling the amount of traffic allows avoiding congestions and performance deterioration in the network;
- *signaling*: in a multi-domain Internet, it is imperative to install and manage QoS in each domain. Therefore a signaling across all domains on the data-path is needed.

The contributions which are described in this paper have been mainly performed within the EuQoS project [9], whose main goal was to develop a new QoS architecture over a multi-domain heterogeneous Internet. The key objective of EuQoS was to research, integrate, test and demonstrate end-to-end QoS technologies to support the infrastructure upgrade for advanced QoS-aware applications over multiple, heterogeneous network domains. The EuQoS system has been deployed on a Pan European test bed for the purposes of trial and validation.

More precisely, our contributions aim to offer QoS guarantees by coupling provisioning and admission control in a single signaling protocol. We propose an evolving architecture that decouples the service and the control plane that minimizes constraints for network administrators. We also propose a dynamic user centric provisioning, which allows optimizing the use of the most costly resources still satisfying the QoS demand.

The rest of the paper is organized as follows: the QoS architecture on which our proposition is built as well as related work are presented in Section 2. Section 3 details our dynamic end to end provisioning approach. Section 4 introduces the specifications of the signaling protocol that supports the process. These proposals are evaluated in Section 5 which also addresses scalability issues. Finally, conclusion and future work are given in Section 6.

## 2. QoS Architecture

### 2.1. General Considerations

Guaranteeing quality of service requires mastering the vastly distributed and dynamic nature of Internet and defining a new architecture and a set of new mechanisms. Our proposal is based on the following fundamental high level design rules:

- such a complex system cannot be provided without a high level design, i.e. without conceiving the architecture globally and without a strong coherence between all its sub-systems;
- given the geographical distribution and the high heterogeneity of Internet, proposals based on a set of identical solutions for each sub-system cannot work, as it will not possible to efficiently handle all possible underlying (different) technologies;
- as the involved sub-systems can range from simple to very complex, the proposed approach must allow a possible recursive handling of the different networks and technologies;
- starting from the global architecture, only key interfaces or APIs have to be defined, to leave as much freedom as possible to designers and users.

### 2.2. Definitions

1. *QoS-Domains* (or QoS-AS) is a domain that offers QoS guarantees for the transversal traffic.

2. *Over-provisioned QoS-Domain*. A QoS-domain D is over-provisioned if it is able to transfer any entering communication, via an ingress border router, to another domain by an egress border router, introducing

a modification of its properties that is under a well defined and accepted threshold. Naturally, the domain owns enough resources to satisfy all the incoming requests.

3. *Controlled QoS-Domain*. A QoS-domain is controlled if it is not Over-provisioned although it contains a control function allowing selecting a subset of the entering communications, for which it ensures the modification of their properties to be under a given threshold.

Therefore, a QoS-domain is a domain that is either controlled (C-AS) or over-provisioned (O-AS). In order to guarantee end to end QoS, all domains on the data path must be QoS Domains (either over provisioned or controlled). The sequence of domains followed by data is given by the Border Gateway Protocol (BGP), the "de-facto" inter-domain routing protocol in Internet.

### 2.3. Bandwidth Broker

The RFC 2638 [10] defines the Bandwidth Broker in the framework of DiffServ as the entity that has the knowledge of a domain's policies and resource availability, and allocates bandwidth with respect of these policies. In order to have a successful end to end reservation across several domains, the Bandwidth Broker managing a domain must communicate with its adjacent peers, which allows configuring the end to end path. This procedure also requires a particular agreement between involved peer domains.

Several concepts of Bandwidth Broker are present in current literature. According to the way they distribute the activity in term of processing, they may be classified as centralized, distributed or hybrid. We adopt the centralized approach in our study. Nevertheless we leave open the possibility to a further evolution of the architecture.

The Bandwidth Broker efficiency depends on the interoperability between all of its subcomponents. Bandwidth Broker functions are distributed both horizontally (among the different QoS domains) and vertically among different layers. Our Bandwidth Broker instantiation functionalities are summarized as follows:

- it acquires topology and routing information;
- it controls end to end network technology independent QoS;
- it implements the suitable signaling in order to support the QoS in all domains on the end to end data path;
- it performs the inter-domain admission control (distributed among all Bandwidth Broker) and local admission control based on resource availability or policies;
- it sets up path and ensures that the requirements will be met by the network when a request is accepted.

### 2.4. Underlying Network Representation

As stated previously, our proposal aims guaranteeing end to end QoS for communications that cross several heterogeneous domains. We present an approach that attempts to address the provisioning, admission control and signaling problems.

First of all we separate the service plane (set of devices offering application functionalities) from the control plane (equipments in charge of establishment and management of communications) and the inter-domain signaling from the intra-domain one. In this way, the domain administrators can implement any particular solution inside their own domain. Moreover, the reservation system can be triggered independently by an application, proxy, gateway or any other trusted entity and the associated network QoS management is independent from any application layer negotiation (for instance SIP, H323 or other proprietary solution).

Our QoS mechanism relies on the characterization of performances inside a domain between each couple of border routers. For such characterization model, please refer to our previous work [11]. As a result, we consider a high level representation of the physical network (devices and interconnections). This representation is independent from the underlying technology and it is stored in a database handled by the Bandwidth Broker. We reduce the topology of the domain to its border routers and the performances between them. The management of each domain is addressed in a hierarchical manner by using the concept of Bandwidth Broker (presented paragraph 2.3). The goal of our approach is to eliminate from the network topology the internal equipments keeping only the border routers (see Fig. 1).

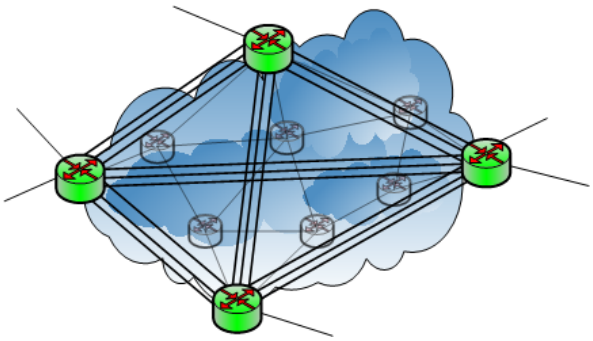


**Fig. 1: Topology Representation**

For instance, considering the bandwidth available between a couple of border routers, we do not store all values from intermediate routers in the database, but a unique value that is the result of composition of all intermediate values (the minimum bandwidth available from each link). Interested readers may refer to [12] for an optimized algorithm to perform such mapping.

### 2.5. Admission Control

Resource provisioning is not enough in order to guarantee at any moment the resource availability on the data path. Consequently, the solution is to set up an admission control applied to all new QoS requests. Therefore, admission control is one of the major tasks that a Bandwidth Broker has to perform, in order to decide if a new request is accepted or not. Admission control modules could have various and sophisticated algorithms using several metrics as peak bit rate, latency, network occupation etc.

There are two possible approaches to perform admission control: measure-based and reservation-based. The first one assumes a periodical estimation of resource load and consumption by the new requests. A new flow is accepted if its load does not produce a congestion risk level. The second approach (that we follow in our proposal) assumes a reservation (by flow, class or aggregate) in some devices. A new request is accepted if the resources are available taking into account prior reservations. In order to pass on the requests to next involved devices, a signaling protocol is required.

In our proposal, the second approach is adopted. We divide the admission control process in two steps:

- Step 1: based on the knowledge from the database and from routing information (network topology, resource availability); a high level admission control is performed using the mapping mechanism described in 2.4. We call this processing “pre reservation”. We prompt that the result of the database is reliable and coherent with the physical resource availability.
- Step 2: after having the confirmation of all following domains (using the signaling protocol that propagates the request to the next peer Bandwidth Broker), the pre-reservation is ratified. This procedure ends with the configuration of devices (routers) by the means of protocols such as COPS, SNMP, etc.

Considering the reservation activation, there are two types of request, taking into account the period for which the resources are desired:

- immediate requests: the resource reservation is effective right after the request result;
- advance requests: the reserved resources are to be used in the future, and do not need immediate installation. The classical example of application that uses this type of reservation is the current pre-established audio or video conference.

Our work is mainly centred on the first design, considering a system delivering QoS on demand with instant result. Nevertheless, the signalling protocol that we propose is not affected by one of these approaches. It can be equally used in both cases by introducing advance request functionality in the BB.

### 2.6. Service Provisioning

To guaranty an end-to-end QoS in the Internet, it is necessary to take into account the heterogeneity of the IP services provided by the domains involved in the data path, by *choosing a concatenation of services* matching the client’s QoS requirements. This choice is part of the *service provisioning* problem, which we address in this paper.

Two points of view are considered to deal with the end to end provisioning:

- in the first one, the problem is approached from a static point of view, where providers offer their clients with end-to-end services having pre-defined static performances; the goal is to establish a compromise between the global capacity of the domain and its external links, and the number of potential clients;
- the second point of view relies on the “on-demand” mechanism using a signaling that dynamically checks and select the end-to-end service matching the QoS requirements; the goal is to invoke the service that best suits the QoS requirements.

In the first case, the user invokes the service that has been defined to be *a priori* adapted to its application. In the second case, the invocation is done only after the adequacy of the concatenated service classes has been verified. Our work follows the second point of view and proposes a characterization of the end-to-end performances, which allows performing a concatenation choice guided by a quantitative expression of the QoS requirements.

So as for the reader to have a more detailed view of the different service provisioning approaches, the next section (2.7) is devoted to the corresponding state of the art. Our contribution to service provisioning is then detailed in section 3.

### 2.7. State of the art of provisioning

Several solutions have been proposed to tackle the service provisioning problem. They are detailed hereafter.

Following the static point of view:

– reference [13] proposes a solution based on the establishment of end-to-end pipes for which bandwidth reservation has been performed. Those pipes only concern the provider's domain and are only established for a few predefined services, supported by all domains. The concatenation choice consists in the choosing of a pipe compatible with the request. The main drawback of the proposed solution comes from the strong homogeneity of the provided services; moreover, the automation of the pipe set up is an open issue, which is necessary to make the approach dynamic;
– the MESCAL approach [14] proposes a solution for inter provider provisioning that quantifies the performances of services before the invocation step. The concatenation choice is based on the extended QoS class concept (e-QC), which is recursively defined as the concatenation of a local class (l-QC) and a e-QC of an adjacent domain. Prior to the subscription requests, the concatenations of the l-QC with the e-QC of the adjacent domains are evaluated; then, one of those satisfying the SLS is retained [15].

Following the dynamic point of view:

– [16] proposes mechanisms allowing the providers to exchange QoS parameters for the supported services in order to provide information on the available QoS before the SLS.
– in [17], the author's proposition is based on service vectors allowing the choice of different successive services retrieved using PROBE RSVP messages on each router, in order to obtain a concatenation matching the targeted requirements. However, the multi domain context is not explicitly considered.

Our proposed provisioning method is also of dynamic nature. However, it defers from [16] by the fact that the concatenation choice is performed at the time of the QoS request (and not at the subscription step). Moreover, we propose a model of service characterization which: (1) allows the request to take the usual forms of the QoS request into account, and (2) is applicable at the scale of one or more domains (typically those involved in the data path). Our proposal follows a similar approach to [17], but in a multi domain context. The dynamic end-to-end provisioning relies on a signaling protocol coupled with an optimized algorithm to choose the best concatenation of classes of services that fulfils QoS requirements and respects a set of predefined preferences.

The reminder of the paper is structured as follows. Section 3 presents the end-to-end service characterization. We first illustrate the interest of the proposed model. Next, we explain the composition and the compliance with a multi domain context. In Section 4, we discuss the signaling solution adapted to dynamic end-to-end provisioning. Finally, Section 5 presents a set of simulation results followed by the conclusion and future work.

## 3. End-to-End Dynamic Provisioning

### 3.1. Service Characterization

The usual performance characterization of a domain (from an entry point to an exit point) is often given in terms of maximal transit delay and/or jitter. The drawback of this model is that it conducts to non optimal characterization when considering the end-to-end service provided by several domains.

In previous work [18], based on ns-2 simulations and real measurements, we propose to characterize the performances between two edge routers by the *cumulative distribution function (CDF) of the transit delay*. Considering *X*, the transit delay of each packet between two edge routers as a random variable, the CDF $F_X$ is defined by $F_X(t) = P(X < t)$, where P defines, for a packet, the probability that its transit delay is lower than *t*.

Such a characterization is interesting for different kinds of application requirements. For instance, if the required QoS is expressed in terms of:

– partial reliability $\tau_r$ (e.g. for a video stream without strong constraint on the delay), then it is satisfied if $\tau_r \leq \lim F(t \to \infty)$;

- partial reliability τd and constraints on the maximal transit delay b (e.g. for distributed games), then it is satisfied if τd < F(b);
- bounded jitter g and constraint on the average transit delay dm (e.g. for interactive audio), then it is satisfied if g ≥ k.σ and dm ≤ $\overline{x}$ , where $\overline{x}$ and σ respectively define the mean value and the standard deviation of the transit delay (k defining a constant function of the probability law of the delay).

## 3.2. Multi Domain Composition

In a multi domains context, there is a need for characterization of the performances resulting from the concatenation of several service classes along a data path involving several domains (Fig. 2).

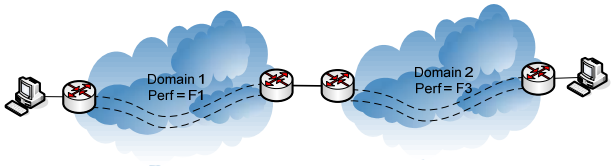



**Fig. 2: Multi Domain Case**

In the case of two consecutive domains $D_1$ and $D_2$, let $X_1$ and $X_2$ be the transit delays of a same packet crossing each of the domains, and let $F_1(t)$ and $F_2(t)$ be the corresponding CDF. One can assume that the transit delays observed in each domain are independent in probability. Thus, the CDF $F_{1,2}(t)$ of the end-to-end transit delay $X_{1,2} = X_1 + X_2$, is given by:

$$F_{1,2}(t) = \frac{d}{dt}\left(F_1(t) * F_2(t)\right)$$

where * indicates the convolution product. The generalization for $n$ domains is obtained using the associative property of the concatenation resulting in:

$$F_{1,n}(t) = \frac{d}{dt}\left(F_n(t) * F_{1,n-1}(t)\right)$$

This result makes it possible to consider that an application can determine the concatenation(s) of classes which meet its QoS requirements once having the knowledge of performances for all services in domains on the data path.

## 3.3. Formalization

### 3.3.1. Hypothesis

Our proposal relies on the following assumptions:

- Inside each domain, there are one or more Classes of Service (CoS) supported by the providers;
- An entity in each domain has the knowledge at each moment of resource availability for each CoS between all couples of ingress-egress point (routers) in the domain.

For each CoS *i* in the domain D, we associate:

- An amount $Bw_{iD}$ of available bandwidth;
- A cost $C_{iD}$ for a client to use it;
- A QoS function $Fqos_{iD}$ which represents the resource characterization between each couple of ingress-egress point in the domain. In our work we considered for now the characterization described in section 3.1 assimilating to the CDF, but other function may be used as well.

### 3.3.2. Multi domain service composition

We propose an evolutionary approach that: (1) first evaluates the end-to-end performances on the data path and resulting from all possible concatenation and (2) choose and invokes the most adequate one with regard to a set of preferences and that fulfils application's QoS requirements at the same time. The preferences can be chosen from various criteria, grouped into two main categories:

- User-oriented (lower cost) or
- Provider oriented (income maximization, resources utilization).

Let us remark that a combination of several preferences can be also considered. Nevertheless, in this paper we illustrate our proposal with an approach that minimizes the cost by choosing the less expensive concatenation from all possible alternatives.

Based on the client QoS request (in terms of bandwidth and parameters presented in section 3.1), the conditions to be fulfilled by the chosen concatenation of CoS are:

$$1) \rightarrow Bandwidth \leq Min\left(Bw_i\right)$$

$$2) \rightarrow F_{target} > \left(Fqos_{1D} \circ Fqos_{2D} \circ .. \circ Fqos_{ND}\right)$$

where "o" represents a general function composition. In our instantiation, the target function is represented by the convolution product as explained in section 3.2. This means that on the data path there is enough bandwidth for the service and additionally, the application QoS requirements are satisfied.

Let suppose N QoS domains and at most M CoS in each domain. The goal of our selection mechanism is

to choose a vector $[CoS_{D1},..,CoS_{DN}]$ (one CoS in each domain on the data path) under conditions 1) and 2) such that $\underset{D}{Max}\, QoS(D)$). In other words, the mechanism is able to find a concatenation of CoS in each domain that fulfils application requirements (1 and 2) and that maximize a *QoS* function with respect to imposed preferences. An example is illustrated in Fig. 3. In our study, the *QoS* function that minimizes the cost is:

$$QoS(D) = \sum_{i=1}^{N} \underset{j=1}{\overset{M}{Min}}(C_{ij})$$

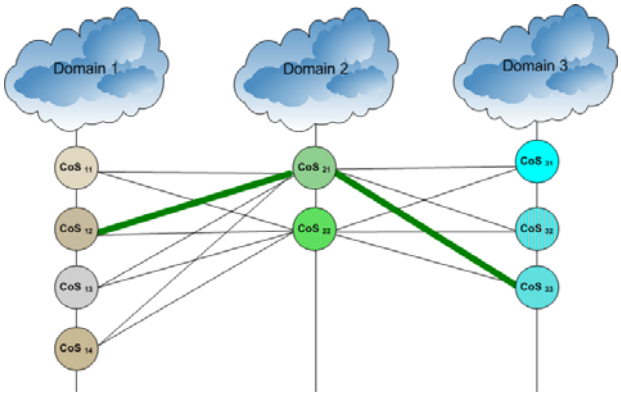



**Fig. 3: Class of Service Selection**

## 4. Signaling Protocol

As presented in Section 2, several models of QoS management have been proposed for multi domain Internet. The convergence point of these proposals is the necessity to set up a signaling in order to interoperate between different network equipments. Several contributions dealing with the signaling issue have been considered, mainly at the IETF (especially Next Step In Signaling NSIS work group). Two perspectives are present in the literature:

- path-coupled signaling (also called "on-path") that extends the IntServ/RSVP view. The entities involved in signaling process are mandatory on the data path;
- path-decoupled signaling (or "off-path"). The signaling entities cannot be all located on the data path, but they are aware it.

In our work we adopt a path decoupled approach in a hierarchical way using the Bandwidth Brokers (BB) concept [9] for the intra domain management. Moreover, we assume the knowledge by each BB of the IP services for all couples of its border routers. Therefore the BBs are the main admission control and signaling equipments of the domain. Considering the hierarchical management of a domain, we consider the off-path solution in order to impose arrival of signaling messages to the Bandwidth Broker.

The concatenation of the domains and the related admission control are performed dynamically after a QoS request expressed in terms of parameters such as a maximal transit delay, a maximal loss rate, etc.
The concatenation choice is resolved in three steps:

- first, the classes of service (and their performances expressed by means of the CDF of the transit delay) available on each domain of the data path(s) are discovered;
- second, the end-to-end performance model is evaluated for all the service classes available on the data path; this evaluation is performed by means of convolution;
- third, the choice of the adequate service satisfying the QoS request and given preferences is performed. This algorithm is implemented at the client's level (or proxy).

The signaling protocol will then be handled by:

- the sender and receiver hosts or dedicated equipments such as proxies.,
- the Bandwidth Broker of each domain.

Our protocol relies on BGP (Border Gateway Protocol), the inter AS routing protocol used in the Internet. The sequence of domains and the two data and signaling paths are illustrated in Fig. 4. Our solution decouples then the inter-domain signaling from the intra domain one. In all domains, decisions are local, and so any routing protocol can be defined within one domain. The purpose of this approach is to give a maximum degree of freedom to providers to implement the most suitable solution inside their domain. Consequently, the end-to-end inter-domain path is given by BGP tables and internal path (i.e. within each AS) can be freely selected by the AS providers, depending on local constraints.

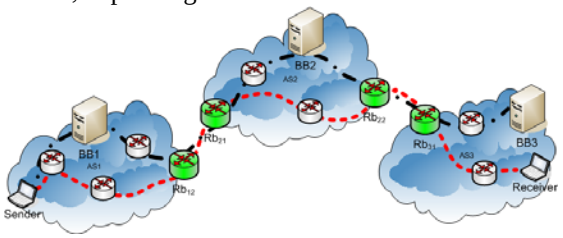



**Fig. 4: BGP Data Path and Signaling Path**

The performance of the end-to-end path and later the corresponding admission algorithm requires checking availability of the needed resources along the data

path. This has to be done by some dedicated equipment, the Bandwidth Broker. The following Fig. 5 illustrates the case of a QoS request which transverse three domains. Following [19], the selection of the service matching the QoS requirements is based on:

- the discovery of the available services on the data path (request/response PDU exchanges),
- the characterization of the end-to-end services resultant from the composition of the available services classes, and then the selection of the cheapest service matching the QoS requirements,
- the reservation and the refreshing of the selected service class.

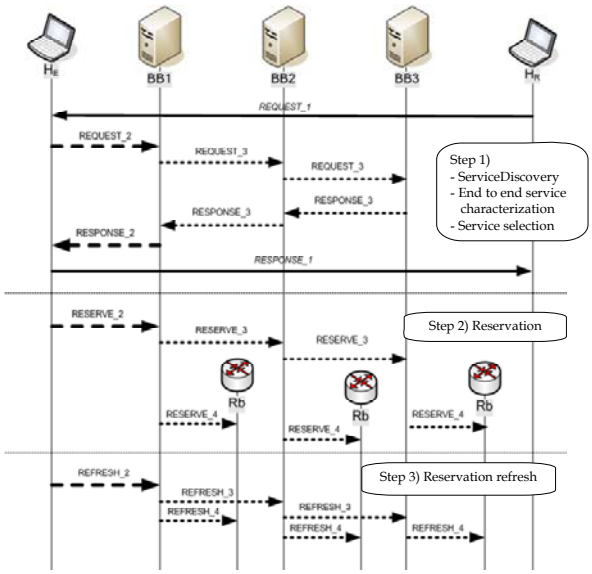



**Fig. 5: PDU exchange**

## 5. Simulation Results

In this section, the simulation results obtained for both the signaling protocol and the dynamic provisioning algorithm presented previously are presented. In both studies, issues regarding the scalability of the approaches are identified and discussed.

### 5.1. Evaluation of the architecture and signaling protocol

#### 5.1.1. Performance tests

The performance tests performed on our signaling protocols were conducted on a platform composed by:

- power edge 750 computers with Intel Pentium 4 processor at 3 GHz and 1 GB RAM memory; (running Linux Debian or Fedora with 2.);
- two Cisco switches c2960.

In order to test and validate our implementation, we developed multithread tools that simulate the high level behavior of an application and trigger the QoS system. The java virtual machine used was 1.5.0_11 and the database server was MySQL 5.0.18. Using the testbed described above, we emulated the multi-domain context presented in figure 4.

We first measured the processing time on all Bandwidth Brokers, the results being comparable for all of them. We stressed the system by launching simulation of 5, 10, 50, till 200 requests per second. Fig. 6 illustrates the answer time for each of the above scenarios. Note that time consists of request parsing, several access to the database, state management and also an emulation of device configuration. Let us observe the increase response time when the number of requests became greater than 150 per seconds. This can be explained by the different time spent in the queues while waiting for processing. Tuning the implementation prototype to take into account this limitation is one of the prospects of this work.

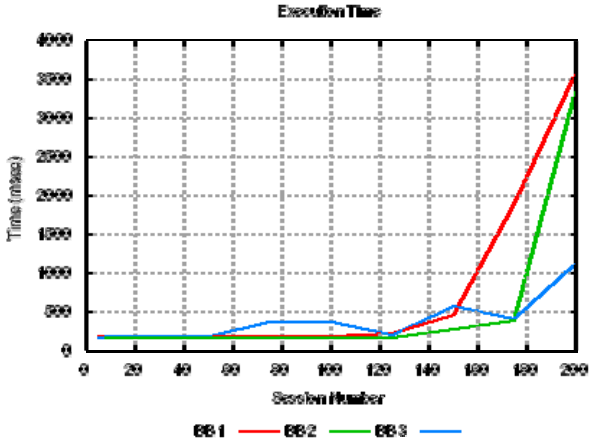



**Fig. 6: Processing time**

In a second time, we measured the round trip time between the client request and the response arrival (Fig. 7). We suppose a communication crossing three domains and this time includes: the processing time in each Bandwidth Broker, the time to exchange signaling messages (RESERVE and RESPONSE) and the update of configuration in each domain.

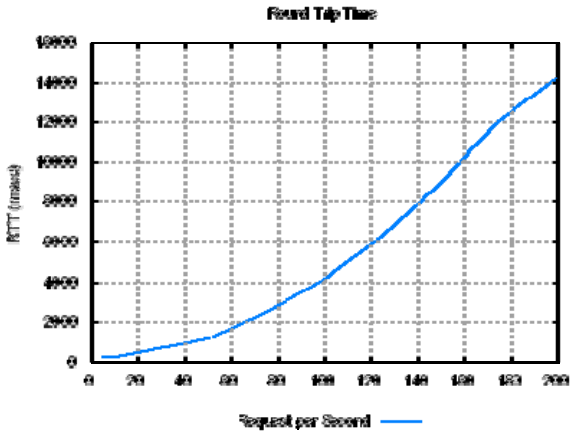



**Fig. 7: RTT for three domains**

Let us remark that the response time is increasing with the number of requests. The mean RTT remains under 4 seconds for less than 100 requests per second and under 8 seconds between 100 and 150 requests per second. We also observe an increase of this time after 150 requests per second, with a mean around 12 seconds for 200 requests per second. However, the processing and request handling in each domain of our protocol conducts to satisfactory values of the RTT for reasonable number of requests per second.

Next Fig. 8 and 9 describe the usage of CPU and memory, measured during the same simulation on the second BB. During the processing of all 200 flows, a separate process investigates each second the resource utilization of the computer where the Bandwidth Broker is located.

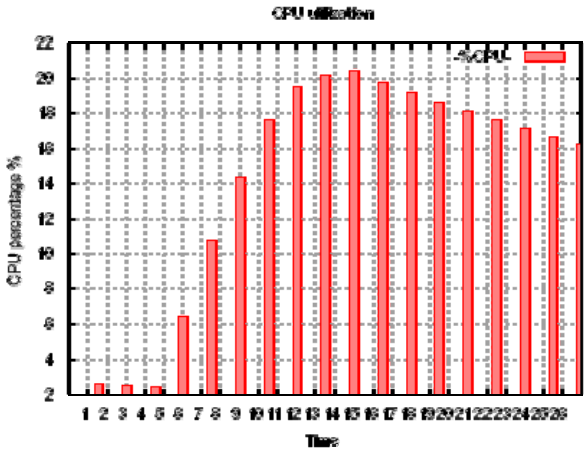



**Fig. 8: CPU utilization**

It is worth mentioning that for this specific test we used the results obtained by the Linux based tools (top, htop, ps). Therefore, the CPU utilization represents the CPU time used divided by the time the process has been running (CPU time / real time ratio) expressed as a percentage. We can observe the increase during the processing off requests, the values being satisfactory even if we didn't use very powerful equipments (such as the operator owns).

Let us note the augmentation of memory (the physical resident size that a task uses expressed in Kilo Bytes) due to possibly creation of new threads to process new requests. It is worth to observe that after handling all requests, the CPU percentage is decreasing while the memory remains at its last stable level. This is caused by the implementation collaborated with the memory allocation/free algorithm of the virtual machine. When all the threads in the handling pool are started the memory will not increase, having a stable value.

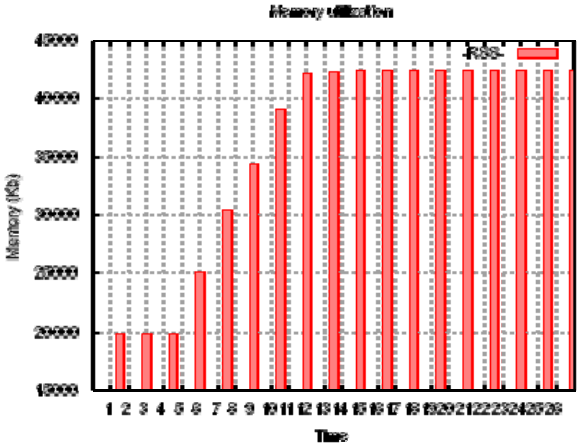



**Fig. 9: Memory Utilization**

### 5.1.2. Conclusions

The previous results show that the memory and CPU consumption of BBs remain in reasonable values. The RTT and the processing time are also acceptable in the vision of a session establishment in a multi-domain context, matching ITU requirements [20]. Moreover, the network overhead introduced by our signaling protocol is limited (in the current implementation 64 bytes to request reservation for a flow and 68 bytes for the response).

In the general case, QoS seems to be difficult to deploy. Nevertheless, the users requesting QoS will be much less numerous than the number of best-effort users. It will start with a reasonable number of users and domains, and extend them when the number of QoS requests will grow. Moreover, when the number of requests becomes important, the tasks of the

Bandwidth Broker can be distributed.

## 5.2. Performance of the dynamic provisioning algorithm

This section is focused on the performance results related to the dynamic end-to-end provisioning scheme which has benne proposed in section 3. Let us recall that we consider an algorithm aimed at choosing the "best" (here the less expensive) CoS concatenation along the data path, using the mechanism described in section 3.2. Details for the composition using cumulative distribution function can be found in [11].

Our simulation is based on a Java implementation of the proposed model. We use a multi domain model similar to the one presented in Fig. 4 considering four domains (identified from 0 to 3). For simplicity, we consider three CoS in each domain, having the same QoS characteristics. We name these classes $CoS_1$, $CoS_2$ and $CoS_3$, and we assume that the quality associated with these classes is such that: $QoS(CoS_3) > QoS(CoS_2) > QoS(CoS_1)$. Consequently, the price to use one of these classes follows the same relation. The bandwidth amount allocated in each domain for each class of service is 60% for $CoS_1$, 20% for $CoS_2$ and 20% for $CoS_3$.

The simulation time is set to take into account one day with collection of results each second using 300000 clients equally spread in each domain. The communication duration of each client follows the Poisson law and the reservation invocation time is uniformly distributed throughout the simulation duration. Each client performs one QoS request through several randomly chosen domains (the destination domain being always identified by a grater number). We compare our model with the general most used one that statically associates a well defined CoS to a given type of application (same in each domain). In this basic approach, the user attempts to reserve the same related CoS in all domains on the data path.

We remind that in our approach, the QoS request does not precise a specific CoS, but only parameters. Retrieving all available CoS on the data path and based on the application requirements and given preferences, a concatenation of CoS (which can be different on each domain) is chosen and resources are reserved afterwards.

We consider three types of application, each one having specific well defined QoS characteristics (i.e. video streaming, telephony, etc). The need in QoS for these applications follows a similar relation as for the CoS: $N(Type_3) > N(Type_2) > N(Type_1)$. A client uses one of the application types following a probability law: 0.6 for $Type_1$, 0.2 for $Type_2$ and 0.2 for $Type_3$.

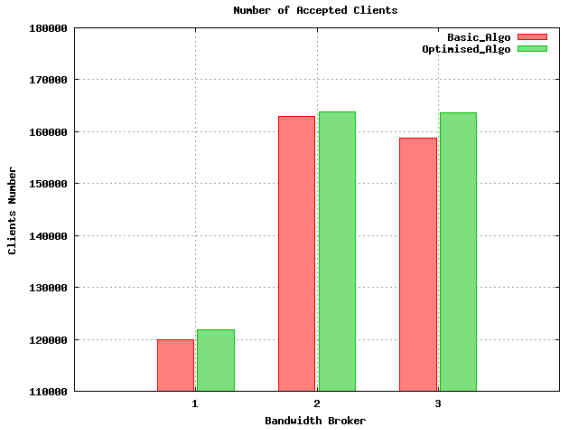



**Fig. 9: Number of Accepted Clients**

First of all, we analyze the number of clients accepted in each domain. Fig. 9 illustrates the number of accepted clients and Fig. 10 the total number of clients that made a QoS request to a BB (per domain).

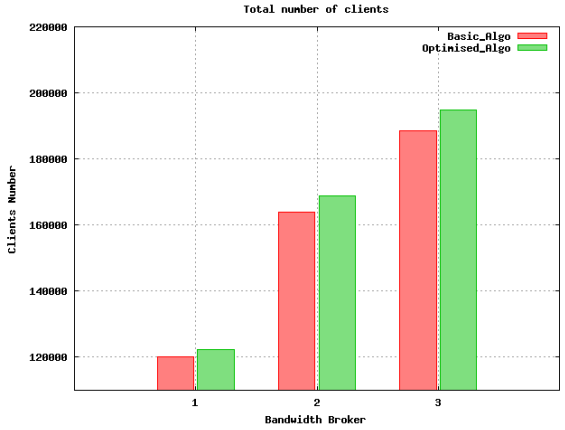



**Fig. 10: Total Number of Clients**

We can observe that with our model, the number of accepted clients is greater on each domain compared with the classical algorithm. Furthermore, the total number of clients is superior in our approach, meaning that more QoS request can be satisfied. The fact that a greater number of client requests are processed with our algorithm results from the behavior of our signaling approach: if the resources are not available, the request is not propagated forward. Compared with the basic general approach, our methodology gathers information for all available CoS and do not reduce the solution to only one CoS. This flexibility allows a greater number of requests to be satisfied.

These results are also confirmed by the Fig. 11

which represents the bandwidth occupation on the second Bandwidth Broker (on the other BB the results are similar; we choose to illustrate the second one as a greater number of requests are processed). We can remark that using our algorithm, we obtain a better occupation of the bandwidth which is predisposed to increase the profits for providers.

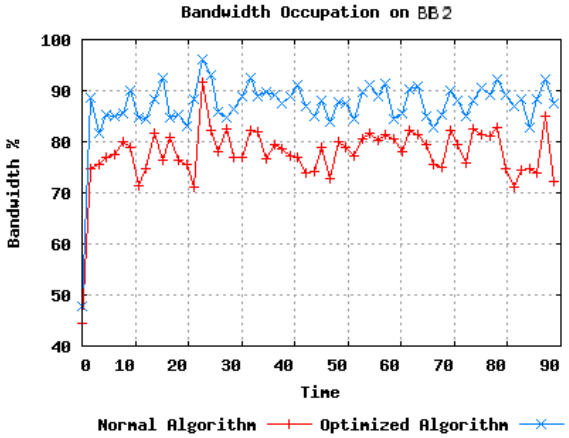



**Fig. 11: Total Bandwidth Occupation**

We also analyzed the bandwidth occupation for each CoS. We illustrate the $CoS_1$ as it is the most used in these simulations (see Fig. 12).

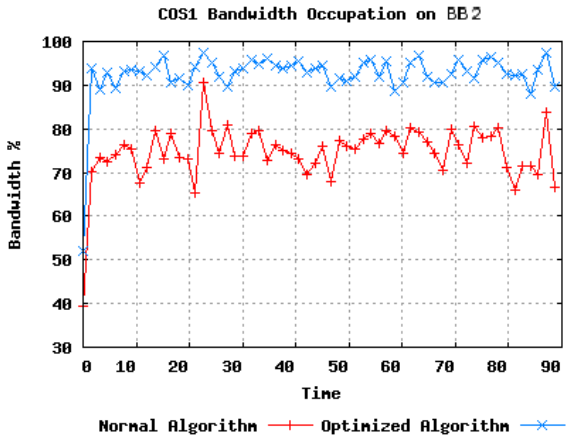



**Fig. 12: CoS 1 Bandwidth Occupation**

Let us remark that we also performed simulations using different distributions of application types with the same three CoS. Results are comparable, the bandwidth occupation being greater with our algorithm, however with increasing differences if the proportion of $CoS_2$ and $CoS_3$ are greater.

Based on these results, we conclude that not only the degree of satisfaction of clients is improved, but also that providers can take advantage of this approach. Having a larger number of clients and a better bandwidth occupation implies more incomes and also an optimization of resources.

## 6. Conclusion

This paper addresses the problem of providing a guaranteed end to end QoS over multiple Internet domains. More precisely, we proposed a solution of QoS domain architecture managed in a hierarchical manner by a Bandwidth Broker. In addition we presented solutions for both inter-domain signaling and optimized QoS management to perform admission control. With respect to other similar contributions, our proposal is independent of the underlying network technology and minimizes the constraints introduced to the global architecture. In addition to the implementation of several modules, we presented a first round of tests that aims at validating our solution.

The main perspectives of this work are:

- to conduct further investigation on performance and scalability;
- to take into account the heterogeneity of the Internet with non QoS domains and to consider several different domains with different technologies.
- to introduce other criteria for our selection algorithm (possible provider preferences. In this paper we used minimization of cost as criteria).
- to extend our solution in order to integrate the evolution of the NSIS protocol suite.

## 7. Acknowledgment

This work has been partially conducted within the framework of the European IST EuQoS project (http://www.euqos.org).

## 8. References

[1] Racaru, F.; Diaz, M.; Chassot, C., "Quality of Service Management in Heterogeneous Networks," *Communication Theory, Reliability, and Quality of Service, 2008. CTRQ '08. International Conference on* , vol., no., pp.83-88, June 29 2008-July 5 2008

[2] Wroclawski J. Specification of the Controlled-Load Network Element Service. RFC 2211, Sept. 1997.

[3] Shenker S et al. Specification. of Guaranteed Quality of Service. RFC 2212, September 1997.

[4] Braden R et al. Resource ReSerVation Protocol (RSVP) - Version 1 Functional Specification. RFC 2205, September 1997.

[5] Blake S et al. An Architecture for Differentiated Services. RFC 2475, December 1998.

[6] Jacobson V. et al. An Expedited Forwarding PHB. RFC 2598, June 1999.

[7] Heinanen J et al. An Assured forwarding PHB. RFC 2597, June 1999

[8] Bernet, T et.al "A Framework for Integrated Services Operation over DiffServ Networks", IETF RFC 2998, November 2000

[9] Dugeon O., Morris D., Monteiro E., Burakowski W., Diaz M, End to End Quality of Service over Heterogeneous Networks (EuQoS), IFIP Network Control and Engineering for QoS, Security and Mobility, Net-Con'2005, November 14–17, 2005, Lannion, France

[10] Nichols N et al. A Two-bit Differentiated Services Architecture for the Internet, RFC 2638, July 1999

[11] Chassot C, Auriol G, Diaz M. Automatic management of the QoS within an architecture integrating new Transport and IP services in a DiffServ Internet. 6th IFIP/IEEE International Conference on Management of Multimedia Networks and Services (MMNS'03), Belfast, Ireland, September 7-10, 2003.

[12] Htira W, Duegeon, O , Diaz, M An aggregated delay/bandwith star scheme for admission control 12th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD'07), Athènes (Grèce), 7 Septembre 2007, 11p.

[13] Mantar H. et al A scalable model for inter bandwidth broker resourse reservation and provisioning. IEEE Journal on selected Areas in Communication, Vol. 22, December 2004.

[14] Goderis D, T'joens Y, Jacquenet C, Memenios G, Pavlou G, Egan R, Griffin D, Georgatsos P, Georgiadis L, Van Heuven P. Service Level Specification Semantics, Parameters and negotiation requirements. Internet draft, June 2001.

[15] Howarth MP et al. Provisioning for interdomain quality of service: the MESCAL approach. IEEE Communication Magazine, Vol 43, June 2005.

[16] Füzesi P, Németh K, Borg N, Holmberg R, Cselényi I. Provisioning of QoS enabled inter-domain services. Computer Communications, vol. 26 n°10, June 2003.

[17] Yang J, Ye J, Papavassiliou S. A flexible and distributed Architecture for adaptive End-to-End QoS Provisioning in Next Generation Networks. IEEE Journal on Selected Areas in Communications, vol. 23, n°2, February 2005

[18] Chassot C. et al Signaling in heterogeneous IP multi domain networks, 6th Internation Conference on Next Generation Teletraffic and Wired/Wireless Advanced Networking (NEW2AN), May 2006.

[19] Chassot C., Lozes A., Racaru F., Auriol G., and Diaz, M. A user based approach for the choice of the IP services in the multi domain DiffServ Internet. First IEEE workshop on Service Oriented Architecture in Converging Networked Environments, Vienna Austria 2006.

[20] ITU-T Recommendation E.721 Network grade of service parameters and target values for circuit-switched services in the evolving ISDN.

# Multi-band Gigabit Mesh Networks: Opportunities and Challenges

L. Lily Yang and Minyoung Park
{lily.l.yang, minyoung.park}@intel.com
Intel Corporation, Hillsboro, OR 97124

***Abstract* — 60 GHz has attracted a lot of commercial interest due to its abundance of unlicensed frequency spectrum and the recent advances in building inexpensive transceivers. IEEE 802.11 has started a new Task Group, 802.11ad, to develop a 60 GHz PHY and MAC that can deliver at least 1Gbps MAC throughput. The 802.11ad amendment is also required to enable seamless session switch between the existing 2.4/5 GHz and the new 60 GHz radio. This paper proposes the system concept of multi-band gigabit mesh networks that can potentially satisfy the requirements set out in 802.11ad. The benefits of multi-band gigabit mesh networks are presented, including the diversity gain, range extension and spatial reuse gain. In particular, a 2-hop 60 GHz mesh is simulated for an office WLAN example to demonstrate that the spatial reuse gain can be very significant in 60 GHz mesh networks thanks to the nature of the highly directional beamformed links. Such high degree of spatial reuse can significantly improve the end to end network throughput of a multi-hop mesh network while providing reasonable range. The key challenges for the multi-band gigabit mesh are also discussed along with some open research questions for future work.**

***Keywords: mmWave, 60GHz, Multi-band, Multi-channel, Mesh Networks, Gbps Wireless Networks, IEEE 802.11ad, IEEE 802.11VHT***

## I. INTRODUCTION

The abundance of bandwidth in the unlicensed 60 GHz band (57-66 GHz band, also known as the millimeter-wave band) has attracted more and more interest from both the research community and the industry [1-11] in recent years for short range indoor wireless communications including both Wireless Personal Area Networks (WPAN) and Wireless Local Area Networks (WLAN). Recent advances [12] of using SiGe and CMOS to build inexpensive 60 GHz transceiver components has created commercial interest to productize and standardize 60 GHz radio technology for mass market applications.

This higher frequency band comes with a larger free space propagation loss which must be compensated for by high gain directional antennas. Fortunately, high gain directional antennas are feasible to implement even for small form factor devices due to the shorter wavelengths (5 mm).

60GHz channel generally exhibits quasi-optical properties, meaning the strongest components tend to be Line of Sight (LOS). Non Line of Sight (NLOS) components do exist, mostly in the form of reflection. However, the short wavelengths in this band impose some serious challenges such as greater signal diffusion and difficulty diffracting around obstacles. 60 GHz band measurements [13] show that in general, the strongest reflected components are at least 10 dB below the line of sight (LOS) component. Even more challenging are the problems caused by obstructions. A human body walking into the path between the transmitter and the receiver can attenuate the signal by 15 dB or more and easily break the link. Common objects such as furniture, walls, doors and floors found in indoor environments can also be problematic. As a result, the practical indoor operation range at 60 GHz is likely to be limited by penetration loss instead of free space propagation loss and therefore mostly confined to a single room. In comparison, the link characteristics are very different in the lower frequency bands such as 2.4/5 GHz, where penetration loss is less, rich multi-path exists to provide diversity, and the range can reach up to hundreds of meters. Millions of users have come to enjoy the convenience of broadband wireless access thanks to 802.11-based WLAN technology (aka Wi-Fi) in the home, office and hotspots. It is commonplace for Wi-Fi users to experience link quality fluctuations and even link outage, but typically not just because someone walks by. As most wireless users don't really care about or even know the difference between RF bands, it will be natural for the users to compare their usage experience of 60 GHz products with that of Wi-Fi. While higher throughput will enhance the user experience, other factors such as ease-of-use, robustness and range will also significantly affect the experience. We believe delivering satisfactory range and robustness along with gigabit level performance is one of the most important challenges for MAC and system design at 60 GHz, and that is the motivation behind the concept of multi-band gigabit mesh networks proposed in this paper.

This paper proposes the system design concept of multi-band gigabit mesh networks and articulates why this may significantly improve the user experience with 60 GHz technology. The focus of the paper is on quantifying the benefit of such concept instead of the protocol design and implementation details.

Section II briefly surveys the multiple standardization efforts taking place today, including the newly formed IEEE 802.11ad Task Group. Section III compares the data rate and range tradeoff at 5 and 60 GHz and show the complementary nature of these bands. Such complementary nature of 2.4/5 and 60 GHz bands motivates the design concept of multi-band gigabit mesh networks. Section IV introduces the concept of multi-band gigabit mesh, presents the target usages for this concept and the benefits. Spatial reuse in 60 GHz mesh is especially significant due to the nature of highly directional beamformed links, and detailed simulation results are presented for an office scenario as an example in Section V. Section VI contrasts our proposal with some of the prior work. Section VII highlights new design and research challenges in the

framework of multi-band Gigabit mesh and Section VIII concludes the paper.

## II. 60GHZ STANDARD EFFORTS

Several international standard bodies have ongoing standard development efforts for 60 GHz specifications. ECMA TC48 [6, 14] has completed a 60 GHz PHY and MAC standard specification to provide high rate WPAN transport. The usage cases are high definition (uncompressed or lightly compressed) AV streaming, wireless docking station and short range sync-n-go. The IEEE 802.15.3c Task Group [7] is also in the process of developing a millimeter-wave-based alternative PHY for the existing 802.15.3 WPAN Standard 802.15.3-2003. This mmWave WPAN will support at least 1 Gbps and optionally 2 Gbps for applications such as high speed internet access, streaming content download, multiple real time HDTV video streams and wireless data bus for cable replacement.

The IEEE 802.11 working group began a new "Very High Throughput" study group (VHT SG) [10] in 2007 to investigate technologies beyond 802.11n capabilities for WLAN. The Wi-Fi Alliance (WFA) was consulted to develop the usages for VHT SG and the six categories of usages envisioned [17] include wireless display, in home distribution of video, rapid upload and download to and from a remote server, mesh or point-to-point backhaul traffic, campus or auditorium deployments, and manufacturing floor automation. Two 802.11 Task Groups have been formed toward the end of 2008 as a result of the work done by VHT SG. One of the two Task Groups, 802.11ad, is chartered to define standardized modifications to both the 802.11 physical layers (PHY) and Medium Access Control Layer (MAC) to enable operation in the 60 GHz frequency band capable of at least 1 Gbps, as measured at the MAC data service access point (SAP). The 802.11ad amendment will also enable fast session transfer between 60 GHz and 2.4/5 GHz PHYs. The fast session transfer between 60 GHz and 2.4/5 GHz PHYs will distinguish 802.11ad solution from the others including ECMA TC48 and IEEE 802.15.3c.

The system concept of multi-band gigabit mesh network proposed in this paper is largely motivated by the objectives of 802.11ad to leverage both the existing 802.11 (a.k.a. Wi-Fi) solutions in 2.4/5 GHz and the new solution in 60 GHz band to reach gigabit level performance for WLAN and WPAN applications.

## III. RANGE AND PEROFRMANCE TRADEOFF IN 5 AND 60 GHZ BANDS

In order to understand the motivation and benefits of multi-band gigabit mesh networks, let us first examine the propagation characteristics and channel properties of 60 GHz band in comparison with that of 5 GHz.

Using the log-distance path loss model [20], the average path loss $\overline{PL}(d)$ between a transmitter and a receiver separated by $d$ (m) can be calculated as follows

$$\overline{PL}(d)[dB] = 20\log\left(\frac{4\pi d_0}{\lambda}\right) + 10n\log\left(\frac{d}{d_0}\right) \quad (1)$$

where $d_0$ is the close-in reference distance, $\lambda$ is the wavelength, and $n$ is the path-loss exponent. The penetration loss of a non-LOS (NLOS) environment is abstracted as a larger path loss exponent. In [19], the 60GHz channel measurements show that $n$ is approximately 3.5 for NLOS and 2 for LOS at 60 GHz; the path loss exponent is only 2.6 for NLOS at 5 GHz.

TABLE I
PARAMETERS FOR 5GHZ AND 60GHZ

| | 5GHz | 60GHz |
|---|---|---|
| Number of antennas | 4 | 36 |
| Maximum EIRP (FCC) | 30dBm + 6dBi (antenna gain) | 40dBm |
| Maximum transmit power/antenna ($P_t$) | 24dBm | 4dBm |
| Transmit beamforming gain ($G_t$) | 0 | 15dB |
| Power combining gain ($G_c$) | 0 | 15dB |
| Receive beamforming gain ($G_r$) | 0 | 15dB |
| Aperture loss @ 1m | -48dB | -68dB |
| NLOS path loss exponent (n) [19] | 2.6 | 3.5 |

TABLE II
MCS FOR 5GHZ

| Data rate | 120Mbps | 180Mbps | 360Mbps | 600Mbps |
|---|---|---|---|---|
| Modulation | BPSK | QPSK | 16QAM | 64QAM |
| Code rate | 3/4 | 3/4 | 3/4 | 5/6 |
| $E_b/N_0$@BER10e-5 | 9.6dB | 9.6dB | 14.5dB | 19dB |

TABLE III
MCS FOR 60GHZ

| Data rate | 1.2Gbps | 2.5Gbps | 5Gbps |
|---|---|---|---|
| Modulation | BPSK | QPSK | 16QAM |
| Code rate | 3/4 | 3/4 | 3/4 |

Although 60 GHz experiences much higher path loss than the 5 GHz band, the very short wavelength makes it possible to integrate a very large number of antennas (e.g. 36 antenna elements) in a very small area and use the array antenna for beamforming to compensate for the additional 20 dB of path loss due to operation at a much higher frequency. Assuming the transmitter and receiver both have $N_a$ antennas, the transmit and receive beamforming gain can be expressed as $G_t$[dB]= $G_r$[dB]=10log$N_a$. Assuming 36 antenna elements on each side of the link, the beamforming gain is approximately 30 dB, which not only compensates for the additional path loss but also increases the link budget of the 60 GHz link by 10dB.

Assuming the transmit power of each antenna is $P_t$[dBm], the link budget of a 60 GHz link $P_{LB}$[dB] can be expressed as follows

$$P_{LB}[dB] = P_t[dBm] + G_c[dB] + G_t[dB] + G_r[dB] - \overline{PL}(d)[dB] - N[dBm] - L[dB] - SNR_{\min} \quad (2)$$

where $G_c$ is the power combining gain due to the distributed power amplifiers on each RF chain, $N$ is the noise power, $L$ is the sum of noise figure and other implementation losses, and $SNR_{\min}$ is the minimum signal-to-noise ratio (SNR) for a specific modulation and coding scheme (MCS) that guarantees the quality of the link (e.g. bit error rate (BER) <10e-5). From (1) and (2), the transmission range $d$ that satisfies BER<10e-5 can be derived.

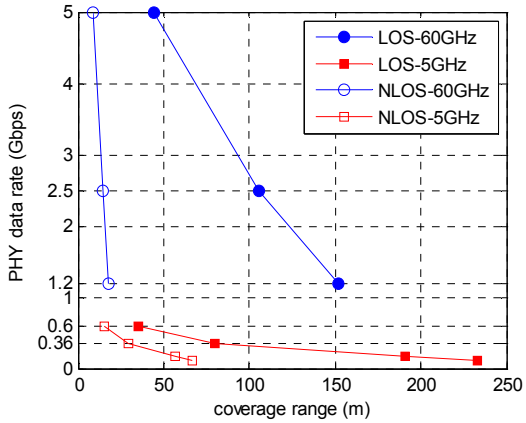



Fig. 1. Comparison of transmission ranges and data rates of 5 GHz and 60 GHz in LOS and NLOS environments using maximum transmit power per antenna ($P_t$ = 24 dBm for 5 GHz and $P_t$ = 4 dBm for 60 GHz)

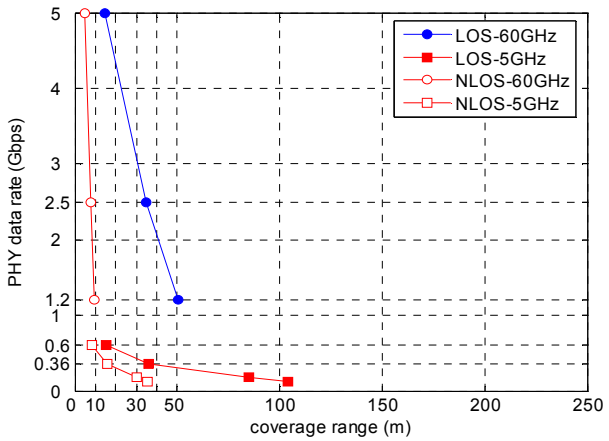



Fig. 2. Comparison of transmission ranges and data rates of 5 GHz and 60 GHz in LOS and NLOS environments using typical transmit power ($P_t$ = 17 dBm for 5 GHz and $P_t$ = -5.6 dBm for 60 GHz)

TABLE I shows some of the parameters of 5 GHz and 60 GHz. TABLE II and TABLE III show the MCS and PHY data rates used for numerical analysis. Using the parameters shown in Table I, II, and III, and (1) and (2), the transmission ranges of various MCS modes of 5 GHz and 60 GHz can be calculated and compared. Fig.1 shows the numerical results comparing the transmission ranges and the data rates of 5 GHz and 60 GHz in both LOS and NLOS environments based on the measurement results of [19] with the maximum transmit power per antenna ($P_t$ = 24 dBm for 5 GHz band assuming a WiFi device and $P_t$ = 4 dBm for 60 GHz band). It is interesting to note that in a LOS environment, 60 GHz is comparable to 5 GHz in terms of the transmission range and also has a much higher data rate. The reasons are as follows:

- the additional 20 dB path loss at 60 GHz is already compensated for by the transmit and receive beamforming gains (~30 dB)
- 60 GHz has higher EIRP (Effective Isotropic Radiated Power) than 5 GHz
- 60 GHz can use a very simple modulation scheme such as BPSK or QPSK to achieve a very high data rate (>1 Gbps) by utilizing a very wide bandwidth (~1.7 GHz), which requires very low $E_b/N_0$ (~10 dB) for reliable communications.

For a NLOS environment, however, the transmission range of 60 GHz quickly decreases and becomes much shorter than 5 GHz due to much higher penetration loss of obstacles between the transmitter and receiver. Considering the typical transmit power of 5 GHz and 60 GHz devices, the transmission range further decreases. Fig. 2 compares the transmission ranges and the data rates of 5 GHz and 60 GHz with a typical transmit power. For an 802.11 device, a typical transmit power per antenna is approximately 17 dBm. For a 60 GHz device, the total transmit power of a 60 GHz device is limited to 10 dBm due to the regulations in Korea, Japan, and Australia [39], which limits the transmit power per antenna to $P_t$=-5.6dBm for the 36 antenna case. Fig. 2 show that the transmission range of 60 GHz over a NLOS channel is less than 10 meters.

Fig.1 and Fig. 2 clearly show that the tradeoff of range and performance for 5 GHz and 60 GHz is different and complementary in nature. While 5 GHz excels in achieving larger range and robust performance in the NLOS channel, the performance of 60 GHz link is much higher with very short range but it drops off rather quickly as the range increases, especially with NLOS channel. The complementary nature of these two bands suggests a compelling reason to combine the two so to keep the best of both worlds. This is exactly the motivation behind the concept of multi-band gigabit mesh networks.

## IV. BENEFITS OF MULTI-BAND GIGABIT MESH

### A. *Multi-band Gigabit Mesh: The Concept*

The concept of multi-band gigabit mesh is to allow the flexibility of two devices communicating with each other in either the low frequency band such as 2.4 or 5 GHz, or in the 60 GHz frequency band. Moreover, there is also the flexibility to choose a multi-hop 60 GHz path, if it is advantageous for the network performance and user experience. There are three distinct characteristics that define a multi-band gigabit mesh: 1) Some, if not all, of the devices in the mesh are capable of operating in multiple radio bands, more specifically, in a low frequency band such as 2.4 or 5 GHz (using Wi-Fi technology), and in the 60 GHz band. The network should exploit the multi-band capabilities of these devices to improve the user experience with these devices and the network in general. 2) There exists at least one multi-hop 60 GHz path among the 60 GHz devices. 3) The end to end performance between any two devices in the mesh should be above 1 Gbps measured at the MAC level.

### B. *The Target Usages for Multi-band Gigabit Mesh*

Looking across the various standard efforts for 60 GHz, it is clear that the superset of the usages addressed by these efforts really span across WPAN and WLAN. Let's consider the requirements of these usages in terms of range, channel condition and link robustness, and see which usages can benefit from the concept of the multi-band gigabit mesh networks.

- *Very short range (<1 meter) with LOS guarantee*

The first set of usages operate at a very short range (<1 m) with a LOS condition almost always guaranteed. One such usage example is sync-n-go. Sync-n-go refers to the usage of

rapidly transferring data from one device to another. It may be downloading a multimedia file like HD (High Definition) movie from a Kiosk to a handheld device; it may also be exchanging photos between two peer to peer devices such as cell phones. The speed is the key to make this usage compelling, especially when the amount of data is huge. Sync-n-go typically does not involve a range beyond 1 meter, and so the user has greater control to help position the devices so that they are in range with LOS toward each other. This set of usages is generally considered the least challenging from range, link budget and channel condition's point of view. This category of usages does not need multi-band gigabit mesh to achieve satisfactory user experience and so such usages are *not* the target applications of multi-band gigabit mesh networks.

- *Medium range (1-10 meters), without LOS guarantee*

The second set of the usages operate at medium range (1-10 meters, within one room) with reasonable probability of LOS but no guarantee. Even if there is a clear LOS path, movement of people or objects may easily disrupt the LOS path and so it is expected that NLOS links would be heavily used for these usages. Some examples include: i) Wireless HDMI (High-Definition Multimedia Interface) [11] to replace the HDMI cable between a TV and other video devices with a high rate 60 GHz link. This may be used in the home, a conference room or an auditorium. ii) Wireless docking in the office: The wireless docking station may be embedded into a display or monitor or may be a fixed standalone device. Mobile devices such as laptops or MID (Mobile Internet Devices) are connected wirelessly to the docking station when in the office. Other fixed devices such as keyboard, mouse, printer, and storage device (e.g., a hard drive) may be plugged into the docking station either via a wired interface (e.g., USB) or wirelessly. iii) Densely deployed 60 GHz WLAN: 60GHz APs mounted on the ceiling to deliver Gbps connectivity to many stations on an office floor.

Fig. 2 shows that it is feasible to deliver 1.2 Gbps PHY rate at the range of 10 meters under NLOS condition in 60 GHz. This may translate into 1 Gbps MAC rate if the MAC efficiency is 84% or more. But the actual range would greatly depend on the reflection surface materials and obstacles in the path. Lower MAC efficiency would also lower the achievable MAC performance below the targeted 1 Gbps. So it could still be challenging to deliver 1 Gbps MAC rate at 10 meter range in some indoor environment. A multi-hop path in 60 GHz may allow shorter distance for each single hop link and hence may increase the possibility of LOS for each single hop link. As the distance between the transmitter and the receiver increases, the likelihood of blockage with people moving about in the room also dramatically increases. While it is generally expected that 60 GHz radio will build in antenna tracking and re-training capability upon link breakage, concept such as multi-band mesh networks can help to provide additional diversity and robustness, and minimize the impact of 60 GHz link outage on the user experience.

- *Long range (>10 meters), NLOS*

As Fig. 2 shows that it may be very challenging to achieve transmission range beyond 10 meters for NLOS 60 GHz channel. Even more challenging than the distance is the obstruction caused by walls, doors, windows, furniture and other clutters commonly found in indoor environments. Such severe obstructions make it extremely difficult to cover a larger room or multiple rooms with a single hop 60 GHz link. For example, to provide full house coverage with multiple rooms, doors and floors, a multi-band mesh network is a must-have in order to meet the basic expectations of coverage. With multi-band mesh, it becomes possible to provide coverage comparable to lower bands and higher peak throughput in part of the house. The user may experience different levels of performance when moving about the house, but that is nothing new since the user would have experienced similar effect even with 802.11n products, albeit at a less profound level.

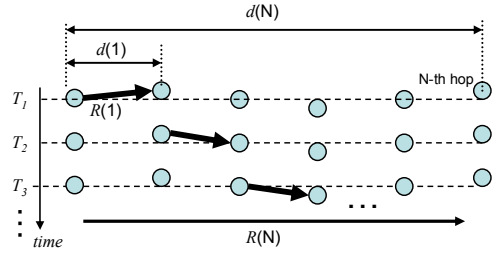
So in summary, the target usages for the multi-hop gigabit mesh networks are any usage that requires a distance of more than 1 meter without LOS guarantee.

### C. The Benefits of Multi-band Gigabit Mesh

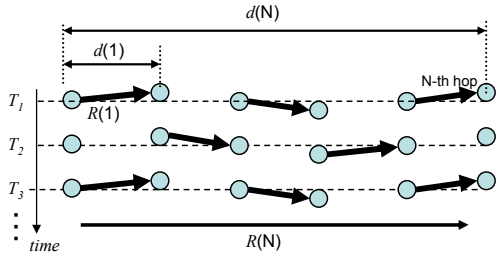
We've already touched on some of the benefits of multi-band gigabit mesh networks in the previous discussion; here we examine them more closely. The first benefit is the diversity gain, which may be due to the flexibility to switch between different bands, or the flexibility to choose different path within the 60 GHz mesh. Another major benefit is range and spatial reuse performance gain provided by the 60GHz mesh. While there may exist spatial reuse gain in any mesh network in any band, the spatial reuse gain in 60 GHz mesh networks is much more significant due to the nature of highly directional beamformed links in 60 GHz. Detailed simulation results are provided in the next section for an office mesh deployment scenario to illustrate the extend of spatial reuse gain.

- *Multi-band diversity gain by switching between bands*

1) *Data plane*: The multi-band aspect refers to the fact that the wireless system consists of multi-band wireless devices which are capable of operating in both the lower frequency bands (e.g., 2.4 and/or 5 GHz) and a higher frequency band (60 GHz). This combines the coverage and link robustness benefit offered by the lower frequency bands with the high rate benefit offered by the 60 GHz band as illustrated in Fig. 1 and Fig. 2. In other words, a data link would be carried over 60 GHz whenever possible (e.g., when the receiver is within reach at 60 GHz), and would fall back to 2.4/5 GHz when the 60 GHz link breaks.




(a) mesh without spatial reuse




(b) mesh with spatial reuse and perfect scheduling

Fig. 3. Illustrations of the link schedules in a linear topology multi-hop mesh with and without spatial reuse (source to destination is separated by $N$ hops)

2) *Control plane*: Diversity gain can also be achieved in the control plane as well as the data plane. Most of the 60 GHz WPAN specifications employ TDMA (time division multiple access) to access the shared medium efficiently between different devices [6][7][11]. Although TDMA has higher efficiency in utilizing the medium compared to a random access scheme, it needs a scheduler that schedules the traffic in the network and the scheduling control messages need to be exchanged between the scheduler and the devices before the actual data transmissions to their target devices. When the scheduling messages are lost due to the link breakage between the devices and the scheduler in the 60 GHz band, the devices can fall back to the lower bands and maintain the control plane with the scheduler and continue to exchange data with their target devices in the 60 GHz band with minimal interruption.

Another example to leverage multi-band integration in the control plane is to facilitate faster link establishment in one band when a link is already established in another band. For example, device discovery in 60 GHz can be time-consuming due to the directivity of 60 GHz link. One 60 GHz device looking for other 60 GHz devices needs to employ some kind of omni-communication because there is no prior knowledge of the existence and the direction of the other devices. Having a link already established in 2.4 or 5 GHz enables some information about the other multi-band devices to be communicated more quickly in 2.4 or 5 GHz band to speed up the discovery process in the 60 GHz band.

- *Multipath diversity gain by switching path within 60GHz*

There may exist different paths between a pair of 60GHz devices that intend to communicate. For example, there may be a single hop path and a 2-hop path between a transmitter and a receiver. If the single hop path is broken due to obstacles such as a person walking by, another path such as the 2-hop path may be taken to get around the obstructions. It is also possible that the 2-hop path is a concatenation of two direct LOS links

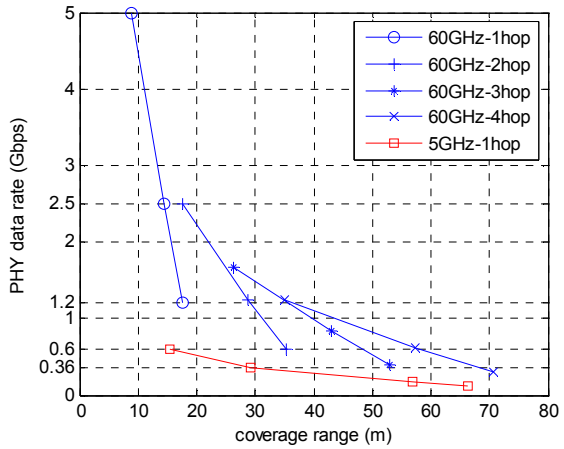



Fig. 4. Comparison of transmission ranges and data rates of 60 GHz multi-hop mesh and a single hop 5 GHz for NLOS environments using maximum transmit power per antenna

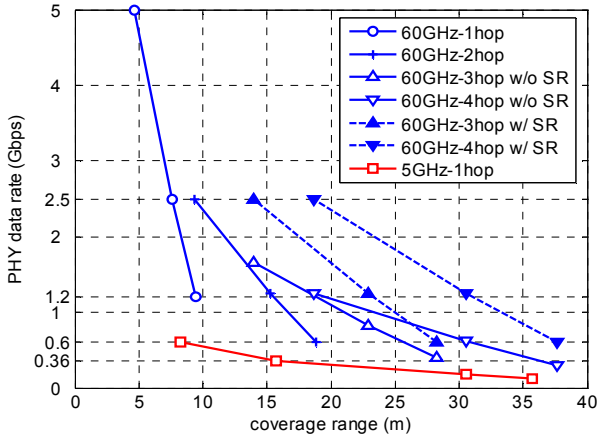
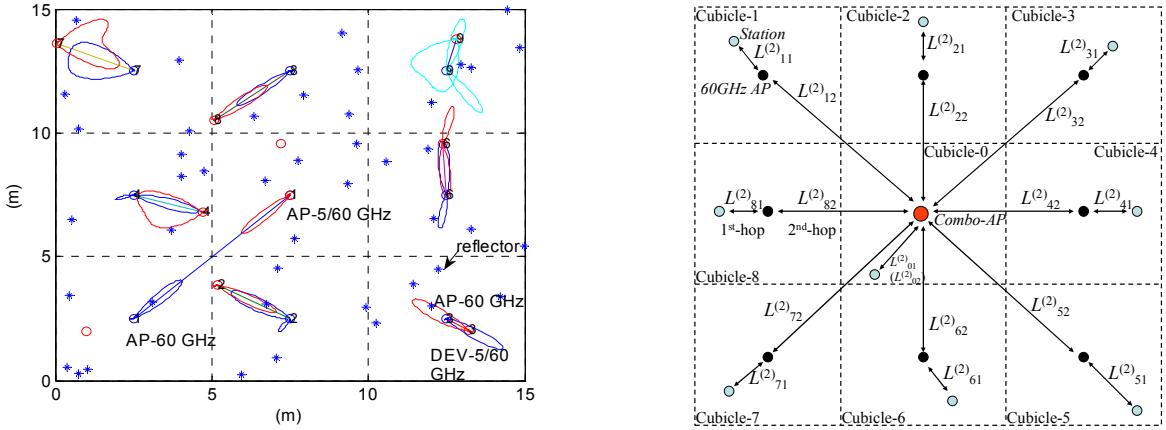



Fig. 5. Comparison of transmission ranges and data rates of 60 GHz multi-hop mesh with and without the spatial reuse (SR) and a single hop 5 GHz for NLOS environments using typical transmit power per antenna

and such a path is better than a single hop NLOS path. So having an option of taking the multi-hop path can also be considered as another aspect of the diversity gain

- *Range extension by the 60 GHz mesh*

Let's first consider the range extension benefit alone without considering spatial reuse gain achievable within the mesh. As illustrated in Fig. 3(a), for a linear topology of mesh without considering any spatial reuse possibility, if the number of hops between the source and the destination is $N$, the coverage range $d(N)$ increases by $N$; but the effective end to end data rate $R(N)$ decreases by $N$ [27], without taking into account the overhead associated with creating and maintaining the mesh network. Fig 4 shows a very simplistic comparison of multi-hop (2~4 hops) 60 GHz mesh links with a single hop 5 GHz link for NLOS environments in a linear topology network such as shown in Fig 3(a). Similar comparisons can be done for LOS (not shown here). The important finding from Fig. 4 is that a 4-hop 60 GHz link still significantly outperforms a 1-hop 5 GHz link while the range remains comparable. While a single hop link in 60 GHz can achieve 1.2 Gbps or above for a range of 18 meters, a multi-hop link can achieve 1.2 Gbps or above




(a) 9-cubicle network topology with 5 reflectors/cell

(b) network topology diagram of the 2-hop mesh network

Fig. 6. A network topology of 2-hop mesh network for a dense office

for a range of 36 meters, effectively doubling the range while still maintaining Gbps level performance. Note that two factors that would affect the mesh performance are not yet accounted for in Fig 4, namely, the MAC overhead in the mesh, and the spatial reuse due to highly directional beam formed links. While the MAC overhead may reduce the actual performance, on the other hand, the spatial reuse may significantly increase the performance, as discussed below.

- *Spatial reuse performance gain within the 60 GHz mesh*

With the spatial reuse benefit, the effective throughput of the 60 GHz multi-hop mesh can be improved significantly. Since a large number of antenna elements are used in the 60 GHz devices, a very narrow beam can be formed in a particular direction with very high gain and thus the mesh can have a good chance of having multiple active links simultaneously without interfering with each other. Assuming full spatial reuse and perfect scheduling in the mesh, for example, for the linear topology mesh as shown in Fig. 3(b), half of the segments of the multi-hop link can be active simultaneously regardless of the number of hops. Therefore, the effective end to end data rate $R(N)$ can be expressed as $R(N)=R(1)/2$ for the linear topology. Fig.5 compares the PHY data rate and the transmission range of the multi-hop (2~4 hops) 60 GHz mesh links with and without exploiting the spatial reuse benefit for NLOS environments using the typical total transmit power (10 dBm). Considering a partial spatial reuse gain, imperfect scheduling in a real network, and the actual deployment scenario, the effective end-to-end PHY rate of the linear topology mesh will fall somewhere in between the solid and dashed lines shown in Fig.5. More realistic bound of the spatial reuse is shown in the next section by simulation of an office WLAN deployment example.

## V. Simulation Study on Spatial Reuse of a 60 GHz Office Mesh

### A. Simulation Scenario and Assumptions

To further quantify the spatial reuse performance in a mesh for a realistic deployment scenario, let's consider the example of a multi-band WLAN mesh for office environments as shown in Fig. 6. The office area WLAN is comprised of devices in nine cubicles. There are two kinds of APs in this network: 60 GHz-only-APs, each covering a small area such as one cubicle; and a combo-AP that operates in both 5 and 60 GHz and hence can cover the larger area of all nine cubicles. The combo-AP also serves as the 60 GHz AP for the center cubicle. This combo-AP has a wired connection such as Gigabit Ethernet, acting as the connectivity gateway to the external network for all the devices in this office WLAN. The other eight 60 GHz-only APs do not require wired Ethernet connections on the ceiling as they can communicate with the combo-AP using 60 GHz links. Through these eight 60 GHz-only APs and one combo-AP, all the stations in the office area can form a 2-hop mesh network in 60 GHz.

The focus of our simulation study is on the spatial reuse aspect of the 60 GHz mesh. We implemented a 60 GHz MATLAB simulator that can quantify the spatial reuse gain in this 60 GHz office mesh network.

1) *Simulation scenario*: The network is comprised of nine cubicles each with dimensions of 5×5×3 (width × length × height in meters). All the APs are placed on the ceiling with the height of 3 meters. Each cubicle has one station that is placed randomly within its boundary and 1 meter above the floor level (assuming the stations are on a desk). For 2-hop data transmission in 60 GHz, a station in a cubicle first transmits data to its 60 GHz AP in the first hop, and in the second hop the 60 GHz AP communicates with the combo-AP. In order to make the system design simpler for 2-hop data transmission in the 60 GHz band, a station in one cube routes its data only through the 60 GHz AP above the station's cubicle.

2) *Channel model*: The channel model for the first hop between a station and its 60 GHz AP is modeled as NLOS (path loss exponent $n$=3.5 [19]) to capture the obstacles in the office environment, and the channel model for the second hop between the 60 GHz AP and the multi-band 5/60 GHz AP is modeled as LOS ($n$=2 [19]) since APs are installed on the ceiling and hence unlikely to encounter obstacles.

3) *Reflector model*: Very strong specular reflectors (such as metallic bookshelves or cabinets commonly found in the office) are modeled by randomly placing them in each cubicle between the height of 1~2 meters. It is assumed that a signal ray is reflected by the reflector only once without any signal

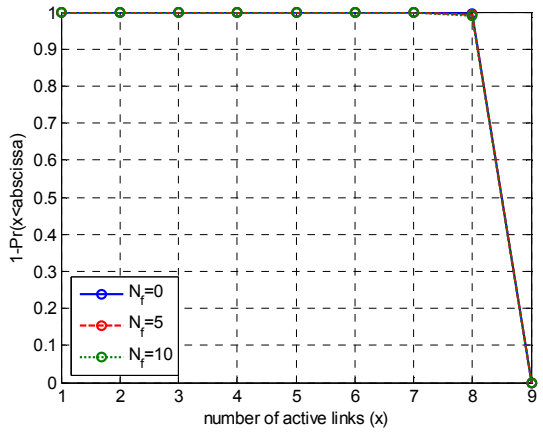



Fig. 7. Distribution of the number of simultaneous active links in 60 GHz 2-hop mesh network (all the stations and the APs use the 6x6 square array antenna)

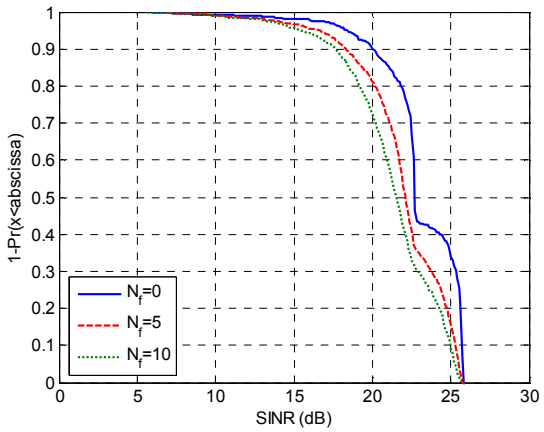



Fig. 8. Distribution of the SINR of the 2nd hop links between one of the 60 GHz AP and the combo-AP for the two-hop mesh (all the APs using the 6x6 square array antenna)

attenuation to capture only the first order reflections. By varying the number of reflectors ($N_f$= 0, 5, and 10) in a cubicle the effect of a multi-path environment can be simulated.

4) *Antenna model:* All the APs are equipped with a 6x6 square array antenna (36 antenna elements) and the stations are equipped with a 6x6 or a 4x4 square array antenna considering the fact that there are more limitations such as form factors and cost for the stations than the APs. The adjacent antenna elements are separated by a half wavelength. The total transmit power of each array antenna is fixed to 10 dBm. Each antenna element is assumed to be an ideal isotropic radiator. The orientation of the array antenna is randomly rotated in the x, y, and z-axis.

TABLE IV
MCS AND DATA RATES [11]

| MCS mode | MCS1 | MCS2 | MCS3 |
|---|---|---|---|
| *Modulation* | QPSK | QPSK | 16 QAM |
| *Code rate* | 1/2 | 2/3 | 2/3 |
| *Data rate* | 0.952 Gbps | 1.904 Gbps | 3.807 Gbps |

5) *PHY Data rates*: Depending on the SINR of each link, the data rate can be chosen from three different MCS modes shown in Table IV. For the simulations, the SINR thresholds are set to 5.5 dB, 13 dB, and 18 dB for MCS1=0.952 Gbps, MCS2=1.904 Gbps, and MCS3=3.807 Gbps, respectively to have BER lower than 10e-5.

## B. Metrics to Measure Spatial Reuse

Several metrics are used to measure spatial reuse in the 60 GHz mesh. The first metric is the number of simultaneous active links in the network.

The notation $L_{ij}^{(2)}$ is used to denote the link in this 2-hop office mesh network, the superscript $^{(2)}$ denoting the 2-hop path between the stations and the combo-AP, with the cubicle index $i$ = 0,1,…,8 denoting the location of the station where the data communication is initiated or destined, and the hop index $j$=1,2 distinguishing if it is the first or the second hop. This is shown in Fig. 6(b). Note that for the center cubicle there is only a single hop from the station to the combo-AP, while for the other cubicles there will be a 2-hop path. Just for the convenience of notation, the center cubicle is assigned with

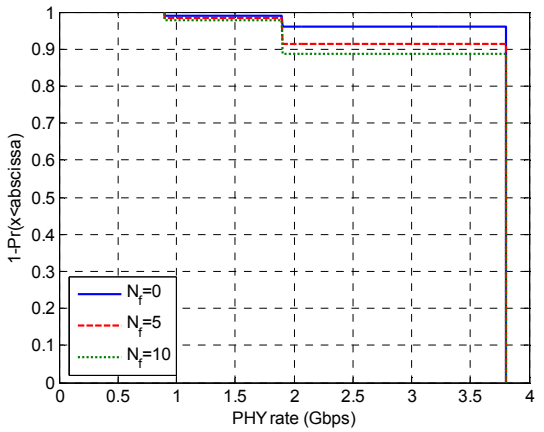



Fig. 9. Distribution of the PHY rate of the 2nd hop link between one of the 60 GHz AP and the combo-AP for the two-hop mesh (all the APs using the 6x6 square array antenna)

index $i$=0; and so effectively $L_{01}^{(2)} = L_{02}^{(2)}$. Therefore, there are totally 17 distinct links in this office network, denoted as $\{ L_{ij}^{(2)} : i = 0,1,\ldots,8 ; j=1,2 \}$. For the rest of this subsection, the set of links between the stations and the 60 GHz-APs, $\{L_{i1}^{(2)} : i = 0,1,\ldots,8\}$, is referred to 1st hop links, and the set of links between the 60 GHz APs and the combo-AP, $\{L_{i2}^{(2)} : i = 0,1,\ldots,8\}$, referred to 2nd hop links.

Note that only one of the 2nd hop links can be active at any given time. So theoretically there cannot be more than 9 simultaneously active links in this mesh network, and the only possible set of 9 simultaneously active links is $\{L_{i1}^{(2)}: i = 0,1,\ldots,8\}$ . If the number of active links is no more than 1, there is effectively no spatial reuse in the network. So the number of simultaneous active links is a very intuitive metric to indicate the degree of spatial reuse. But it is not the most accurate one as it does not reflect the quality of these active links. Therefore another metric, the aggregated end to end PHY throughput in the mesh, is introduced as a more accurate measurement of the mesh performance at the PHY level.

Both metrics are presented in the form of CCDF (Complementary Cumulative Distribution Function) curves, generated from Monte Carlo simulation of 1000 runs. For each run, the stations and the reflectors are placed randomly within

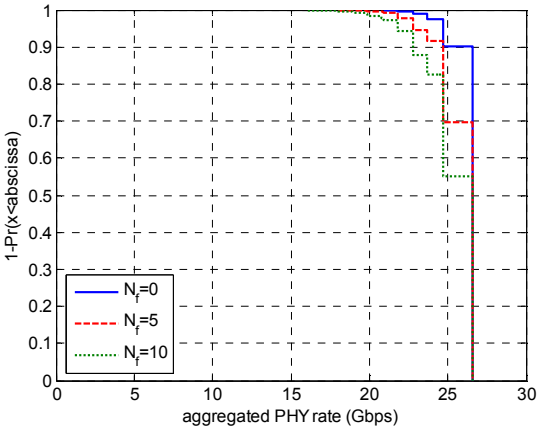



Fig. 10. Distribution of the aggregated PHY rate of the 1st hop links between the stations and the 60 GHz APs for the two-hop mesh (all the stations and the APs using the 6x6 square array antenna)

the cubicles and the antenna array is oriented randomly. It is assumed that a link can be active only if the SINR of the link is above the lowest SINR threshold (i.e. 5.5 dB) and if the link does not lower the SINR of the preexisting active links below the lowest SINR threshold.

### C. *Simulation Results with 6x6 Antenna Array*

Fig. 7 shows the distribution of the number of simultaneous active links in the network with all the APs and the stations using the 6x6 square antenna array. The way to determine the number of simultaneous active links in each run of simulation is to first randomly pick a 60 GHz AP to form a 2nd hop link with the combo-AP; then determine which and how many of the 1st hop links can be activated successfully with the chosen 2nd hop link. The order of activation among the 1st hop links is random in each run. The simulation results show that eight links are almost always active and the number of reflectors ($N_f$ = 0, 5, and 10) does not cause much change on that. Remember that there can be only one 2nd hop link active at any given time, say $L_{12}^{(2)}$. This means all the rest of 7 cubicles (index $i$ = 2 to 8) can still be actively transmitting or receiving from their 60 GHz-only AP at the same time. Fig. 7 shows that extremely high degree of spatial reuse is obtained in this scenario. This is possible because all the APs and stations are using an antenna array with a large number of antenna elements (i.e. 36) which provides very high transmit and receive beamforming gain (~30 dB) and very narrow beam width, the distance between a station and the AP in its cell is very close, and the beams pointing to or from the APs on the ceiling do not interfere with each other.

While the number of reflectors does not impact the number of active links, it does affect the quality of these links. Fig. 8 shows the SINR distribution of the 2nd hop links $\{L_{i2}^{(2)} : i = 0,1,\ldots,8\}$ (that is, links between one of the 60 GHz APs and the combo-AP on the ceiling). As the number of reflectors increases, interference from the other active first hop links in the network increases and thus the SINR of the 2nd hop link between the APs decreases. The SINR distribution in Fig. 8 is converted into the distribution of the PHY rate of the 2nd hop links in Fig. 9. The results show that for the moderate number of reflectors ($N_f$ =5), the 2nd hop links can maintain the highest MCS (i.e. 3.8 Gbps) for more than 90% of the time. Since there

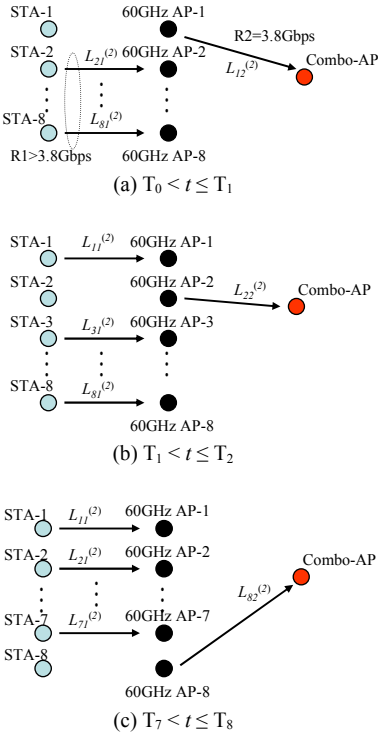



Fig. 11. Illustration of the two-hop mesh operation over the time $T_0 < t \leq T_8$ achieving the end-to-end throughput close to the maximum PHY rate of the 2nd-hop link

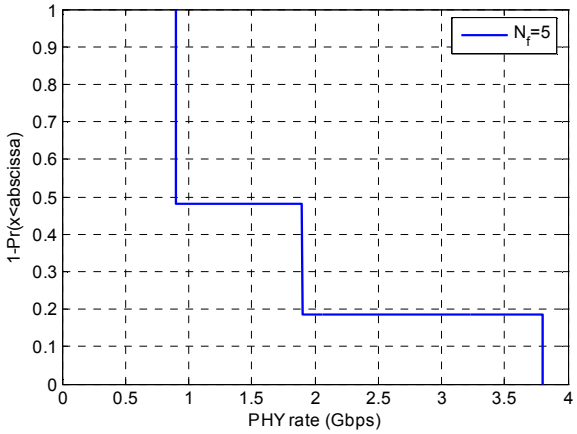



Fig. 12. Distribution of the PHY rate of the one-hop link between one of the stations and the 60 GHz AP both using the 6x6 square array antenna

can be only one active 2nd hop link at any given time, Fig. 9 effectively shows the aggregated throughput of the 2nd hop link at the combo-AP in the network.

Let's now consider the aggregated PHY rate of 1st hop links in Fig. 10. The aggregated PHY rate is defined as the sum of all the active links' PHY rate. Similar to the situation with the 2nd hop, as the number of reflectors increases the aggregated PHY rate of the 1st hop links decreases due to increased interference between each other. However, the aggregated PHY rate for the 1st links is still extremely high, over 20 Gbps most of the time. This is because of the extremely high degree of spatial reuse among the 1st hop links.

Now consider the end to end aggregated PHY rate in the 2-hop mesh. Fig. 7 and Fig 9 shows that the 2nd hop link can

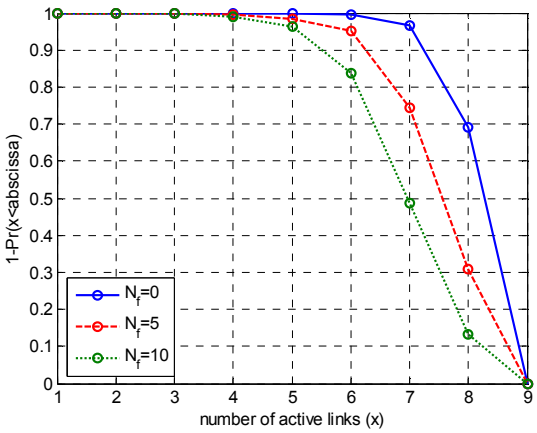



Fig. 13. Distribution of the number of simultaneous active links in 60GHz 2-hop mesh network (the stations with 4x4 square array antenna and the APs with 6x6 square array antenna)

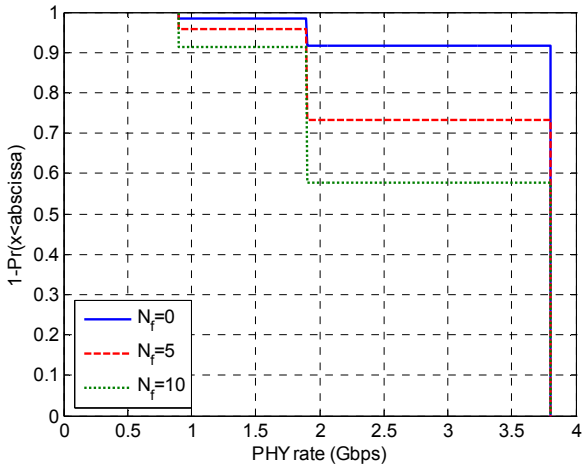



Fig. 15. Distribution of the PHY rate of the 2nd hop link between one of the 60 GHz AP and the combo-AP for the two-hop mesh (all the APs using the 6x6 square array antenna)

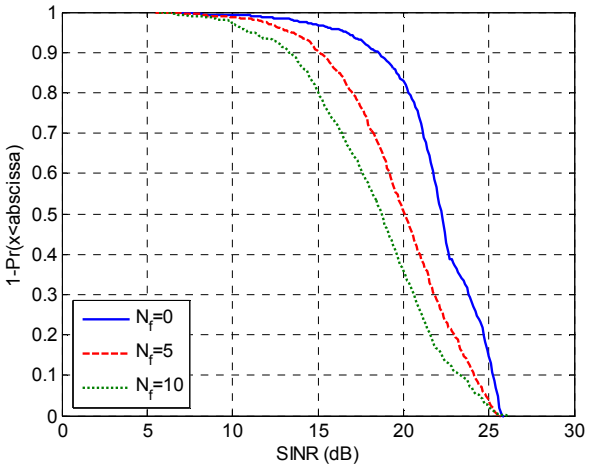



Fig. 14. Distribution of the SINR of the 2nd-hop link between one of the 60 GHz AP and the combo-AP for the two-hop mesh (all the APs using the 6x6 square array antenna)

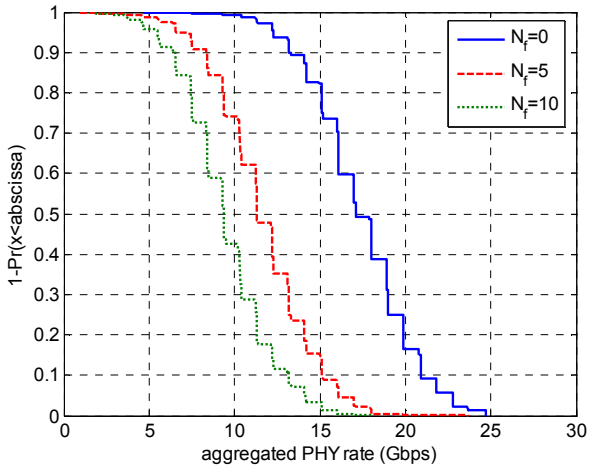



Fig. 16. Distribution of the aggregated PHY rate of the 1st hop links between the stations and the 60 GHz APs for the two-hop mesh (the stations with 4x4 square array antenna and the APs with 6x6 square array antenna)

maintain the highest data rate (3.8Gbps) more than 90% of the time while seven other 1st hop links are active simultaneously. This is illustrated in Fig. 11. In Fig. 11(a), at one time instance ($T_0 < t \le T_1$), the seven 1st hop links $\{L_{i1}^{(2)} : i = 2,3,4,5,6,7,8\}$ may be active simultaneously with the 2nd hop link $L_{12}^{(2)}$. Fig. 11(b) shows the next time instance ($T_1 < t \le T_2$) when $\{L_{i1}^{(2)} : i = 1,3,4,5,6,7,8\}$ and $L_{22}^{(2)}$ 2nd-hop are the simultaneously active links. Fig. 11(c) shows another time instance ($T_7 < t \le T_8$) when $\{L_{i1}^{(2)} : i = 1,2,3,4,5,6,7\}$ and $L_{82}^{(2)}$ are the simultaneously active links. Since the aggregated throughput of the 1st hop links (shown in Fig. 10) is much higher than the 2nd hop (shown in Fig. 9), and one of the 2nd hop links can be almost always active simultaneously with seven other 1st hop links (shown in Fig. 7), the end to end throughput is basically determined by the 2nd hop link throughput. So the end to end throughput distribution should look like the throughput distribution of the 2nd hop shown in Fig. 9 as well. That is, 90% of the time the end to end throughput can reach 3.8 Gbps.

It is remarkable that a two-hop mesh network reaches the highest data rate (3.8 Gbps) that is ever achievable by a single link, thanks to the high degree of spatial reuse. It can be shown that such 2-hop network actually outperforms a one hop network where there is only one 60 GHz AP serving all the stations in the 9 cubicles. Fig. 12 shows the distribution of the PHY data rate of such single hop links for $N_f$=5. The results show that such one hop link can achieve 3.8 Gbps only 19% of the time, which is much worse than the two-hop mesh. This is because the path loss exponent is higher between the station and the AP due to more obstacles compared to the 2nd hop link between the 60 GHz AP and the combo-AP in the two-hop case and the distance between the station and the AP is longer.

Another remarkable insight one may gain from this example is that even though the aggregated PHY rate for the 1st hop links can be over 20 Gbps most of the time, it does not translate directly into the end to end throughput for the mesh, because the 2nd hop is the bottleneck of the network. This is typical of a network with star or tree like topology, and congestion and flow control is a well known and well studied

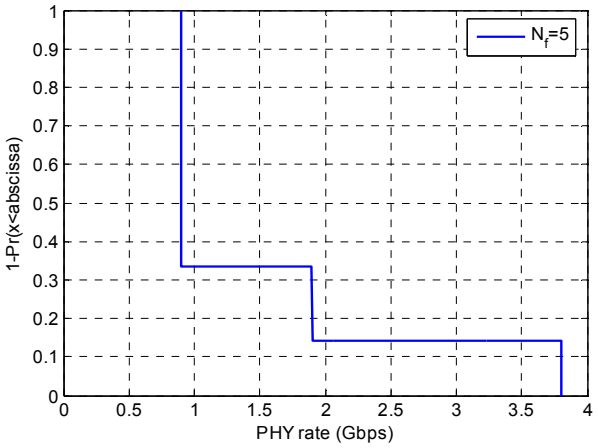



Fig. 17. Distribution of the PHY rate of one-hop link between one of the stations and the 60 GHz AP (the stations with 4x4 square array antenna and the APs with 6x6 square array antenna)

problem in mesh network literatures [31-34]. However, this problem becomes even more severe in our example because of the spatial reuse with highly directional links in 60 GHz. Without proper congestion control, much of the huge spatial reuse potential is wasted. So this example demonstrates the need to further study effective congestion and flow control that can maximize the return of spatial reuse.

### D. Simulation Results with 4x4 Antenna Array at the Stations

Now suppose the stations in the two-hop network have only 16 antenna elements (a 4x4 square array antenna), which might be more realistic considering the small form factor of the stations and the cost constraint. As the number of antenna elements decreases, not only the link quality degrades due to the decreased beamforming gain but also interference to and from the other active links increases due to the wider beam width.

Fig. 13 shows the effect of the decreased number of antenna elements at the stations on the spatial reuse gain. The spatial reuse gain decreased significantly comparing to Fig. 7 where the stations are equipped with 36 antenna elements. However, more than five links can still be active simultaneously more than 90% of the time, which is still very high spatial reuse gain.

Fig. 14 and Fig. 15 show the effect of the decreased number of antenna elements on the SINR and the PHY rate of the 2nd hop links. Comparing to Fig. 8 and Fig. 9, although more interference from the stations degrades the quality of the link, the link can still support the highest data rate for more than 70% of the time with a moderate number of reflectors ($N_f$=5).

Fig. 16 shows the aggregated PHY rate of the 1st hop links. Comparing to Fig. 10, the PHY rate is significantly lower with 4x4 antenna array at the stations. However, the results show that the aggregated PHY rate is still sufficiently higher than the 2nd hop PHY rate in Fig. 15. This is because 100% of the time at least 3 links (i.e., one 2nd hop link and two 1st hop links) can be active simultaneously as shown in Fig. 13, and so the end to end PHY throughput is still determined by the 2nd hop. So the end of end PHY throughput with 4x4 antenna array at the stations has similar distribution as shown in Fig. 15.

The above results are compared to the one-hop scenario in Fig. 17 and it shows that the one-hop case also suffers from the decreased beamforming gain and now the link can support the highest data rate for only 14% of the time (vs. 73% of the time with two-hop mesh in Fig. 15). So even with the 4x4 antenna array, a 2-hop mesh still outperforms the one hop network in this office WLAN example and maintains the end to end PHY throughput close to the highest PHY rate of the 2nd hop for majority of the time. This shows that the stations can have less number of antenna elements than the APs but still enjoy the end to end throughput performance benefit by employing mesh.

## VI. Previous Work

Multiple research projects in Europe [8, 9] have been focusing on the design of a holistic wireless system that provides higher performance for WLAN. One such example is the European Information Society Technologies (IST) Broadway project [8, 16] which targeted specifically for a WLAN with the hybrid dual frequency system design concept based on HIPERLAN/2 at 5 GHz and a fully ad-hoc extension at 60 GHz. The usage scenarios envisioned include hotspot coverage, public Internet access, high density dwellings and flats deployment, corporate and campus environments. However, IST limited its bandwidth in the 60 GHz band to no more than 240 MHz and hence the maximum data rate achievable was only 720 Mb/s, substantially below 1 Gbps.

Conceptually what it is proposed in this paper is similar to the approach taken by the IST Broadway project but there are three important distinctions that lead to a substantially different solution: i) Broadway is based on HIPERLAN/2 and there is very tight integration of 5 GHz and 60 GHz at the RF front end. Our work is based on the 802.11 MAC to maximize reuse and integration with existing 802.11 solutions at 2.4/5 GHz. ii) Broadway limited its channel width at 60 GHz band to no more than 240 MHz and hence the max link data rate was only 720 Mb/s. The latest PHY proposals in both ECMA TC 48 [7] and IEEE 802.15.3c [8] use a much wider channel (close to 2 GHz) for obtaining multi-Gbps PHY rates. Similar assumptions are taken here at the PHY to ensure multi-Gbps data rate capabilities. The MAC efficiency can be deeply affected by the operating PHY rate [1, 18] and so MAC throughput does not always scale linearly with the PHY rate. It is important to re-examine and sometimes re-design the MAC protocol when the PHY rate is substantially increased. iii) It was assumed in Broadway that only infrastructure devices such as APs used directional antenna at 60 GHz and all stations used an omni-directional antenna. Our work assumes that it is feasible to employ directional antennas for stations such as laptops, or MIDs. This allows a significant spatial reuse benefit and can affect the design substantially in both the PHY and MAC layers.

## VII. Design Challenges of Multi-band Gigabit Mesh

This section presents high level considerations of the system design concept of multi-band Gigabit mesh networks and the new research challenges within that framework.

### A. Integration Architecture to Support Seamless Session Switch

The concept of multi-band integration is straightforward but the design and implementation are non-trivial. One of the important design questions is where in the protocol stack such

integration should happen - at the RF front end, baseband, MAC, or above the MAC?

The answer partly depends on how similar or different the radio stack (antenna, PHY, MAC, etc.) would look like across the bands. Given the very different channel properties of the 60 GHz band, the antenna, RFIC and baseband design for the 60 GHz would be quite different from that of 2.4 or 5 GHz band [38]. We also have strong reasons to believe that the media access mechanism for 60 GHz would be quite different from the CSMA/CA based media access mechanism for Wi-Fi in the 2.4 and 5 GHz bands, as explained below.

First and foremost, the current design of CSMA/CA assumes that devices in the same physical proximity can carrier sense and overhear each other because of the omni-directional antenna typically used in 2.4 and 5 GHz bands and so all communications can be assumed to be broadcast at the physical level. This is no longer the case in 60 GHz, because high gain directional antenna would be employed in order to reach decent range. This directivity fundamentally violates the assumption of CSMA/CA and so direct reuse does not make sense.

Another reason that we may consider to modify the 802.11 MAC is that current 802.11 MAC does not provide strong QoS guarantee, which may be acceptable for Internet connectivity type of applications but not acceptable for high performance media applications such as wireless display. Media access such as TDMA (Time Division Multiplex Access) that can provide better QoS assurance is probably needed. While TDMA typically is used successfully for licensed band applications such as cellular networks, it is relatively unproven for unlicensed band applications such as 60 GHz. The main difficulty with TDMA in unlicensed band is to cope with the interference from independent overlapping networks without causing instability in the networks. The fact that 60 GHz links typically employ directional communication between devices can somewhat mitigate such a problem as directional communication helps lesson the probability of interference and hence improves space reuse as evidenced from our simulation results shown in this paper.

For these reasons, we believe media access mechanism for 60 GHz would be quite different from that of existing Wi-Fi at 2.4 and 5 GHz. This begs the question of how these different media access mechanisms be reconciled or integrated in the multi-band framework. Such reconciliation needs to be investigated carefully in order to design a reasonable integration architecture that can support seamless session switch between different bands.

### B. *Radio concurrency*

Depending on how tightly the 60 GHz radio and 2.4/5 GHz radio are integrated, a multi-band device may or may not be able to operate in different bands concurrently. Therefore it is important to have the flexibility of allowing both configurations to function in the network. If two radios can be used concurrently, it opens up the possibility of using both bands to further optimize the performance for the device and the network beyond what a single radio can provide. On the other hand, full concurrent operations may increase hardware cost and may consume too much power for the mobile devices. If two radios are not to operate simultaneously for a long period of time, then how and when to switch from one band to another is also an interesting question to consider.

### C. *Multi-hop Multi-band Mesh Challenges*

Some of the issues with directional ad hoc networks such as medium access control, neighbor discovery and routing have been well studied [22-30], albeit for lower frequency bands. It is necessary to reevaluate the ideas and applicability for higher frequency bands, with realistic antenna patterns and higher PHY rates. Some concepts might be more readily applicable than others. For example, congestion control [31-34] is a well recognized problem in multi-hop ad hoc networks. As demonstrated in our office WLAN example in Section V, the congestion problem could be even more pronounced for some topology like star- or tree- like networks, and the un-even spatial reuse gain in different part of the network may actually worsen the congestion at certain point of the network. How to address congestion and flow control in such highly directional networks is a new research topic. The concept of network coding [35] has shown promise to combat performance issues such as congestion. But such a concept may be challenging to apply in a directional mesh because network coding leverages omni-directional broadcast which is a very expensive operation at 60 GHz; so further study might be needed in that area as well.

The topic of multi-channel ad hoc networks has been studied somewhat in the past [36-37], however, most of the work assumes homogeneous channels within the same band. Multi-band mesh works across characteristically very different bands, and so imposes a new set of problems to solve but we have seen very little work done on this yet.

Power consumption is another aspect that needs to be carefully studied in the context of multiband mesh. For example, as pointed out earlier, radio concurrency may have negative impact on power while boosting the performance. Another important factor to consider is how much power the radio consumes when operating in different band.

## VIII. CONCLUSION AND FUTURE WORK

The industry is positioning 60 GHz radio as a high performance radio that is capable of delivering gigabit performance in a wide range of usage scenarios. 60GHz represents a technological opportunity This means that 60 GHz radio has to live up to expectations similar to Wi-Fi since most users are familiar with that experience. In this paper we point out that one of the key challenges in meeting those expectations is to improve range and robustness at 60 GHz. This paper proposes the system design concept of multi-band gigabit mesh networks to meet that challenge and quantify its potential benefits in range extension and spatial reuse with analysis and simulation results. The integration of 60GHz radio with the existing 2.4/5GHz band WiFi radio presents an opportunity to provide a unified technology for both gigabit WPAN and WLAN, thus further reinforcing the technology convergence that is already underway with the widespread adoption of Wi-Fi technology [40]. Much more work need to be done in order to prove the feasibility of this concept with a detailed protocol design. Our current analysis and simulation do not take into account the MAC overhead associated with the mesh creation and maintenance, and we would like to take that into account in our further study to tighten the performance bound. We also want to continue to investigate

the topics discussed in the last section to provide a complete solution for multi-band gigabit mesh networks.

## REFERENCES

[1] L. L. Yang and M. Park, "Applications and Challenges of Multi-band Gigabit Mesh Networks", MESH 2008 (Best paper award), Cap Estera, France, August 2008

[2] M. Park, C. Cordeiro, E. Perahia, and L. L. Yang, "Millimeter-Wave Multi-Gigabit WLAN: Challenges and Feasibility", Invited paper for IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), September 2008.

[3] C. Park, T. S. Rappaport, "Short-Range Wireless Communicaitons for Next-Generation Networks: UWB, 60 GHz Millimeter-Wave WPAN, and ZigBee", IEEE Wireless Communications, August 2007

[4] N. Guo, R. C. Qiu, S. S. Mo, K. Takahashi, "60-GHz Millimeter-Wave Radio: Principle, Technology and New Results", EURASIP Journal on Wireless Communications and Networking, 2007

[5] P. Smulders, "Exploiting the 60 GHz Band for Local Wireless Multimedia Access: Prospects and Future Directions", IEEE Communications Magazine, Jan. 2002

[6] ECMA TC48 – High Rate Short Range Wireless Communications, http://www.ecma-international.org/memento/TC48-M.htm

[7] IEEE 802.15 WPAN Millimeter Wave Alternative PHY Task Group 3c (TG3c), http://www.ieee802.org/15/pub/TG3c.html

[8] European Information Society Technologies (IST) Broadway Project, http://www.ist-broadway.org

[9] WIGWAM – Wireless Gigabit with Advanced Multimedia Support, http://www.wigwam-project.de/

[10] IEEE 802.11 Very High Throughput Study Group (VHT SG), http://grouper.ieee.org/groups/802/11/Reports/vht_update.htm

[11] WirelessHD 1.0 http://www.wirelesshd.org/

[12] S. K. Moore, "Cheap chips for next wireless frontier," *IEEE Spectrum*, vol. 43, 2006.

[13] H. Xu, V. Kukshya, and T. Rappaport, "Spatial and Temporal Characteristics of 60 GHz Indoor Channels," *IEEE Journal in Selected Areas in Communications*, April 2002.

[14] "ECMA TC48 draft standard for high rate 60 GHz WPANs White Paper", http://www.ECMA-international.org/activities/Communications/tc48-2008-024-Rev1.doc, Feb 2008.

[15] G. Fettweis and R. Irmer, "WIGWAM: System Concept Development for 1 Gbit/s Air Interface", In 14th Wireless World Research Forum (WWRF 14), July 2005

[16] IST-2001-32686 BROADWAY "WP 1, D2 – Functional System Parameters description", http://www.ist-broadway.org/documents/deliverables/broadway-wp1-d2.pdf

[17] A. Myles, R. de Vegt, "Wi-Fi (WFA) VHT Study Group Usage Models", IEEE 802.11-07/2988r4, 2007

[18] M. Park, "Analysis on IEEE 802.11n MAC Efficiency," IEEE 802.11-07/2431r0

[19] J. Kivinen, "60-GHz Wideband Radio Channel Sounder," IEEE Trans. on Instrumentation and Measurement, vol. 56, no. 5, Oct. 2007

[20] T. S. Rappaport, Wireless Communications: Principles and Practices. 2nd Edition. New Jersey: Prentice Hall, 2002.

[21] G. Li and L. L. Yang, "On Utilizing Directional Antenna in 802.11 Networks: Deafness Study", The Second International Workshop on Wireless Personal and Local Area Networks (WILLOPAN), 2007.

[22] G. Li, L. L. Yang, W. S. Conner, and B. Sadeghi, "Opportunities and Challenges for Mesh Networks Using Directional Antennas", First IEEE Workshop on Wireless Mesh Networks (WiMesh), 2005

[23] R. Ramanathan, "On the performance of Ad hoc networks with beamforming antennas," ACM MobiHoc 2001.

[24] R. R. Choudhury, X. Yang, R. Ramanathan and N. H. Vaida, "Using directional antennas for medium access control for ad hoc networks," Mobicom 2002.

[25] M. Takai et al., "Directional virtual carrier sensing for directional antennas in mobile ad hoc networks," ACM MobiHoc, June 2002.

[26] H. Gossain, C. Cordeiro, and D. Agrawal, "MDA: An Efficient Directional MAC Scheme for Wireless Ad Hoc Networks," *in IEEE Globecom*, November 2005.

[27] A. Nasipuri, S. Ye, J. You, and R. Hiromoto, "A MAC Protocol for Mobile Ad Hoc Networks using Directional Antennas," in IEEE WCNC, Sep. 2000.

[28] S. Yi, Y. Pei and S. Kalyanaraman, "On the capacity improvement of ad hoc wireless networks using directional antennas," MobiHoc2003.

[29] A. Spyropoulos and C. S. Raghavendra, "Capacity bounds for ad-hoc networks using directional antennas," ICC'2003.

[30] A. K. Saha and D. B. Johnson, "Routing improvements using directional antennas in mobile ad hoc networks," Globecom 2004.

[31] V. Gambiroza, B. Sadeghi, and E. Knightly, "End-to-end performance and fairness in multihop wireless backhaul networks," in ACM Mobi-Com'04, September 2004.

[32] B. Sadeghi, A. Yamada, A. Fujiwara, L. L. Yang, "A Simple and Efficient Hop-by-hop Congestion Control Protocol for Wireless Mesh Networks" 2nd Annual Intern. Wireless Internet Conference (WICON), Boston, August, 2006

[33] A. Yamada, A. Fujiwara, L. L. Yang, and B. Sadeghi, "EDCA Based Congestion Control for WLAN Mesh Networks", VTC spring 2006. Melbourn, Austrlia.

[34] Y. Yi and S. Shakkottai, "Hop-by-Hop Congestion Control over Wireless Multi-hop Network," IEEE INFOCOM, March 2004..

[35] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, "XORs In The Air: Practical Wireless Network Coding," ACM SIGCOMM, 2006.

[36] P. Bahl, R. Chandra, and J. Dunagan, "Ssch: slotted seeded channel hopping for capacity improvement in IEEE 802.11 adhoc wireless networks", MobiCom 2004.

[37] P. Kyasanur and N. H. Vaidya, "Capacity of Multi-channel Wireless Networks: Impact of Number of Channels and Interfaces." MobiCom 2005.

[38] C. H. Doan, S. Emami, D. A. Sobel, A. M. Niknejad, and R. W. Brodersen, "Design Considerations for 60 GHz CMOS Radios", IEEE Communications Magazine, December.

[39] S. K. Yong and C. Chong, "An Overview of Multigigabit Wireless Through Millimeter Wave Technology: Potentials and Technical Challenges", EURASIP Journal on Wireless Communications and Networking, vol. 2007, article ID 78907.

[40] L. L. Yang, "60GHz: Opportunity for Gigabit WPAN and WLAN Convergence", ACM SIGCOMM Computer Communication Review, January 2009.

# Time Dependent Lévy Flights Models for Internet Traffic

György Terdik and Tibor Gyires
*University of Debrecen, Hungary and Illinois State University, USA*
*Terdik.Gyorgy@inf.unideb.hu, TBGyires@ilstu.edu*

## Abstract

*Measurements of local and wide-area network traffic in the 90's established the relation between burstiness and self-similarity of network traffic. Several papers demonstrated that the widely used Poisson based models could not be applied for the past decade's network traffic. Recent papers have questioned the direct applicability of these results in networks of the new century. Some authors of these papers demand the revision of previous assumptions on the Poisson traffic models. They argue that as newer and newer network technologies are implemented and the amount of Internet traffic grows exponentially, the burstiness of network traffic might cancel out due to the huge number of aggregated traffic flows. Some results are based on analyses of high-speed Internet backbone links and other traffic traces. We analyzed the same traffic traces and applied novel methods to characterize them in terms of packet interarrival time. We demonstrate that the series of interarrival times in the 2003 traces is still close to a self-similar process. Since then, new traffic traces have been made public, including ones captured from OC-192 links of the Internet backbone in 2008. We also compare the 2008 traffic traces with the ones captured in 2003 and apply our analytical methods to illustrate the tendency of Internet traffic burstiness in recent years. We found that the burstiness of the interarrival times decreased significantly compared to earlier traces.*

***Index Terms*— Internet traffic; burstiness; Lévy Flights.**

## 1. Introduction

Network congestion can be caused by several factors. The most dangerous cause of congestion is the burstiness of the network traffic. Recent results make evident that high-speed network traffic is more bursty and its variability cannot be predicted as assumed previously. It has been shown that network traffic has similar statistical properties on many time scales. Traffic that is bursty on many or all time scales can be described statistically using the notion of self-similarity. Self-similar traffic has observable bursts on all time scales.

One of the consequences of burstiness is that combining the various flows of data, as it happens for example in the Internet, does not result in the smoothing of traffic. Measurements of local area network traffic [20] and wide-area network traffic have proved [21] that the widely used Markovian process models could not be applied for today's network traffic. If the traffic were Markovian process, the traffic's burst length would be smoothed by averaging over a long time scale contradicting with the observations of today's traffic characteristics. Combining bursty data streams will also produce bursty combined data flow. Various papers discuss the impact of the burstiness on network congestion [1], [3] and [8]. Their conclusions are that congested periods can be quite long with losses that are heavily concentrated.

The self-similarity of network traffic was observed in numerous papers, such as [3], [9], [22] and [25]. These and other papers showed that packet loss, buffer utilization, and response time were totally different when simulations used either real traffic data or synthetic data that included self-similarity [10], [11].

Papers, such as [16] and [31] challenge the direct applicability of these results for today's network traffic. They argue that traditional Poisson models can be used again to characterize the aggregate traffic flow of multiplexed large numbers of independent sources in the Internet backbone [17], [30]. Their explanation is that as the amount of Internet traffic grows dramatically mainly due to the implementation of fiber optic backbone links, the burstiness of network traffic might cancel out as a result of the large number of multiplexed packet flows. The paper describes the analyses of packet traces captured in the Internet backbone. The authors found that packet arrivals followed the Poisson distribution at sub-second time scales, appeared to be nonstationary at multi-second time scales, and the same packet trace showed evidence of long-range dependence at scales of seconds and above.

By the end of 90's, many previous works also analyzed the burstiness and the correlation structure of Internet traffic in various time scales according to the behavior of the Transmission Control Protocol (TCP) in terms of timeouts, congestion avoidance, self-clocking, etc. The paper [5] applied a wavelet-based multiresolution tool to analyze the scaling behavior of Internet traffic on short time scales. This paper was one of the first works showing evidence that Internet traffic could be analyzed by a multifractal model. Recent studies in [[7], [2], and [24]] have also proved that Internet traffic exhibits not only monofractal properties, but also a multifractal nature. Actual measurements have demonstrated that low-aggregate network traffic can have more complex properties than assumed previously.

The paper [12] illustrated that short time scale burstiness was independent of the TCP flow arrival process and showed that in networks with light traffic, correlations across different flows did not have an effect on the short scale burstiness. Internet traffic was classified in alpha and beta flows in the paper [28]. It was shown that large transfers over high-capacity links, called alpha flows, produced non-Gaussian traffic, while the beta flows, low-volume transmissions, produced Gaussian and long-range dependent traffic. Long sequence of back-to-back packets can cause significant correlations in short time scales. The reasons of sending long back-to-back packets in TCP or UDP sources were analyzed in [14], such as UDP message segmentation, TCP slow start, lost ACKs, etc. The same authors in [15] identified the actual protocol mechanisms that were responsible for creating bursty traffic in small time scales. It was shown that TCP self-clocking could shape the packet interarrivals of a TCP connection in a two-level ON-OFF pattern. The pattern causes burstiness in time scales up to the round-trip time of the TCP connection.

In our paper we analyzed the same traffic traces as in [16] and [35], and applied novel methods to characterize them. The network traffic traces are considered as a time series of the arrival times of the packets. Due to space limitation the analysis of the packet lengths is omitted. The arrival times form a monotone increasing series. The interarrival times are independent, identically distributed random variables. The classical modeling of the interarrival times goes back to Erlang, who successfully modeled the phone calls by a Poisson process with interarrival times distributed exponentially. We generalize his model by changing the distribution to a general family of infinitely divisible distributions and by the corresponding Lévy processes [29]. Since a subset of these distributions—called $\alpha-$stable distributions (asymmetrical in our case)— provides self-similar processes, we can analyze not just if the packet traces are self-similar, but we go beyond the results of previous papers and measure how close these packet traces are to being self-similar. The instrument of our analysis is the so called Truncated Lévy Flights [34]. The current paper is a continuation of our research project to investigate the tendency of the Internet's traffic burstiness. As part of the research project we have been comparing traffic traces captured from various years covering the last and current decade. The current paper relies on traces captured in 2003 and 2008 in terms of burstiness of the interarrival times. In another paper [36] we already analyzed the traces captured only in 2003 in terms of the burtiness of both the interarrival times and packet lengths. The current paper uses similar statistical methods as in [36] and investigates the traces captured in 2008 and compares it to the characteristics of the traces from 2003. In this paper we demonstrate that the 2003 traces are totally different from the 2008 traces in terms of burstiness and we conclude that based on the sample traces, the Internet is losing its self-similar nature that was so prevalent for years.

The second section describes the mathematical models applied for the analyses of the traces. The third section discusses the types of traces used in our work. The fourth section presents the results of the application of our model for the data, followed by the conclusion in section five.

## 2. Model: Smoothly Truncated Lévy Flights

In this section we introduce a model: The Smoothly Truncated Levy Flights (STLFs). It will be applied in section IV for describing the distribution of the interarrival times of the packet traces. The time series of the interarrival times under consideration is the sequence of the differences between consecutive arrivals of packets collected in the Internet backbone. The data collection details are described in section III.

The Truncated Lévy Flights were introduced by Mantegna and Stanley [23] as models for random phenomena, which exhibit properties at small time-scales similar to those of self-similar Lévy processes. The Truncated Lévy Flights have distributions with cutoffs at large time-scales, i.e., they have finite moments of any order. Building on Mantegna and Stanley's ideas Koponen [19] defined the Smoothly Truncated Lévy Flights (STLFs), which had the advantage of a nice analytic form. Independently, the same family of distributions was described earlier by Hougaard [13] in the context of a biological application. The concept of the more general distribution, called tempered stable distribution, is due to Rosiński [27] (see, e.g., [34] and [33] for a partial history of these works).

Since the interarrival times are positive, we consider STLF with a totally asymmetric distribution. It is given by the cumulant function (log of the characteristic function)

$$\psi_X(u) = a\Gamma(-\alpha)\left[(\lambda - iu)^\alpha - \lambda^\alpha\right], \qquad (1)$$

where $\alpha \in (0,1)$ and $\lambda, a > 0$. A more general discussion of STLF is given in Appendix C. This distribution depends on three parameters: the *index* $\alpha$, the *truncation* parameter $\lambda$, and the *scale* parameter $a$. These parameters provide some information about the position of the distribution in the following manner:

**Property 1.** If $\alpha$ and $a$ are fixed and $\lambda$ tends to zero, then the limit distribution is a totally asymmetric $\alpha$-stable distribution and the corresponding Lévy process is self-similar.

**Property 2.** If $\lambda$ and $a$ are fixed and $\alpha$ tends to zero, then the limit distribution is Gamma with parameters $(a, \lambda)$. In particular, if $a$ is 1, then the limit is exponential, therefore the Lévy process is Poisson.

**Property 3**. If $\lambda$ and $\alpha$ are fixed, then for small $a$ the distribution is close to the $\alpha$- stable distribution and for large $a$ the distribution is close to Gaussian. More precisely, moments of any positive order $\varrho$ (including fractional) have the following asymptotics:

$$\log \mathbf{E}(|X|^\varrho) \sim \begin{cases} \min(\varrho/\alpha, 1)\log a + c_1, & \text{as} \quad a \to 0; \\ \varrho \log a + c_2, & \text{as} \quad a \to \infty. \end{cases}$$

For $m \geq 1$, the cumulants, derived from the cumulant function (1), are given in terms of the parameters $\alpha$, $\lambda$, and $a$, namely,

$$\mathrm{cum}_m(X) = a\lambda^{\alpha-m}\Gamma(m-\alpha). \qquad (2)$$

# 3. Traffic traces

The traffic traces were captured from OC-48 (2.5 Gbps) connections of the Internet backbone collected by CAIDA [38][1] (The Cooperative Association for Internet Data Analysis). CAIDA's OC-48 traffic gathering devices compile packet headers at large peering points of several large Internet Service Providers (ISPs) in the United States. We used the traces collected in both directions of an OC-48 link at AMES Internet Exchange (AIX) on three different times. The traces have been divided into a collection of 5-minute files and another collection of 60-minute files to allow downloading the traces easier. These packet traces include the packet headers of packets with IP addresses anonymized with the prefix-preserving Crypto-PAn library. These traces include only IPv4 traffic. The precision of the traces is in the order of microseconds. Table 1 includes the details of the traces.

[1]Support for CAIDA's OC48 Traces is provided by the National Science Foundation, the US Department of Homeland Security, DARPA, Digital Envoy, and CAIDA Members.

## 3.1. OC-192 traces

We also analyzed packet traces collected for four hours by CAIDA in May, 2008. The data sets contained anonymized traffic traces from an Internet data collection monitor on an OC-192 Internet backbone link (9953.28 Mbps). The Internet data collection monitor was located in Chicago, IL, and was connected to a Tier1 ISP between Chicago, IL and Seattle, WA. The traffic was captured by two network monitoring cards in both directions. A single card was connected to a single direction of the full-duplex backbone link. The directions were denoted by A (Seattle to Chicago) and B (Chicago to Seattle). The anonymized trace data contains layer 3 and layer 4 protocols: IP for layer 3, and TCP, UDP or ICMP for layer 4. These packets are originally encapsulated in layer 1 and layer 3 protocols. On the physical layer the protocol is PoS (Packets over SONET), on the data link layer the protocols are cHDLC (Cisco's version of HDLC), or PPPoHDLC (PPP over HDLC). Between layer 2 and 3 the service provider also inserts one or more MPLS headers [4].

The packets were captured more then an hour resulting in two traces direction A and B (Compressed size of direction A is 4.1 GB, compressed size of the trace in direction B is 14 GB).

The traces were captured by dedicated network measurement cards built by Endace Measurement Systems especially designed to provide very high quality packet time-stamps. Since GPS transmitters broadcast the current time based on atomic clocks, all Endace network measurement cards are equipped with ports allowing a GPS receiver to be connected providing clock synchronization.

Endace's DAG Universal Clock Kit (DUCK) provides per packet time-stamps that are both high resolution and capable of accurate synchronization to the Coordinated Universal Time (UTC). When a packet is captured, the DUCK time-stamps the beginning of the packet arrival in hardware unlike in NIC-based packet capture. NIC based time-stamping occurs in the host computer sometime after the packet has arrived estimating a time-stamp for the end of the packet arrival. The DUCK represents time in a single 64-bit fixed point number representing seconds since midnight on the first of January 1970. The high 32-bits store the integer number of seconds, the lower 32-bits contain the binary fraction of the second. This method provides a resolution of 232 seconds, or approximately 233 picoseconds.

Since the card's output file format was not supported by the majority of traffic analysis tools, CAIDA converted the original traces to a format with nanosecond

| Date | Duration | Length of the trace in bytes |
|---|---|---|
| Aug 14, 2002 | 3 hours, with 1 hour gap in one direction | 108GB |
| Jan 15, 2003 | 1 hour, in both directions | 30GB |
| Apr 24, 2003 | 1 hour, in both directions | 13GB |

**Table 1. Details of the traces.**

timestamp precision along with the packet lengths for both IPv4 and IPv6 packets separately. It is noticeable from the size of the traces that direction A had less traffic then direction B. A possible reason of the difference is that many content servers were located at one end of the link. Another interesting observation of the traffic was that based on a smaller sample, only a small portion (~8.6%) of IPv4 addresses was captured as both source and destination IP addresses in packets after merging both directions. This could be the indication that the network traffic in this area of the backbone may have been routed asymmetrically (Email communications with Emile Aben, Data Administrator, CAIDA/SDSC/UCSD).

# 4. Packet interarrival times

The series of interarrival times in the OC-48 traces are modeled as stochastic series. If the series correspond to a Poisson process, then the interarrival times have exponential distribution. In Figure 1 we fitted the Gamma distribution to the interarrival times of the OC-48 traces captured on April, 24, 2003 (20030424-001000-0-anon.pcap). (The Gamma distribution is more general than the exponential distribution. It reduces to the exponential distribution, if the shape parameter is 1.)

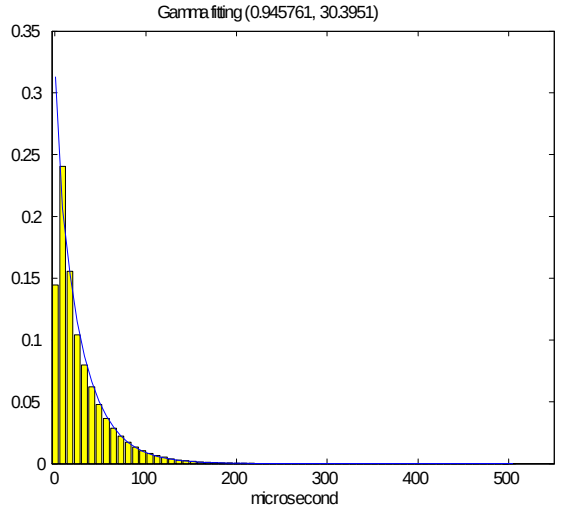



Fig. 1. **Gamma distribution of the interarrival times.**

Although the estimated parameters (0.945761, 30.3951) suggest that the distribution of the interarrival times is close to the exponential distribution, the Kolmogorov-Smirnov test strongly rejects the hypothesis that the series follows the Gamma distribution. Consequently, the corresponding process cannot be a Poisson process. Therefore we reject the hypothesis that the packet trace follows a Poisson process.

We continue the search for a distribution that would be suitable for characterizing the interarrival times by applying the family of Lévy processes.

The Poisson process is one of the simplest Lévy processes (see, e.g., [29]) with the main assumption that the increments—the interarrival times— are independent, homogeneous and exponential. Changing the distribution of the increments we obtain a wide variety of Lévy-stable processes as candidates for modeling the interarrival times [37]. Lévy-stable processes show heavy tail behavior making it impossible to apply them for the measured interarrival times: Figure 1 depicts that there are very few measurements after 200 micro second. The heavy tail of a distribution also implies that the moments do not exist, so these distributions are not appropriate for modeling purposes. Other members of the family of Lévy processes, the Smoothly Truncated Lévy Flights (STLF), have higher order moments. Since they have been successfully applied for finance, biological, and physical phenomena it is reasonable to apply it for traffic analysis as well. Some applications of the STLF are demonstrated in [23], [19], [13], and [34]. The following formula of the cumulants of STLF provides a means for estimating the parameters by the method of moments, i.e., calculating the empirical values from the traffic traces and compare them with the theoretical values above:

$$\mathrm{cum}_m(X) = a\lambda^{\alpha-m}\Gamma(m-\alpha),$$

More precisely, for a given trace we calculate the estimated cumulants $\widehat{\mathrm{cum}_m}$, $m = 1, 2, \ldots 8$, then we use the least squares method for finding the estimates $\widehat{a}$, $\widehat{\lambda}$, and $\widehat{\alpha}$ (for the details, please see the authors).

## 4.1 Analysis of OC-48 traces

We carried out these calculations for the OC48 trace captured on April 24, 2003 (20030424-002500-0-anon.pcap). Figure 2 shows the log of estimated cumulants $\widehat{\mathrm{cum}_m}$, and the log of cumulants $\widetilde{\mathrm{cum}_m}$, $m = 1, 2, \ldots 8$, of the Smoothly Truncated Lévy Flights when the parameters are estimated, i.e.,

$$\widetilde{\mathrm{cum}_m}(X) = \widehat{a}\widehat{\lambda}^{\widehat{\alpha}-m}\Gamma(m-\widehat{\alpha}),$$

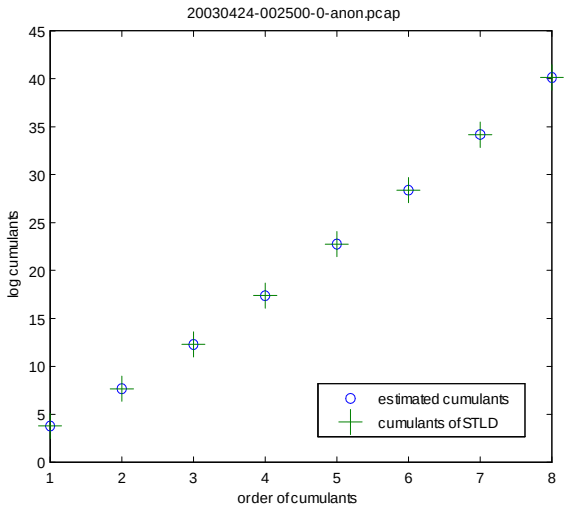



Fig. 2. **Comparison of the cumulants and estimated cumulants of the OC48 trace.**

| $Trace$ $(Direction\,'0')$ | $\alpha$ | $\lambda$ | $a$ |
|---|---|---|---|
| 00_0 | 0.13106 | 0.02010 | 1.21465 |
| 05_0 | 0.17247 | 0.01871 | 1.37994 |
| 10_0 | 0.2162 | 0.0177 | 1.5266 |
| 15_0 | 0.13525 | 0.01828 | 1.23914 |
| 20_0 | 0.15424 | 0.01771 | 1.29506 |
| 25_0 | 0.15570 | 0.01768 | 1.30567 |
| 30_0 | 0.19040 | 0.01773 | 1.42341 |
| 35_0 | 0.22436 | 0.01802 | 1.56060 |
| 40_0 | 0.25456 | 0.01717 | 1.68368 |
| 45_0 | 0.19989 | 0.01760 | 1.44594 |
| 50_0 | 0.11637 | 0.01699 | 1.14885 |
| 55_0 | 0.14671 | 0.01818 | 1.26795 |

**Table 2. The estimated parameters of the OC-48 5 minute traces in direction '0'.**

where $\widehat{\alpha} = 0.15570$, $\widehat{\lambda} = 0.01768$, $\widehat{a} = 1.30567$. Since the fitting is good, it implies that this trace is close to the self-similar process because the value of $\lambda$ is very small. At the same time the trace is not too far from the exponential distribution considering that the value of $\alpha$ is small and $a$ is close to 1.

Table 2 and 3 show the estimated parameters of the OC-48 five minute traces.

In general, we can conclude that the distribution of these traces are close to $\alpha$ - stable distribution, since the estimations of $\lambda$ are very small, hence the process is close to a self-similar process (see Property 1 in section II. A). It is also clear from the parameters that the traces in direction '1' are closer to the exponential distribution (see Property 2 in section II. A) than the ones in direction '0', since the parameter $\alpha$ is small and $a$ is close to 1 at least in these traces: 05_1, 10_1, 25_1, 45_1, and 50_1. Therefore, the traces in direction '1' are closer to a Poisson process then the traces in direction '0'. The reason for the different characteristics of the traffic traces in directions '0' and '1' is under investigation.

| $Trace$ $(Direction\,'1')$ | $\alpha$ | $\lambda$ | $a$ |
|---|---|---|---|
| 00_1 | 0.19944 | 0.02666 | 1.31111 |
| 05_1 | 0.06867 | 0.03078 | 0.99380 |
| 10_1 | 0.08212 | 0.02729 | 0.99712 |
| 15_1 | 0.17804 | 0.02431 | 1.29634 |
| 20_1 | 0.17372 | 0.02390 | 1.28337 |
| 25_1 | 0.07781 | 0.02525 | 0.98236 |
| 30_1 | 0.16226 | 0.02449 | 1.20388 |
| 35_1 | 0.11714 | 0.02452 | 1.07384 |
| 40_1 | 0.20719 | 0.02221 | 1.35552 |
| 45_1 | 0.07819 | 0.02307 | 1.00036 |
| 50_1 | 0.08903 | 0.02259 | 1.01705 |
| 55_1 | 0.16310 | 0.02214 | 1.21355 |

**Table 3. The estimated parameters of the OC-48 5 minute traces in direction '1'.**

## 4.2 Analysis of OC-192 traces

Since the OC-192 datasets are significantly larger that the OC-48 datasets, we consider the parameters of the model being time dependent. We analyze the behavior of the model at every 0.1 second. The parameters of the model are estimated for the duration of 1 second interval. We assume that the traffic traces are locally stationary. The following figures depict the characteristics of the traffic flow in direction B and A. (We use the notations direction A and direction B to clearly distinguish the results related to the traces captured in 2003 and in 2008.) Figure 3 clearly demonstrates that $\alpha(t)$ is not close to zero, therefore the traffic flow in direction B does not exhibit the attributes of a self-similar or a Poisson process (see Property 1 and 2).

We obtained a similar figure for $\lambda(t)$ and $a(t)$ as well, see Figure 4-5.

In Direction A $\alpha$ is equal to zero in 43% of the samples, while $a$'s values are close to 1. Figure 6 shows these values of $a$. The figure demonstrates that the traffic trace is approaching the Poisson process in 43% of the total samples.

# 5. Conclusion

We presented a novel model for analyzing the self-similarity of Internet traffic captured by CAIDA. The network traffic traces were considered as time series of the arrival times of the packets. We characterized the traffic traces with three parameters of Lévy Flights and placed a particular trace somewhere in the space generated by the Poisson and self-similar Lévy processes. Previous papers characterized the same traces as either self-similar or not self-similar traces. We were able to measure how close these packet traces were to being self-similar. We analyzed two sets of traces; one captured

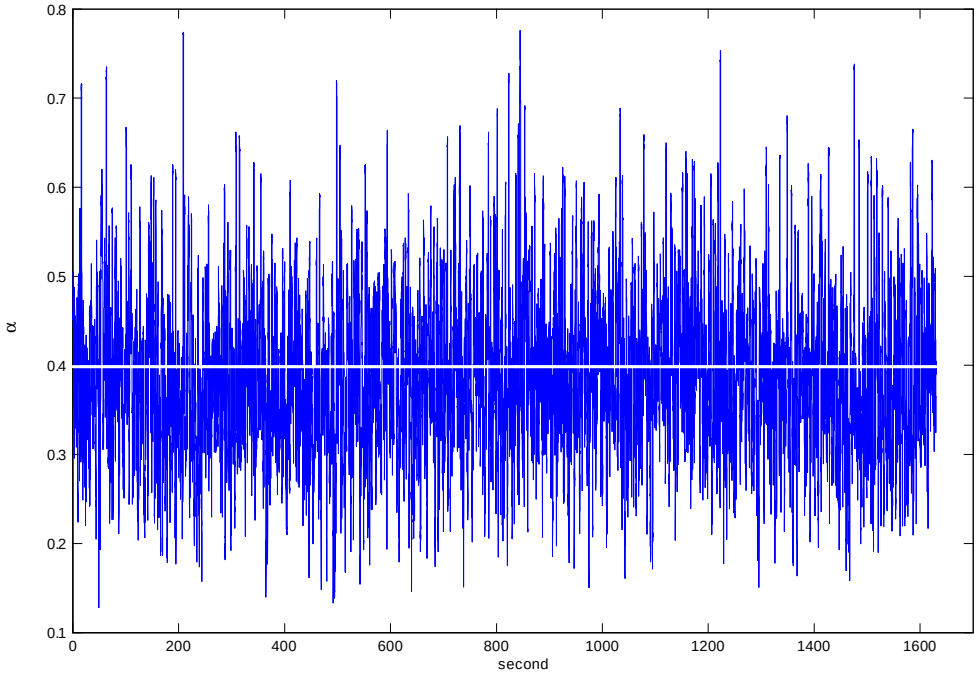



**Fig. 3.** $\alpha(t)$ **in Direction B.**

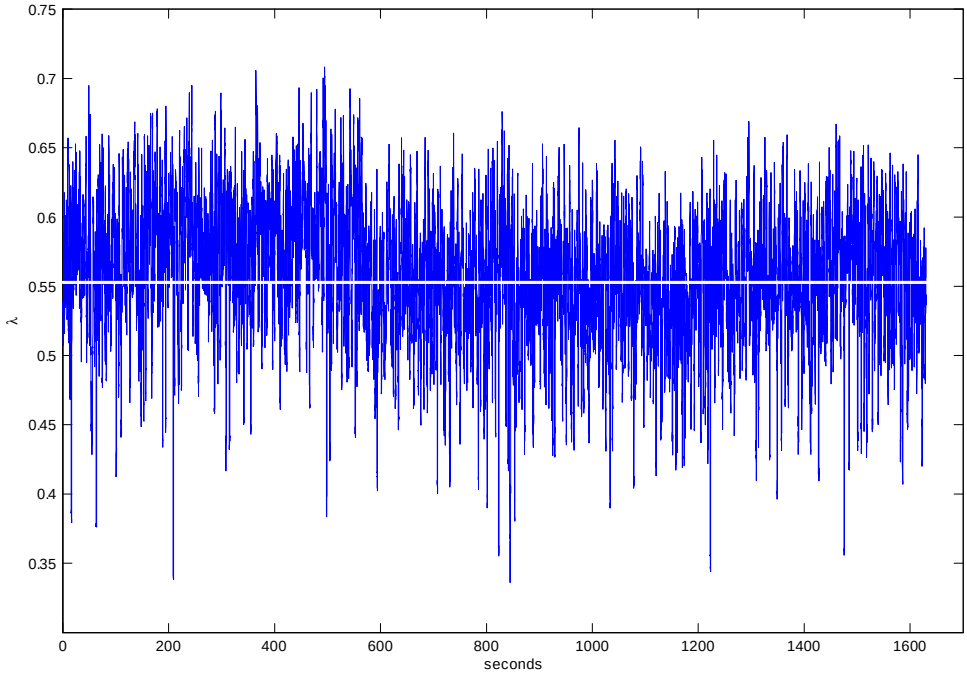



**Fig. 4.** $\lambda(t)$ **in Direction B.**

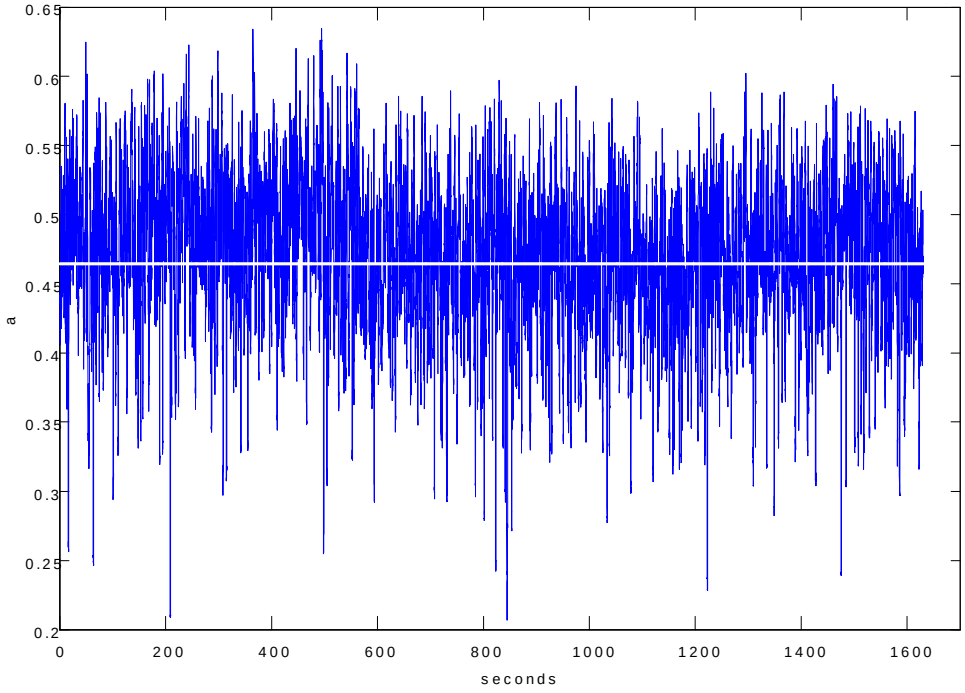



**Fig. 5.** $a\,(t)$ **in Direction B.**

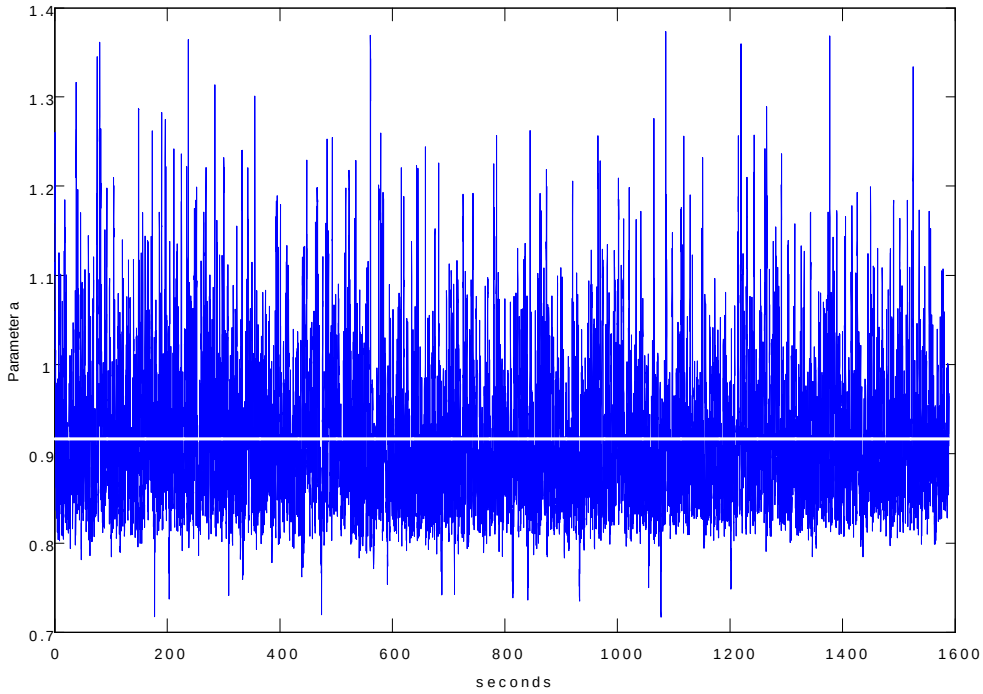



**Fig. 6. Values of** $a$ **when** $\alpha\,(t) = 0$**, mean**$(a) = 0.91639$

from an OC-48 link in 2003 and another from an OC-192 trace in 2008. We concluded that the distribution of the 2003 traces was close to $\alpha-$stable distribution, since the estimations of $\lambda$ were very small; hence the process was close to a self-similar process. It was also clear from the parameters that the traces in direction '1' were closer to the exponential distribution than the ones in direction '0'. Therefore, the traces in direction '1' were closer to a Poisson process then the traces in direction '0'. Regarding the traces from 2008 we concluded that in Direction B the traffic flow can be modeled by the Lévy Flights, but in Direction A, a large portion of the trace shows evidence of the properties of the traditional Poisson process.

# 6. Appendix

## 6.1. Cumulants

It is well known that there is a one to one correspondence between the moments and cumulants. The expected value is the cumulant of first order:

$$\mathrm{cum}_1(X) = \mathsf{E}X.$$

The cumulants of order 2 and 3 are equal to the central moments

$$\begin{aligned} \mathrm{cum}_2(X) &= \mathrm{Cov}(X,X) \\ &= \mathsf{E}\left(X-\mathsf{E}X\right)^2, \\ \mathrm{cum}_3(X) &= \mathsf{E}\left(X-\mathsf{E}X\right)^3, \end{aligned} \tag{3}$$

but this is not true for higher order cumulants. One might easily check this for the case of cumulants of order four. Let us denote the central moment of $k^{th}$ order by $m_k = \mathsf{E}\left(X-\mathsf{E}X\right)^k$, then we have

$$\begin{aligned} \mathrm{cum}_4(X) &= m_4 - 3m_2^2, \\ \mathrm{cum}_5(X) &= m_5 - 10m_3m_2, \\ \mathrm{cum}_6(X) &= m_6 - 15m_4m_2 - 10m_3^2 + 30m_2^3, \\ \mathrm{cum}_7(X) &= m_7 - 21m_5m_2 - 35m_4m_3 + 210m_3m_2^2, \\ \mathrm{cum}_8(X) &= m_8 - 28m_6m_2 - 56m_5m_3 - 35m_4^2 \\ &\quad +420m_4m_2^2 + 560m_3^2m_2 - 630m_2^4, \end{aligned} \tag{4}$$

see [18] p.64, [32] p.10. If a sample $x_1, x_2, \ldots x_n$ is given, then the estimated expected value, i.e., first order cumulant is the mean $\overline{x}$, and the estimated $k^{th}$ order central moment

$$\begin{aligned} \widehat{m}_k &= \overline{\left(x-\overline{x}\right)^k} \\ &= \frac{1}{n}\sum_{j=1}^{n}\left(x_j-\overline{x}\right)^k. \end{aligned}$$

Now, the estimated cumulants are given in terms of estimated central moments (see formulae (4) above). For example, the $4^{th}$ order estimated cumulant $\widehat{\mathrm{cum}}_4$ is calculated by

$$\widehat{\mathrm{cum}}_4(X) = \widehat{m}_4 - 3\widehat{m}_2^2.$$

## 6.2. STLF

Let us recall that the STLF $X(t)$ is a Lévy process, i.e., a process with homogeneous and independent increments and $X(0)=0$. The probability distribution of $X = X(1)$ has characteristic function of the form

$$\varphi_X(u) = \exp\left(\psi_X(u)\right),$$

where the *cumulant function*

$$\psi_X(u) = a\lambda^\alpha\left[p\zeta_\alpha\left(-u/\lambda\right) + q\zeta_\alpha\left(u/\lambda\right)\right] + iub,$$

and $\lambda > 0,\ a, p, q \geq 0,\ p+q=1,\ b$ is a real number, and

$$\zeta_\alpha(r) = \begin{cases} \Gamma(-\alpha)\left[(1-ir)^\alpha - 1\right], & \text{for} \quad 0<\alpha<1; \\ (1-ir)\log(1-ir) + ir, & \text{for} \quad \alpha = 1; \\ \Gamma(-\alpha)\left[(1-ir)^\alpha - 1 + i\alpha r\right], & \text{for} \quad 1<\alpha<2. \end{cases}$$

(See [34] for details.)

Without loss of generality, we only consider the case when the shift parameter $b=0$. Parameters $p$ and $q$ describe the *skewness* of the probability distributions, and $p=q=1/2$ yields a symmetric distribution. Parameter $\lambda$ will be referred to as the *truncation* parameter.

In the case of $0<\alpha<1$, the cumulant function is given by the formula

$$\psi_X(u) = a\lambda^\alpha\Gamma(-\alpha)\left[p\left(1+i\frac{u}{\lambda}\right)^\alpha + q\left(1-i\frac{u}{\lambda}\right)^\alpha - 1\right], \tag{5}$$

and if $p=0$, the cumulant function

$$\begin{aligned} \psi_X(u) &= a\lambda^\alpha\Gamma(-\alpha)\left[\left(1-i\frac{u}{\lambda}\right)^\alpha - 1\right] \\ &= a\Gamma(-\alpha)\left[(\lambda-iu)^\alpha - \lambda^\alpha\right], \end{aligned} \tag{6}$$

describes a distribution totally concentrated on the positive half-line. The distribution of $X$ will be denoted by $STLF_\alpha(a,p,\lambda)$. The index $\alpha$ corresponds to the nontruncated limit when $\lambda=0$. In this case the distribution of $X$ is the classical Lévy's $\alpha-$stable probability distribution. The scale parameter $a$ tunes the time unit to $a$, hence the distribution of $X(t)$ is $STLF_\alpha(at,p,\lambda)$.

The role of the truncation parameter $\lambda$ is obvious in the following particular case. For the one-sided $STLF_\alpha(a,0,\lambda)$ distribution with $0<\alpha<1$, the cumulant function has the form

$$\psi_X(u) = a\lambda^\alpha\Gamma(-\alpha)\left[\left(1-i\frac{u}{\lambda}\right)^\alpha - 1\right]. \tag{7}$$

As $\lambda \to 0$, the distribution $STLF_\alpha(a,0,\lambda)$ converges to the $\alpha-$stable distribution $STLF_\alpha(a,0,0)$. The parameter $\lambda$ looks appropriate for measuring the distance

from the $\alpha-$stable distribution, but it can be noticed that scaling $X$ will change the value of $\lambda$ as well. More precisely, if $X$ distributed as $STLF_\alpha(a,0,\lambda)$ then the distribution of $cX$ is $STLF_\alpha(ac^\alpha,0,\lambda/c)$, where $c>0$. Therefore the distance from the $\alpha-$stable distribution can be measured by the parameter $\lambda$ when the value $a$ is fixed to 1.

For a fixed $\lambda, a > 0$, as $\alpha \to 0$, the distribution $STLF_\alpha$ tends to the Gamma distribution $\Gamma(a,\lambda)$. Indeed, for $0 < \alpha < 1$, the Laplace transform $\phi_\lambda$ of $STLF_\alpha(a,0,\lambda)$ is

$$\phi_\lambda(u) = \exp\left(a\lambda^\alpha \Gamma(-\alpha)\left[(1+u/\lambda)^\alpha - 1\right]\right),$$

and

$$\lim_{\alpha\to 0} \exp\left(-a\Gamma(1-\alpha)\frac{(\lambda+u)^\alpha - \lambda^\alpha}{\alpha}\right) = \exp\left(-a\log(1+u/\lambda)\right) = (1+u/\lambda)^{-a},$$

by the L'Hospital rule.

### 6.3. Estimating the parameters of $STLF_\alpha(a,0,\lambda)$

Take the logarithm of

$$\mathrm{cum}_m(X) = a\lambda^{\alpha-m}\Gamma(m-\alpha).$$

We obtain

$$\log \mathrm{cum}_m(X) = \log a + (\alpha-m)\log\lambda + \log\Gamma(m-\alpha). \quad (8)$$

Plug the estimated cumulants $\widehat{\mathrm{cum}_m}$ (see (4) above) into the left side of equation (8), then we have three unknowns $a$, $\lambda$, and $\alpha$. In order to find the parameter values for the best fitting start with the system of equations when $m = 2,3,4$, i.e.,

$$\begin{aligned}
\log\widehat{\mathrm{cum}_2}(X) &= \log a + (\alpha-2)\log\lambda + \log\Gamma(2-\alpha), & (9)\\
\log\widehat{\mathrm{cum}_3}(X) &= \log a + (\alpha-3)\log\lambda + \log\Gamma(3-\alpha) & (10)\\
&= \log a + (\alpha-3)\log\lambda + \log(2-\alpha) + \log\Gamma(2-\alpha),\\
\log\widehat{\mathrm{cum}_4}(X) &= \log a + (\alpha-4)\log\lambda + \log\Gamma(4-\alpha) & (11)\\
&= \log a + (\alpha-4)\log\lambda + \log(3-\alpha) + \log(2-\alpha) + \log\Gamma(2-\alpha).
\end{aligned}$$

The difference of the first two equations (9-10) gives

$$\log\widehat{\mathrm{cum}_3}(X) - \log\widehat{\mathrm{cum}_2}(X) = -\log\lambda + \log(2-\alpha) = \log\frac{2-\alpha}{\lambda},$$

hence

$$\alpha = 2 - \lambda\frac{\widehat{\mathrm{cum}_3}(X)}{\widehat{\mathrm{cum}_2}(X)}.$$

Similarly from the last two equations (10-11)

$$\alpha = 3 - \lambda\frac{\widehat{\mathrm{cum}_4}(X)}{\widehat{\mathrm{cum}_3}(X)},$$

therefore we obtain

$$\begin{aligned}
\widehat{\lambda} &= \frac{\widehat{\mathrm{cum}_3}(X)\,\widehat{\mathrm{cum}_2}(X)}{\widehat{\mathrm{cum}_4}(X)\,\widehat{\mathrm{cum}_2}(X) - \left[\widehat{\mathrm{cum}_3}(X)\right]^2},\\
\widehat{\alpha} &= 2 - \frac{\left[\widehat{\mathrm{cum}_3}(X)\right]^2}{\widehat{\mathrm{cum}_4}(X)\,\widehat{\mathrm{cum}_2}(X) - \left[\widehat{\mathrm{cum}_3}(X)\right]^2},\\
\widehat{a} &= \frac{\widehat{\mathrm{cum}_2}(X)}{\widehat{\lambda}^{\widehat{\alpha}-2}\Gamma(2-\widehat{\alpha})}.
\end{aligned}$$

We obtain more precise estimations for the parameters, if we use these estimates as initial values and refine the estimates using nonlinear least squares, which minimizes

$$\sum_{m=1}^{8}\left[\mathrm{cum}_m(X) - a\lambda^{\alpha-m}\Gamma(m-\alpha)\right]^2.$$

### 6.4. Gamma distribution

The Gamma pdf is

$$f(x\,|a,b) = \frac{x^{a-1}}{b^a\Gamma(a)}\exp(-x/b),\ x>0,$$

where $a$ and $b$ are positive and called as shape and scale parameter respectively. If $a = 1$, then it reduces to the exponential distribution.

**ACKNOWLEDGEMENT**

The authors would like to thank the referees for their constructive comments and suggestions to improve this paper.

REFERENCES

[1] M. E. Crovella and A. Bestavros. Self-similarity in world wide web traffic: Evidence and possible causes. *IEEE/ACM Transactions on Networking*, 5(6), 1997.

[2] H. Doi, T. Matsuda, M. Yamamoto, Influences of TCP congestion control mechanisms to multi-fractal nature of generated traffic, in: Proc. GLOBECOM 2003, 2003, pp. 3658–3662.

[3] A. Erramilli, O. Narayan, and W. Willinger. Experimental queueing analysis with long-range dependent packet traffic. *IEEE/ACM Trans. Networking*, 4(2):209–223, 1996.

[4] Stephen Donnelly, PhD Endance Technology Ltd Endance DAG Time-Sampling Whitepaper, http://www.endace.com/assets/files/timestamping_whitepaper.pdf, 2007.

[5] A.Feldmann,A.C.Gilbert,and W.Willinger, Data Networks as Cascades: Investigating the Multifractal Nature of the Internet WAN Traffic. *IEEE/ACM Trans. Networking* In Proceedings of ACM SIGCOMM, 1998.

[6] A.Feldmann, A.C.Gilbert, P. Huang, and W.Willinger, Dynamics of IP Traffic: A Study of the Role of Variability and The Impact of Control. *IEEE/ACM Trans. Networking*In Proceedings of ACM SIGCOMM, 1999.

[7] J. Gao, I. Rubin, Multiplicative multifractal modeling of long-range dependent network traffic,Int. J. Commun. Syst. 14 (2001) 783–801.

[8] W.-B. Gong, Y. Liu, V. Misra, and D. Towsley. Self-similarity and long range dependence on the internet: A second look at the evidence, origins and implications. *Computer Networks*, 48(Issue 3):377–399, June 2005.

[9] T. Gyires. Simulation of the harmful consequences of self-similar network traffic. *The Journal of Computer Information Systems*, Summer issue:94–111, 2002.

[10] G. He, Y. Gao, J. C. Hou, and K. Park. A case for exploiting self-similarity of network traffic in TCP congestion control. *Computer Networks*, 45(Issue 6):743–766, August 2004.

[11] G. He and J. C. Hou. On sampling self-similar internet traffic. *Computer Networks*, 50(Issue 16):2919–2936, November 2006.

[12] N.Hohn, D.Veitch,and P. Abry. Does Fractal Scaling at the IP Level Depend on TCP Flow Arrival, Processes? *Computer Networks* In Proceedings of Internet MeasurementWorkshop (IMW), Nov.2002.

[13] P. Hougaard. Survival models for heterogeneous populations derived from stable distributions. *Biometrika*, 73(2):387–396, 1986.

[14] H. Jiang and C. Dovrolis. Source-Level Packet Bursts: Causes and Effects. *ESAIM Probab. Stat.* In Proceedings of Internet Measurement Conference (IMC), Oct.2003.

[15] H. Jiang and C. Dovrolis. Why is the Internet Traffic Bursty in Short Time Scales. *ESAIM Probab. Stat.* In Proceedings of the ACM SIGMETRICS Performance Evaluation Review Conference, 241–252.

[16] T. Karagiannis, M. Molle, M. Faloutsos, and A. Broido. A nonstationary poisson view of internet traffic. In *Proceedings of INFOCOM 2004*, pages 1558– 1569, vol.3, March 2004.

[17] S. Karlin and H. M. Taylor. *A first course in stochastic processes*. Academic Press [A subsidiary of Harcourt Brace Jovanovich, Publishers], New York-London, second edition, 1975.

[18] M. G. Kendall. *The Advanced Theory of Statistics*. Vol. I. J. B. Lippincott Co., Philadelphia, 1944.

[19] I. Koponen. Analytic approach to the problem of convergence of truncated Lévy flights towards the Gaussian stochastic process. *Phys. Rev. E.*, 52:1197–1199, 1995.

[20] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson. On the self-similar nature of Ethernet traffic (extended version). *IEEE/AC Transactions on networking*, 2(1):1–15, 1994.

[21] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson. Statistical analysis and stochastic modeling of self-similar data traffic. In J. Labetoulle and J. W. Roberts, editors, *The Fundamental Role of Teletraffic in the Evolution of Telecommunications Networks, Proceedings of the 14th International Teletraffic Congress (ITC '94)*, pages 319–328. Elsevier Science B.V., Amsterdam, 1994.

[22] W. E. Leland and D. W. Wilson. High time-resolution measurement and analysis of LAN traffic: Implications for LAN interconnection. In *Proceedings of the IEEE NFOCOM'91*, pages 1360–1366, Bal Harbour, FL, 1991.

[23] R. N. Mantegna and H. E. Stanley. Stochastic processes with ultraslow convergence to a Gaussian: The truncated Lévy flight. *Phys. Rev. Lett.*, 73:2946–2949, 1994.

[24] M. Masugi, T. Takuma. Multi-fractal analysis of IP-network traffic for assessing time variations in scaling properties. Physica D 225 (2007) 119–126, 2006.

[25] K. Park, M. Kim and G. Crovella. On the relationship between file sizes, transport protocols, and self-similar network traffic. In *Proceedings of the 4th Int. Conf. Network Protocols (ICNP'96)*, pages 171–180, 1996.

[26] V. Paxson and S. Floyd. Wide-area traffic: The failure of Poisson modeling. *IEEE/ACM Transactions on Networking*, 3(3):226–244, Jume 1995.

[27] J. Rosinski. Tempering stable processes. Technical report, Department of Mathematics of the Univ. of Tennessee in Knoxville, Tennessee, 2004. In press: Stochastic Processes and their Applications, (2006), doi:10.1016/j.spa.2006.10.003.

[28] R. R. S. Sarvotham and R. Baraniuk. Connection-level Analysis and Modeling of Network Traffic. *Phys. Rev. Lett.*In Proceedings of Internet Measurement Workshop (IMW), Nov. 2001.

[29] K. Sato. *Lévy processes and infinitely divisible distributions*, volume 68 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, Cambridge, 1999. Translated from the 1990 Japanese original, Revised by the author.

[30] K. Sriram and W. Whitt. Characterizing superposition arrival processes in packet multiplexors for voice and data. *IEEE J. Select. Areas Commun.*, 4:833–846, 1986.

[31] M. S. Taqqu, V. Teverovsky, and W. Willinger. Is network traffic self-similar or multifractal? *Fractals*, 5(1):63–73, 1997.

[32] Gy. Terdik. *Bilinear Stochastic Models and Related Problems of Nonlinear Time Series Analysis; A Frequency Domain Approach*, volume 142 of *Lecture Notes in Statistics*. Springer Verlag, New York, 1999.

[33] Gy. Terdik and W. A. Woyczynski. Rosiñski measures for tempered stable and related ornstein-uhlenbeck processes. *Probability and Mathematical Statistics (PMS)*, Urbanik Volume:x–xx, 2006.

[34] Gy. Terdik, W. A. Woyczynski, and A. Piryatinska. Fractional- and integer-order moments, and multiscaling for smoothly truncated Lévy flights. *Physics Letters A*, 348:94–109, 2006.

[35] Gy. Terdik and T. Gyires. Internet Traffic Modeling with Lévy Flights, in the Proceedings of the Seventh International Conference on Networking, IARIA, April 13-18, 2008, Cancun, Mexico.

[36] Gy. Terdik and T. Gyires. Lévy Flights and Fractal Modeling of Internet Traffic. IEEE/ACM Transactions on Networking 17(1): 120-129, 2009.

[37] G. Xiaohu, Z. Guangxi, and Z. Yaoting. On the testing for alpha-stable distributions of network traffic. *Computer Communications*, 27(Issue 5):447–457, March 2004.

[38] Colleen Shannon, Emile Aben, kc claffy, Dan Andersen, Nevil Brownlee, The CAIDA OC48 Traces Dataset, http://www.caida.org/data/passive.

[39] http://www.tcpdump.org/, http://ita.ee.lbl.gov/html/contrib/tcpdpriv.html, http://ita.ee.lbl.gov/html/contrib/sanitize.html.

[40] http://ita.ee.lbl.gov/html/contrib/LBL-PKT.html.

[41] http://www.wide.ad.jp/project/wg/mawi.html, Samplepoint-B.


www.iariajournals.org

**International Journal On Advances in Intelligent Systems**
ICAS, ACHI, ICCGI, UBICOMM, ADVCOMP, CENTRIC, GEOProcessing, SEMAPRO, BIOSYSCOM, BIOINFO, BIOTECHNO, FUTURE COMPUTING, SERVICE COMPUTATION, COGNITIVE, ADAPTIVE, CONTENT, PATTERNS
issn: 1942-2679

**International Journal On Advances in Internet Technology**
ICDS, ICIW, CTRQ, UBICOMM, ICSNC, AFIN, INTERNET, AP2PS, EMERGING
issn: 1942-2652

**International Journal On Advances in Life Sciences**
eTELEMED, eKNOW, eL&mL, BIODIV, BIOENVIRONMENT, BIOGREEN, BIOSYSCOM, BIOINFO, BIOTECHNO
issn: 1942-2660

**International Journal On Advances in Networks and Services**
ICN, ICNS, ICIW, ICWMC, SENSORCOMM, MESH, CENTRIC, MMEDIA, SERVICE COMPUTATION
issn: 1942-2644

**International Journal On Advances in Security**
ICQNM, SECURWARE, MESH, DEPEND, INTERNET, CYBERLAWS
issn: 1942-2636

**International Journal On Advances in Software**
ICSEA, ICCGI, ADVCOMP, GEOProcessing, DBKDA, INTENSIVE, VALID, SIMUL, FUTURE COMPUTING, SERVICE COMPUTATION, COGNITIVE, ADAPTIVE, CONTENT, PATTERNS
issn: 1942-2628

**International Journal On Advances in Systems and Measurements**
ICQNM, ICONS, ICIMP, SENSORCOMM, CENICS, VALID, SIMUL
issn: 1942-261x

**International Journal On Advances in Telecommunications**
AICT, ICDT, ICWMC, ICSNC, CTRQ, SPACOMM, MMEDIA
issn: 1942-2601