Tomoya Enokido
Leonard Barolli
Makoto Takizawa (Eds.)

LNCS 4658

Network-Based Information Systems

First International Conference, NBiS 2007
Regensburg, Germany, September 2007
Proceedings

Springer

Tomoya Enokido Leonard Barolli
Makoto Takizawa (Eds.)

Network-Based Information Systems

First International Conference, NBiS 2007
Regensburg, Germany, September 3-7, 2007
Proceedings

Springer

Volume Editors

Tomoya Enokido
Rissho University
Faculty of Business Administration
4-2-16 Osaki, Shinagawa, Tokyo, 141-8602, Japan
E-mail: eno@ris.ac.jp

Leonard Barolli
Fukuoka Institute of Technology
Department of Information and Communication Engineering
3-30-1 Wajiro-Higashi, Higashi-ku, Fukuoka, 811-0214 Japan
E-mail: barolli@fit.ac.jp

Makoto Takizawa
Tokyo Denki University
Department of Computers and Systems Engineering
Ishizaka, Hatoyama, Saitama 350-0394, Japan
E-mail: makoto.takizawa@computer.org

Library of Congress Control Number: 2007933416

CR Subject Classification (1998): C.2.4, C.2.5, H.2.8, H.5.1, I.2.9, K.3.1, K.4.4

LNCS Sublibrary: SL 3 – Information Systems and Application, incl. Internet/Web and HCI

ISSN	0302-9743
ISBN-10	3-540-74572-6 Springer Berlin Heidelberg New York
ISBN-13	978-3-540-74572-3 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12115211 06/3180 5 4 3 2 1 0

Preface

Welcome to the proceedings of the 1st International Conference on Network-Based Information Systems (NBiS-2007), in conjunction with the 18th International Conference on Database and Expert Systems Applications DEXA-2007, which was held in Regensburg, Germany, September 3–4, 2007.

The main objective of NBiS-2007 was to bring together scientists, engineers, and researchers from both network systems and information systems with the aim of encouraging the exchange of ideas, opinions, and experience between these two communities.

NBiS started as a workshop and for 9 years it was held together with DEXA International Conference and is the oldest among DEXA Workshops. The workshop has been very successful in quantity and quality. We received many paper submissions every year, but as a workshop we could accept only a limited number of papers. This was the first year that NBiS was run as an international conference together with DEXA.

We received 122 research papers from all over the world. The submitted papers were carefully reviewed by at least two reviewers. Based on the review results, the Program Committee members selected 55 high-quality papers to be presented during the NBiS-2007 conference.

Many volunteers kindly helped us to prepare and organize the NBiS-2007 International Conference. First of all, we would like to thank all the authors for submitting their papers, Program Committee members, and reviewers who carried out the most difficult task of evaluating the submitted papers. We would like to thank the DEXA Association for giving us the chance to organize the conference. We would like to express our special thanks to Gabriela Wagner for her kind support and help, and also for dealing with conference registration. We would like to thank Kenichi Watanabe and other students of Takizawa Laboratory, Tokyo Denki University, Japan for their hard and timely work to maintain the Web system, distributing of CFP, dealing with the paper submissions, the paper reviewing process, and sending the notification letters to the authors. In addition, we extend our appreciation to the University of Regensburg as Local Organizers. Finally, we would like to thank all the participants of the conference.

June 2007

Tomoya Enokido
Leonard Barolli
Makoto Takizawa

Organization Committee

General Co-chairs

Makoto Takizawa, Tokyo Denki University, Japan
Leonard Barolli, Fukuoka Institute of Technology, Japan

Program Chair

Tomoya Enokido, Rissho University, Japan

Program Committee

Markus Aleksy, University of Mannheim, Germany
Irfan Awan, University of Bradford, UK
Bhed Bahadur Bista, Iwate Prefectural University, Japan
Goutam Chakraborty, Iwate Prefectural University, Japan
Kuo-Ming Chao, Coventry University, UK
Been-Chian Chien, National University of Tainan, Taiwan
S. Misbah Deen, Keele University, England, UK
Alex Delis, Polytechnic University, USA
Andrei Doncesku, University Paul Sabatier, France
Arjan Durresi, Louisiana State University, USA
Thomas Grill, University of Linz, Austria
Lin Guan, Loughborough University, UK
Takahiro Hara, Osaka University, Japan
Naohiro Hayashibara, Tokyo Denki University, Japan
Yoshiaki Hori, Kyushu University, Japan
Hui-huang Hsu, Tamkang University, Taiwan
Ismail Khalil Ibrahim, Johannes Kepler University Linz, Austria
Hiroaki Kikuchi, Tokai University, Japan
Akio Koyama, Yamagata University, Japan
Vincent Lee, Monash University, Australia
Kuan-Ching Li, Providence University, Taiwan
Yinsheng Li, Fudan University, China
Wen-Yang Lin, National University of Kaohsiung, Taiwan
Giuseppe De Marco, Fukuoka Institute of Technology, Japan
Wenny Rahayu, La Trobe University, Australia
Michel Raynal, IRISA-INRIA, France
Fumiaki Sato, Toho University, Japan
Nobuyoshi Sato, Toyo University, Japan
Elhadi Shakshuki, Acadia Univiversity, Canada

Table of Contents

Sensor and Ad-Hoc Networks

Network Security

Secure System Applications

Web Technologies and Middleware Systems

Distributed Systems and Applications

Scheduling

Network Analysis

P2P Systems and Applications

Pervasive and Ubiquitous Systems

Network Applications and Protocols

Embedded Systems

Fuzzy Systems and Their Applications

A Simple Statistical Methodology for Testing Ad Hoc Networks

Makoto Ikeda[1], Giuseppe De Marco[2], and Leonard Barolli[1]

[1] Department of Information and Communication Engineering,
Fukuoka Institute of Technology (FIT)
3-30-1 Wajiro-Higashi-ku, Fukuoka 811-0295, Japan
mgm05001@ws.ipc.fit.ac.jp, barolli@fit.ac.jp
[2] Toyota Technological Institute,
2-12-1 Hisakata, Tenpaku-Ku, Nagoya 468-8511, Japan
demarco@toyota-ti.ac.jp

Abstract. Real-life tests of ad hoc networks are invaluable in order to assess models used in simulation. However, the number of factors affecting the performance of an ad hoc network is high. There are, for example, system factors, such as routing protocols, MAC and physical layer protocols, as well as environment factors, such as the presence of walls, foliage and moving objects. In this regard, it is important to design repeatable experiments of the network, in order to identify the parameters which really affect the system behavior. Here, we leverage methods of statistical testing theory to identify these parameters in a compact manner. In particular, we use OLSR as a routing protocol. Results from real experiments confirm the horizon effect of ad hoc multi-hop networks and shown that there is a treatment effect caused by the window size of OLSR.

1 Introduction

Ad hoc networks are infrastructureless networks, where a number of nodes can interconnect to each other in a decentralized manner. Applications of such networks range from emergence or spontaneous networking to space extension of Internet connections, which is commonly known as mesh networking. So far we can count a lot of simulation results on the performance of ad hoc networks, e.g. in terms of end-to-end throughput, delay and packet loss. However, in order to assess the simulation results, real-world experiments are needed and a lot of testbeds have been built to date [1]. The baseline criteria usually used in real-world experiments is guaranteeing the repeatability of tests. This requirement is very stringent, because in ad hoc networks there are a lot of uncontrollable parameters. Let us think at the wireless channel only. It might happen that in some days the channel is "better" than in other days. Also, effects such as multipath fading may vary along the experiment. Various solutions have been proposed in order to overcome these difficulties. One of the most active project on experimental analysis of ad hoc networks is that of the group at Uppsala University, which implemented a large testbed of 30 nodes [2,3]. They presented

T. Enokido, L. Barolli, and M. Takizawa (Eds.): NBiS 2007, LNCS 4658, pp. 1–10, 2007.

an automatic software called APE which can set and run measurements in an ad hoc network with a particular routing protocol, e.g. AODV, OLSR or LUNAR. The authors of the experiments suggested to use a particular metric to solve the repeatability problem. Here, we propose to use another way to solve this problem which does not need in principle additional metrics. This solution take advantage of the hypothesis test theory, which is often used whenever the experimenter wishes to identify true difference in performance of the system when tested under different scenarios. A scenario in a statistical parlance is a treatment. The results presented here are taken from a real-world testbed of five ad hoc nodes, which was run both in indoor and outdoor scenario. In particular, we used the Open Link State Routing (OLSR) as routing protocol, because it is the most evolving protocol worldwide, especially from the point of view of its software implementation [4,5].

The rest of paper is structured as follows. In Section 2, we review the basic properties of OLSR. In Section 3, we describe the components of our testbed and the methodology used to analyze the data. The application of this methodology is given in Section 4. Conclusions are presented in Section 5.

2 Routing: OLSR

By means of dedicated control messages, OLSR modifies the routing tables of the node it runs on. Its main tasks are neighbors sensing, Multi-Point Relaying (MPR) calculation and Topology Control (TC) messages dissemination. Neighbor sensing is performed by sending periodic broadcast HELLO messages with rate T_{HELLO}^{-1}. In OLSR, there are cross-layer operations to some extent. For example, in order to check the presence of a neighbor, OLSR has to compute a quality metric of all links towards neighboring nodes. According to the value of this metric for both directions of links, OLSR can judge upon the symmetry of links. Asymmetric links are discarded. It is well known [6] that hop-cont based metrics do not result in high throughput, because the routing protocol could choose a worse path (e.g. high delay) even if it has the minimum hop-count. A better metric is based on the Expected Transmission Count (ETX). Every node computes the forward and backward packet loss rate for every neighbor link. The backward packet loss rate is the packet loss "seen" by the neighbor. Then, the ETX is computed as $ETX = \frac{1}{p_f p_r}$ where p_f is the estimate of the forward packet loss rate and p_r is the reverse packet loss rate. This quantity is the mean number of re-transmissions per packet we have to wait for successful transmission over a particular link. The latter quantities are computed within a normalized time window whose default value is $w = 10$. For example, if $T_{\mathrm{HELLO}} = 0.5$s, we have to wait 20s in order to compute a sample of ETX. Since we used single-radio NICs, we did not use more advanced metrics, like the Weight Cumulative Expected Transmission Time (WCETT) [7]. For instance, WCETT is conceived for multi-radio mesh networking.

There are several implementation of OLSR. However, the most known and update open project is OLSRd [4], originally implemented by Andreas Tønnesen.

In our opinion, the true strength of OLSRd is its flexibility. In fact, its plug-in based structure allows to easily implement user-defined functionality, like new control messages and additional routing services.

3 Testbed

The testbed was composed by five laptop computers and one desktop machine acting as gateway. Every machine ran Linux OS with kernel 2.6.17.x. The gateway was always located inside an office room and it served as coordination point of all measurements campaign. The indoor experiments have been carried out by interspersing laptops within a departmental floor near our lab. The total length of the floor was about 20m. The outdoor experiments have been performed in an open area, where moving cars, moving people and parking lots were present. Along the presentation, we use the following definitions.

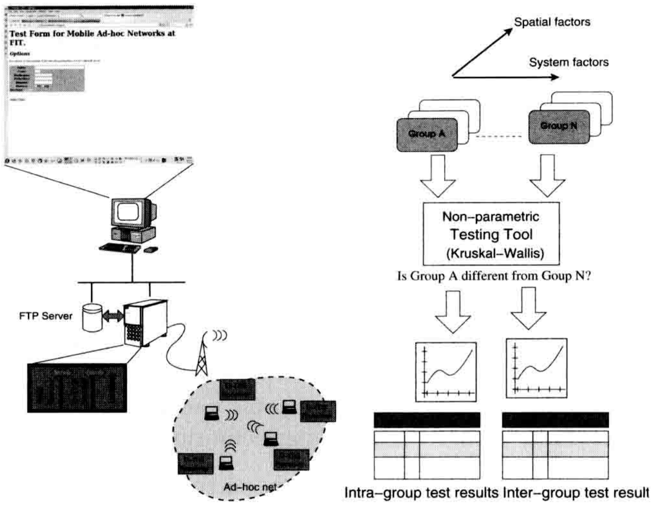
Definition 1. *We refer to hop-distance between two nodes as the absolute difference in their IP addresses.*

Definition 2. *We refer to connection horizon as the maximum hop-distance for which there are not core performance differences or rather performance degradations.*

For example, the hop-distance between *.*.*.1 host and *.*.*.5 host is 4. Actually, the hop-distance might be different from the hop-count of the route selected by OLSR, because high quality direct links can exist. For example, the hop-distance from 1 to 5 is 4, but if there is a direct link between 1 and 5 with high LQ, the hop-count is 1. This depends on the type of topology. The traffic generation has been carried out by means of the open source Distributed Internet Traffic Generator (D-ITG) [8]. The constant bit rate of the generated traffic was set to 122pkt/sec, i.e 499.712Kbps. A web interface was provided in order to let the system acquiring all these and other settings, such as the duration of an experiment, the type of network protocol, as shown in Fig. 1-a.

3.1 IEEE 802.11 MAC

As MAC protocol, we used IEEE 802.11b. The transmission power was set in order to guarantee a coverage radius equal to the maximum allowed geographical distance in the network, especially for the outdoor experiments. For instance, by considering also shadowing phenomena, a value of 250m was fairly enough. Accordingly, the transmission power was set to 20dbm. Since we were interested mainly in the performance of the routing protocol, we kept unchanged all MAC parameters. In regard of the interference, it is worth noting that, during our tests, almost all the IEEE 802.11 spectrum had been used by other access points disseminated within the campus. In general, the interference from other access points is a non controllable parameter. We used wireless USB NICs, which were equipped with an external omnidirectional antenna. During the first trials of



(a) (b)

Fig. 1. Schematic illustration of (a) the testbed and (b) the methodology of analysis

the experiments, we used on-board wireless cards and PCMCIA cards as well. However, we experienced an unpredictable loss of the signal strength which motivated us to turn-off the built-in cards and to use an external antenna-based NIC.

3.2 Topology

Other testbeds [9] used hardware artifacts to control the physical topology of the network, e.g. masking the wireless cards by means of copper plates. Although this procedure can fairly build the desired physical topology, it requires a try-and-test approach, which complicate the setup of the network, especially in the outdoor scenarios. We selected a suboptimal technique, by using the logical topology instead of the physical one. In fact, by means of MAC address filtering, it is possible to quickly build the desired topology[3]. The penalty of this technique is that some residual self-induced interference remains, because the transmitted signal can reach more nodes than those required by the logical topology. We used two kind of topologies, the Linear Topology (LT) and the Meshed Topology (MT). In LT, hop-distance equals the hop-count.

3.3 Methodology

The experimenter shall choose a set of independent parameters, which will be varied during the measurement campaign. The objective of the experimenter is testing whether core differences are present in the system when one of the independent parameters is changed. In statistical parlance, these parameters are

[3] In UNIX-like systems, this is readily done by the `iptables` command.

referred to as *factors*. A *treatment* is determined by a particular choice of factors. In this work, we assumed as factors: the hop-distance between two hosts, the environment (Indoor or Outdoor), the type of topology (LT or MT) and the window size w. As additional factors, we used the transport protocol type, i.e. TCP and UDP. The methodology consists in finding whether these factors impact on the communication performance of the ad hoc network. Precisely, let consider A_N groups of experiments or treatments, where for every A_k and A_j, $k \neq j$, at least one factor value is different. Every A_i is a collection of measurements, i.e. a sample, taken from the testbed. For example, in our testbed, a sample is the collection of 10s measurements of a particular metric. The question is: Do the factor differences really impact on the performance? Usually, many experiments we found in literature used just the mean value of some metric, such as the throughput and the delay. But it is very difficult to see the difference of a group of mean values by inspecting graphical plots. Even numerical data could say nothing about this difference, because data are taken from experimental data. Moreover, outliers and spurious values may arise, as well as different length of the samples. Consequently, the metrics depends on the size of the sample. One could perform long measurements in order to get a "big" sample. But how big? Another more elegant solution is to use a non-parametric hypothesis test among groups, see Fig. 1-b. By loosely speaking, given $1 \leq i \leq N$, the hypothesis test $\mathbf{Y}$ can be formulated as follows:

$$\mathbf{Y} \begin{cases} \mathcal{H}_0 : A_i\text{'s have different means} \\ \mathcal{H}_1 : \text{all } A_i\text{'s have equal means.} \end{cases}$$

The hypothesis $\mathcal{H}_1$ states that there is no difference in the mean of the samples (i.e. samples can be considered as extracted from the same population). On the other hand, if the null hypothesis is true, then we have discovered a core dependence on the factors used in the test. Every method used to perform the test is associated with some probability of error, which is the probability of rejecting the null hypothesis while it is true. The error probability is given in term of the significance level of the test, α. We use the Kruskal-Wallis (KW) test. The null hypothesis of the KW test is that the medians among samples are different. The test formula is not explained here and can be found in [10]. For the KW test, we use the common value of $\alpha = 0.05$. If the KW returns a value smaller than α then the sample medians can be considered different, otherwise we can declare that there is no "treatment effect".

We assume that there are two kind of factors: system factors and spatial factors, as shown in Fig. 1-b. System factors depend on the settings of networking protocol parameters, while spatial factors depends on the physical position of nodes. By incorporating the hop-distance in the spatial factors, we can analyze to which extent the hop-distance impacts on the performance. For sake of clarity and simplicity, the treatments presented in this work have been arranged as shown in Table 1. The testing procedure is performed as follows:

1. By means of Web interface, we set the parameters of the experiments, in particular its duration, the logical topology and the type of transport protocol.

2. At the gateway machine, we collect the samples for: throughput/goodput (T), Round-Trip Time (RTT) and packet loss (P_L)[4].
3. A MATLAB script picks the samples and executes the KW test. The outcomes of the scripts are: the medians of the metrics along with their box plots, and α.

Table 1. Treatments. I=Indoor; O=Outdoor, LT=linear topology; MT=meshed topology.

Treatments	Factor values	
	$w = 10$	$w = 20$
A	(UDP, I, LT)	(UDP, I, LT)
B	(UDP, I, MT)	(UDP, I, MT)
C	(UDP, O, LT)	(UDP, O, LT)
D	(UDP, O, MT)	(UDP, O, MT)
E	(TCP, I, LT)	(TCP, I, LT)
F	(TCP, I, MT)	(TCP, I, MT)
G	(TCP, O, LT)	(TCP, O, LT)
H	(TCP, O, MT)	(TCP, O, MT)

Table 2. Intra-group significance levels for $w = 10$

Treatments	α		
	Goodput	Delay	Loss
A	0.00	0.00	0.0047
B	0.00	0.00	0.00
C	0.00	0.00	0.00
D	0.0099	0.00	0.00
E	0.00	0.00	0.00
F	0.0607	0.00	0.00
G	0.00	0.00	0.00
H	0.00	0.00	0.00

4 Measurement Results

Every experiment lasted 10s and it has been repeated 50 times. OLSR is continuously active during the experiment. Therefore, OLSR has enough estimates of packet losses to compute LQ values. In fact,

$$wT_{\text{HELLO}} < T_{\text{Exp}} ,$$

where T_{Exp} is the total duration of the experiment, i.e., in our case, $T_{\text{Exp}} =$ 500s. However, the testbed was turned on even in the absence of measurement traffic. Therefore, the effective T_{Exp} was much greater. We run the experiments

[4] Excepting the case of TCP, where obviously the packet loss is always 0.