# Journal of Networks

ISSN 1796-2056
Volume 7, Number 1, January 2012

## Contents

### Special Issue: Performance Modeling and Evaluation of Computer and Telecommunication Systems

### Guest Editor: Mohammad S. Obaidat and José Luis (Sevi) Sevillano

Special Issue on Performance Modeling and Evaluation of Computer and Telecommunication Systems

# Guest Editorial

This special issue of the Journal of Networks on "Performance Modeling and Evaluation of Computer and Telecommunication Systems" includes extended versions of selected best papers accepted and presented at the 2010 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS 2009). Six papers were selected based on their excellent review scores. Their authors were invited to submit extended versions which have undergone a second review process. This selection addresses a variety of topics related to the Performance Modeling and Evaluation of Computer and Telecommunication Systems.

The first paper "Software Performance Modeling using the UML: a Case Study", by Issa Traore, Isaac Woungang, Ahmed Awad El Sayed Ahmed and Mohammad S. Obaidat, describes the design of annotated UML (Unified Modeling Language) performance models for the performance analysis of distributed software systems, based on the UML profile for Schedulability, Performance and Time. An approach previously proposed by the same authors, named Model-Driven SPE (MDSPE), is used. An outline of system performance models and metrics is provided and a case study of a business system is used to validate the stated goal.

The second paper is "User-centric Mobility for Multimedia Communications: Experience and Evaluation from a Live Demo", authored by Raffaele Bolla, Riccardo Rapuzzi and Matteo Repetto. Session migration, a key issue in pervasive communications, is evaluated with a user-centric vision. Users' feelings are evaluated with a user-centric networking mobility framework based on the concept of Personal Address. The user evaluation is carried out by a live demo open to a large heterogeneity of potential users at a national science exhibition, through written questionnaires and direct interviews, with the support of a psychologist skilled in this field. General indications for the whole research community about user's expectations and requirements for session migration are presented.

In the third paper, "Tracking Per-Flow State – Binned Duration Flow Tracking", authored by Brad Whitehead, Chung-Horng Lung and Peter Rabinovitch, the issue of network monitoring is addressed. A time efficient novel method – Binned Duration Flow Tracking (BDFT) – of tracking per-flow state by grouping valid flows into "bins" is presented. BDFT is intended for high-speed routers where CPU time is crucial. BDFT is time and space efficient thanks to the use of counting Bloom filters as the data structure that represents the bins. Simulation results show that BDFT can achieve over 99% accuracy on traces of real network traffic.

The fourth paper, "Impact of Retransmission Mechanism on SIP Overload: Stability Condition and Overload Control", is authored by Yang Hong, Changcheng Huang and James Yan. A discrete time fluid model is created to study the impact of retransmission mechanism on SIP (Session Initiation Protocol) overload. The authors also derive a sufficient stability condition that a SIP server can handle the overload effectively under the retransmission mechanism. Simulation results are presented using both a fluid-based Matlab simulation and an event-driven OPNET simulation. A simple overload control solution is also proposed to mitigate the overload by reducing the retransmission rate only, while keeping the original message rate unchanged.

Finally, the fifth paper is entitled "Improving the Resilience of Transport Networks to Large-scale Failures", by Juan Segovia, Pere Vilà, Eusebi Calle and Jose L. Marzo. Two heuristic-based link prioritization strategies for improving network resilience are proposed, which are evaluated through simulation on a large synthetic topology that represents a GMPLS-based transport network. In the studied scenario, several links of the GMPLS-based network fail concurrently in random locations, provoking the loss of all the end-to-end connections. The aim is to identify elements (e.g., links) to which extra network protection can be applied so that the impact of such failure events, in terms of the number of connections affected, is minimized. Results show that the two proposed strategies succeed in decreasing the number of affected connections.

The guest editors would like to thank all the authors and reviewers for their valuable contributions to this special issue. We hope that the papers selected in this special issue will become useful resources for researchers and practitioners in the area of Performance Modeling and Evaluation of Computer and Telecommunication Systems.

**Guest Editors**

**Mohammad S. Obaidat,** Fellow of IEEE and Fellow of SCS
President, Society for Modeling and Simulation International, SCS
Editor-in-Chief, International Journal of Communication Systems, Wiley
Professor of Computer Science and Software Engineering, Monmouth University, W. Long Branch, NJ 07764, USA
E-mail: obaidat@monmouth.edu
http://www.monmouth.edu/mobaidat


doi:10.4304/jnw.7.1.1-3

**José Luis (Sevi) Sevillano**
Associate Professor, Department of Computer Architecture, University of Seville, Seville, Spain
E-mail: sevi@atc.us.es

**Professor Mohammad S. Obaidat** is an internationally well known academic/researcher/ scientist. He received his Ph.D. and M. S. degrees in Computer Engineering with a minor in Computer Science from The Ohio State University, Columbus, Ohio, USA. Dr. Obaidat is currently a full Professor of Computer Science at Monmouth University, NJ, USA. Among his previous positions are Chair of the Department of Computer Science and Director of the Graduate Program at Monmouth University and a faculty member at the City University of New York. He has received extensive research funding and has published Ten (10) books and over Five Hundred (500) refereed technical articles in scholarly international journals and proceedings of international conferences, and currently working on three more books. Prof. Obaidat is the author of a new upcoming book: Wireless Sensor Networks (Cambridge University Press). He is also the editor to 2 new upcoming books: Cooperative Networking (John Wiley &Sons 2010) and Pervasive Computing and Networking (John Wiley &Sons 2010). Prof. Obaidat is the author of the book entitled: "Fundamentals of Performance Evaluation of Computer and Telecommunications Systems," by John Wiley & Sons in 2010. Dr. Obaidat is the Editor of the Book entitled, "E-business and Telecommunication Networks", published by Springer in 2008. He is the co-author of the book entitled, "Security of e-Systems and Computer Networks" published by Cambridge University Press in 2007. He is the co-author of the Best Selling Book, "Wireless Networks" and "Multiwavelength Optical LANs" published by John Wiley & Sons (2003). Obaidat is the editor of the book, APPLIED SYSTEM SIMULATION: Methodologies and Applications, published by Kluwer (now Springer) in 2003. Professor Obaidat has served as a consultant for several corporations and organizations worldwide. Mohammad is the Editor-in-Chief of the International Journal of Communication Systems published by John Wiley. He served as an Editor of IEEE Wireless Communications from 2007-2010. Between 1991-2006, he served as a Technical Editor and an Area Editor of Simulation: Transactions of the Society for Modeling and Simulations (SCS) International, TSCS. He also served on the Editorial Advisory Board of Simulation. He is now an editor of the Wiley Security and Communication Networks Journal, Journal of Networks, International Journal of Information Technology, Communications and Convergence, IJITCC, Inderscience. He served on the International Advisory Board of the International Journal of Wireless Networks and Broadband Technologies, IGI-global. Prod. Obaidat is an associate editor/ editorial board member of seven other refereed scholarly journals including two IEEE Transactions, Elsevier Computer Communications Journal, Kluwer Journal of Supercomputing, SCS Journal of Defense Modeling and Simulation, Elsevier Journal of Computers and EE, International Journal of Communication Networks and Distributed Systems, The Academy Journal of Communications, International Journal of BioSciences and Technology and International Journal of Information Technology. He has guest edited numerous special issues of scholarly journals such as IEEE Transactions on Systems, Man and Cybernetics, SMC, IEEE Wireless Communications, IEEE Systems Journal, SIMULATION: Transactions of SCS, Elsevier Computer Communications Journal, Journal of C & EE, Wiley Security and Communication Networks, Journal of Networks, and International Journal of Communication Systems, among others. Obaidat has served as the steering committee chair, advisory Committee Chair and program chair of numerous international conferences including the IEEE Int'l Conference on Electronics, Circuits and Systems, IEEE International Phoenix Conference on Computers and Communications, IEEE Int'l Performance, Computing and Communications Conference, IEEE International Conference on Computer Communications and Networks, SCS Summer Computer Simulation Conference, SCSC'97, SCSC98-SCSC2005, SCSC2006, the International Symposium on Performance Evaluation of Computer and Telecommunication Systems since its inception in 1998, International Conference on Parallel Processing, Honorary General Chair of the 2006 IEEE Intl. Joint Conference on E-Business and Telecommunications, ICETE2006. He served as General Co-Chair of ICETE 2007-ICETE 2010. He has served as the Program Chair of the International Conference on Wireless Information Networks and Systems from 2008-Present. He is the co-founder and Program Co-Chair of the International Conference on Data Communication Networking, DCNET since its inception in 2009. Obaidat has served as the General Chair of the 2007 IEEE International Conference on Computer Systems and Applications, AICCSA2007, the IEEE AICCSA 2009 Conference, and the 2006 International Symposium on Adhoc and Ubiquitous Computing (ISAHUC'06). He is the founder of the International Symposium on Performance Evaluation of Computer and Telecommunication Systems, SPECTS and has served as the General Chair of SPECTS since its inception. Obaidat has received a recognition certificate from IEEE. Between 1994-1997, Obaidat has served as distinguished speaker/visitor of IEEE Computer Society. Since 1995 he has been serving as an ACM distinguished Lecturer. He is also an SCS distinguished Lecturer. Between 1996-1999, Dr. Obaidat served as an IEEE/ACM program evaluator of the Computing Sciences Accreditation Board/Commission, CSAB/CSAC. Obaidat is the founder and first Chairman of SCS Technical Chapter (Committee) on PECTS (Performance Evaluation of Computer and Telecommunication Systems). He has served as the Scientific Advisor for the World Bank/UN Digital Inclusion Workshop- The Role of Information and Communication Technology in Development. Between 1995-2002, he has served as a member of the board of directors of the Society for Computer Simulation International. Between 2002-2004, he has served as Vice President of Conferences of the Society for Modeling and Simulation International SCS. Between 2004-2006, Prof. Obaidat has served as Vice President of Membership of the Society for Modeling and Simulation International SCS. Between 2006-2009, he has served as the Senior Vice President of SCS. Currently, he is the President of SCS. One of his recent co-authored papers has received the best paper award in the IEEE AICCSA 2009 international conference. He also received the best paper award for one of his papers accepted in IEEE GLOBCOM 2009 conference. Dr. Obaidat received very recently the Society for Modeling and Simulation Intentional (SCS) prestigious McLeod Founder's Award in recognition of his outstanding technical and professional contributions to modeling and simulation.

He has been invited to lecture and give keynote speeches worldwide. His research interests are: wireless communications and networks, telecommunications and Networking systems, security of network, information and computer systems, security of e-based

systems, performance evaluation of computer systems, algorithms and networks, high performance and parallel computing/computers, applied neural networks and pattern recognition, adaptive learning and speech processing. Recently, Prof. Obaidat has been awarded a Nokia Research Fellowship and the distinguished Fulbright Scholar Award. During the 2004/2005, he was on sabbatical leave as Fulbright Distinguished Professor and Advisor to the President of Philadelphia University in Jordan, Dr. Adnan Badran. The latter became the Prime Minister of Jordan in April 2005 and served earlier as Vice President of UNESCO. Prof. Obaidat is a Fellow of the Society for Modeling and Simulation International SCS, and a Fellow of the Institute of Electrical and Electronics Engineers (IEEE).

**Dr. José Luis (Sevi) Sevillano** received his degree in Physics (electronics) and his Ph.D. from the University of Seville (Spain) in 1989 and 1993 respectively. From 1989 to 1991 he was a researcher supported by the Spanish Science and Technology Commission (CICYT). After being Assistant Professor of Computer Architecture at the University of Seville, since 1996 he is Associate Professor at the same University. He has served as Vice Dean of the Computer Engineering School (2004-7) and as Director of Innovations for Teaching (2007-8) at the University of Seville. Currently, he is Coordinator of the Telefónica Chair on Intelligence in Networks, University of Seville, Spain.

Since 2007 Prof. Sevillano is Associate Editor of the *International Journal of Communication Systems*, published by John Wiley. He is also Associate Editor of *Simulation* (Sage). Since 2009, he serves as Vice-President for Membership of The Society for Modeling & Simulation International (SCS). He also has served on several international conferences: as General Chair (SPECTS-11, ICETE 2011), as Program Co-Chair (ACS/IEEE AICCSA 2009, DCNET 2010, SPECTS-2009, SPECTS-2010), as well as member of the TPC. He is also a member of the Steering Committee of the International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS). One of his recent co-authored papers received the Best Paper award of the 13th Communications & Networking Simulation Symposium (CNS 2010). He is author/co-author of more than 60 research reports and papers in refereed international journals and conferences, and has participated in more than 20 research projects and contracts.

# Software Performance Modeling using the UML: a Case Study

Issa Traore
Department of Electrical and Computer Engineering, University of Victoria
Victoria, B.C., Canada
Email: aahmed@ece.uvic.ca

Isaac Woungang
Department of Computer Science, Ryerson University
Toronto, Ontario, Canada
Email: iwoungan@scs.ryerson.ca

Ahmed Awad El Sayed Ahmed
Department of Electrical and Computer Engineering, University of Victoria
Victoria, B.C., Canada
Email: aahmed@ece.uvic.ca

Mohammed S. Obaidat
Department of Computer Science, Monmouth University
West Long Branch, NJ 07764, USA
Email: obaidat@monmouth.edu

***Abstract*[1] —The performance analysis of distributed software systems is a challenging task in which the assessment of performance measures is a vital step. Due to its versatility, the concept of software performance engineering (SPE) has been advocated as a promising solution towards realizing that step. This paper illustrates how by using our recently proposed Model-Driven SPE (MDSPE) approach, one can design annotated UML performance models for the performance analysis of distributed software systems, based on the UML profile for Schedulability, Performance and Time. An outline of system performance models and metrics is provided and a case study of a business system is used to validate the stated goal.**

***Index Terms* —Software performance engineering process; UML profile for schedulability, performance and time; distributed software systems.**

## I. INTRODUCTION

In the design of complex distributed software systems, analyzing the software performance requirements is a mandatory step towards evaluating alternative designs that would produce the best effort architecture – i.e. an architecture that meets the quality of service (QoS) requirements and cost constraints. This step can be achieved by designing suitable performance models that can be used to identify the performance critical scenarios and associated requirements, thereby, allowing for the prioritization and achievability of different performance goals, and their impact on the design.

Several performance modeling approaches have been proposed in the literature, including simulation-based approaches [1, 2], model-based approaches, particularly UML-based approaches [4, 5, 6, 7], to name a few. Ideally, prior to computing and evaluating performance measures, performance requirements should be captured and expressed in the software architecture design. That way, performance concerns can be assessed along with other important software quality attributes such as security, reliability, and maintainability.

This paper argues that the *UML Profile for Schedulability, Performance, and Time* [8] introduced a few years ago provides an important ground toward achieving this goal. We describe the application of our recently proposed Model-Driven SPE (MDSPE) methodology [1] based on UML diagrams with annotations taken from the above-mentioned profile. We provide background information on basic performance modeling concepts and metrics and then present a case study illustrating the application of the above-mentioned profile in expressing performance concerns during the software architecture design phase.

The rest of the paper is organized as follows. In Section 2, few representative works on software performance modeling using UML concepts are discussed and contrasted. In Section 3, we introduce our novel MDSPE process. In Section 4, we provide an outline of the performance modeling concepts and metrics. In Section 5, we describe the main components of the above-mentioned UML profile that deal

[1] An abridged version of this paper (referenced here as [1]) has been published as follows: I. Traore, I. Woungang, A. A. E. Ahmed, and M. S. Obaidat, " UML-Based Performance Modeling of Distributed Software Systems", Proc. of IEEE International Symposium on Performance Evaluation of Computer Telecommunication Systems (SPECTS 2010), July 11-13, Ottawa, Canada, pp. 119-126, 2010.


doi:10.4304/jnw.7.1.4-20

specifically with performance modeling, along with the semantics of the performance values - i.e. a description of how these values can be interpreted as UML tagged values. In Section 6, a case study of a distributed software system is presented. In Section 7, performance testing concepts are presented and experimented using the above-mentioned case study. Finally, in Section 8, we present some concluding remarks.

## II. Related Work

In the recent years, software performance engineering (SPE) has been advocated as both a method and tool that can be used for the evaluation of performance estimates of software systems in the early stage of the software life cycle. In this regards, the election of the Unified Modeling Language (UML) [10] as the standard reference notation to describe the operations of software systems and their performance requirements, has allowed the emergence of several approaches where UML is considered as design language for the SPE process. Few of these works that inspired the topic presented in this paper are summarized in the following.

In [11] Street et al. focused on the use of UML profile for Schedulability Performance, and Time (SPT) in conjunction with coloured Petri nets and statistical simulations for analyzing the performance modeling of object-oriented software systems, by highlighting few lessons learned from this experience. The main highlighted advantage is that UML SPT profile significantly outperforms other known techniques due to its small learning curve since it can be applied directly to existing UML 1.4 models; however, the practical application of SPT is limited because there are less tools and techniques available for modeling using the SPT profile.

As a follow up, a recent paper by Garousi [12] introduced a UML-based method where control flows in sequence diagrams and interaction overview diagrams are analyzed to produce performance bottlenecks in concurrent real-time software.

Similarly, in [13], Cortellessa et al. investigated the open problem of formalizing anti-patterns and detecting them automatically in design models. A UML-driven approach using Object Constraint Language (OCL) engine [14] is proposed as a solution and validated using a case study in UML annotated with the MARTE profile.

In [15], Abdullatif et al. proposed a UML-JML based tool that can be used for studying the expected performance characteristics of an architectural design. Following the same trend, Distefano et al. [16] introduced a technique to identify and define rules and guidelines for specifying UML-ArgoPerformance compliant models, which can be used for implementing the SPE development process.

In [17], a declarative method for modeling the performance impact of container middleware to component-based systems is proposed, which is based on UML activity diagram with UML SPT annotations. Using this approach, the performance impacting factors can automatically be mapped into application UML model. With regards to component-based software systems, a state-of-art review of approaches that target performance predictions during design time for these systems modeled with UML is given in [18].

Similarly, Bernardi et al. [19] investigated SPE approaches that deal with using UML diagrams for specifying the functional and performance requirements of a system. They proposed a method that combines UML and Petri Nets to achieve this same goal.

In [20], an initial UML-based approach to SPE using stress testing is proposed, which can help developers cope with performance related real-time faults in UML-driven developments. To achieve this same goal, but using simulation, Marzolla [21] proposed a technique for translating the UML software specifications into simulation models, thereby defining a set of annotation of UML specifications that can be used to add quantitative performance-oriented information using a UML SPT profile.

In [22], the above same goal was achieved by designing UML-driven performance models based on multi-chain and multiclass queuing networks. In this case, the UML model was annotated according to the UML SPT profile, and the technique consisted in translating the annotated UML specification into QN performance models; then analyzed these models using standard solution methods.

Similar to few of the above-mentioned related works, the UML-based performance model described in this paper targets primarily an effective integration between functional analysis and performance analysis activities. Its design follows the architecture proposed by OMG [8] in the UML Profile for Schedulability, Performance and Time Specification. This model is also relatively easy to construct and evaluate with the performance modeling tools available today.

## III. Proposed MSDPE Process

Software Performance Engineering (SPE) is a process through which performance concerns are taken into account when applying the traditional software engineering activities [2, 3]. In [1], we have proposed a model-driven SPE process referred to as MDSPE process), which consists of deriving the performance models from the UML specifications, annotated according to the Object Management Group (OMG) profile for schedulability and performance [8]. The operational aspects of this MDSPE process integrate functional analysis activities and performance analysis ones as shown in Fig. 1. A detailed description of these aspects can be found in [1]. The part of this process that deals with performance modeling and analysis is represented by the *Prediction* step, right after the design step (as shown in Figure 1). It consists of *Performance Annotation* and *Performance Analysis* steps.

Assuming that the performance measures are known, the *Performance Annotation* step (also referred to as performance modeling step) consists of using the above-mentioned UML profile to encapsulate the performance

characteristics of the hardware and networking infrastructure, as well as the QoS requirements of specific functions, into the software architecture derived from the design step. The result is an annotated model for performance.

The *Performance Analysis* step consists of submitting the obtained annotated model to a performance analyzer in order to compute the performance metrics, thereby predict the software performance. Using these predictions, alternative designs can be evaluated in order to decide the one that is best suited for the implementation step. The rest of this paper will focus on the *Performance Annotation* step.

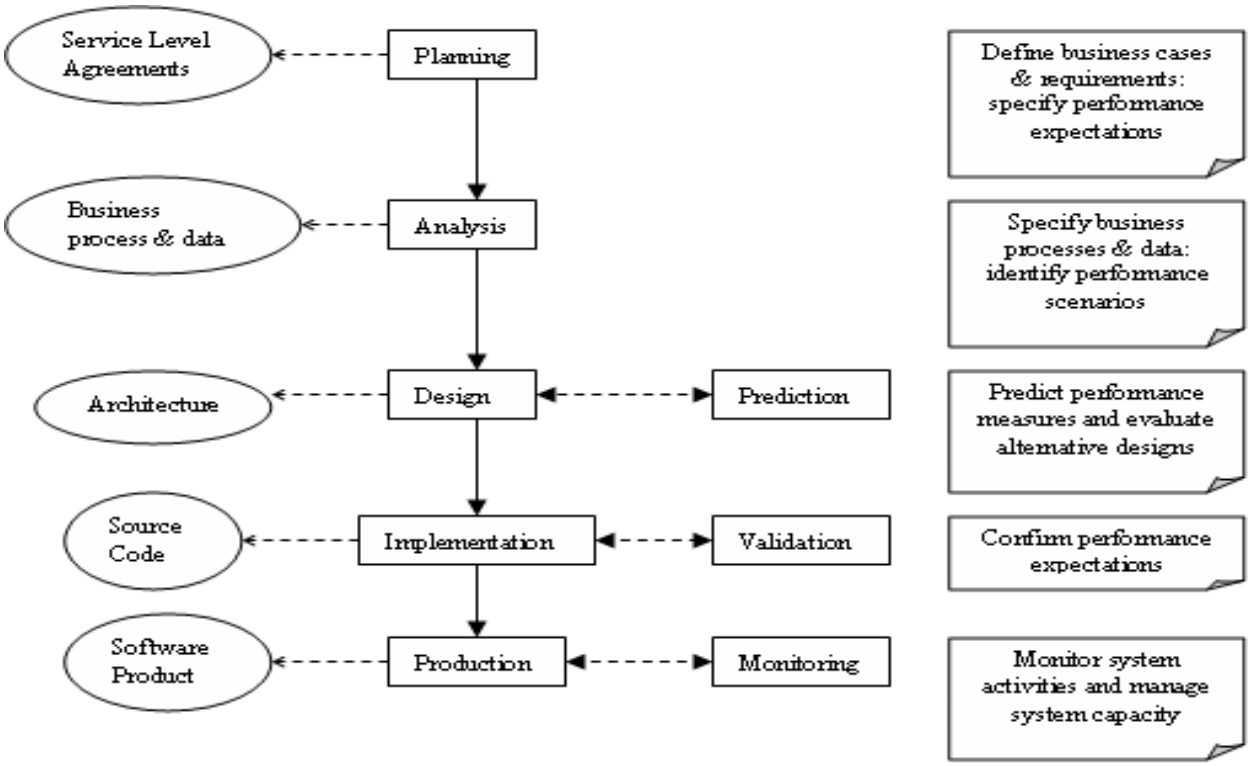



Figure 1: MDSPE process.

## IV. Background on Performance Modeling

In this section, we provide an outline of basic performance concepts, models, and metrics, which are essential elements of any performance modeling and analysis undertaking.

### *A. Performance Parameters*

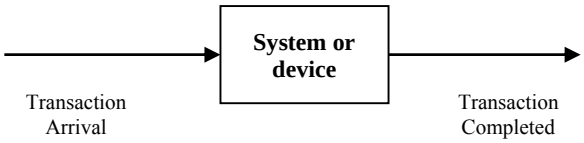



Figure 2: Workload arrival and completion.

Figure 2 depicts a simplified view of a computing system or device. The system is submitted to some workloads arriving at a given rate. The workloads are processed and then leave the system at certain rate. In order to study performance and scalability of a computing system, we are interested in measuring directly or indirectly the workload arrival and completion rates, as well as the delay between the arrival and completion for a given workload. The completion rate corresponds to the throughput and the delay corresponds to the response time. We also need to understand the internal of the system. This is characterized by resource utilizations and service demands. Basic performance characteristics that need to be captured and expressed during performance analysis include the notions of *workload*, *throughput*, *utilization*, *service demand*, and *response time*. We provide a brief definition for each these notions as follows.

- ***Workload:*** Represents an amount or type of job submitted to a system for processing. Examples of workloads include interactive user request, batch jobs, SQL requests, request from client workstation to server machines etc. Workloads received by a web site may include for instance "select", "order", "search" for respectively item selection, placing an order or searching an item. A workload is characterized by its type, size, and frequency. Workload frequency also called workload intensity or workload arrival rate is measured in terms of number of requests received per time unit (e.g. req/sec, req/hour, req/day etc.). Workload size can be measured in terms of bits, bytes etc. Workloads of the same size or of the same statistical characteristics can be grouped into the same class, and treated for performance analysis purpose as a single workload.

There are two categories of workloads: *open* or *closed*. An open workload is a sequence of requests arriving with a known pattern or distribution such as Poisson for instance. A closed workload is characterized by a fixed number of requests or jobs or potential users. Another parameter of a closed workload is the *think time*, which is the elapsed time between the end of one response and the next request.

- ***Throughput:*** Gives a measure of the amount of work processed or completed per time unit. For instance, the throughput of a database server corresponds to the number of database requests processed by the server per time unit. Common throughput measurement units include transaction per second (tps), hits per second (hits/sec) etc.
- ***Utilization:*** The utilization of a given resource (e.g. CPU, Disk, memory, network) gives a measure of the level of use of this resource in percentage. A resource is either busy ($0\%<U\leq100\%$) or idle ($U=0\%$). The maximum possible utilization for a resource is 100%.
- ***Service Demand:*** The service demand of a resource for a given request type corresponds to the time used by the resource to process an instance of this request. It can be measured in seconds, milliseconds etc. The service demand is function of the resource service time and of the number of visits to the resource needed to process the request. The service time is specific to the resource and measure the time used to execute an elementary job unit. For instance, for a processor the elementary job unit corresponds to one instruction. So the service time corresponds in this case to the time used by the processor to execute one instruction. The number of visits required to process a given request will correspond to the number of instructions involved in this request. For a disk, the elementary job unit corresponds to a single input/output. Hence the service demand of a request can be calculated using the following formula: $D = V \times S$, where $V$ and $S$ are respectively the number of visits corresponding to the request and the service time of the resource.
- ***Response Time:*** Correspond to the time or delay between the submission of a request and its completion. The response time can be measured in milliseconds, seconds, hours etc.

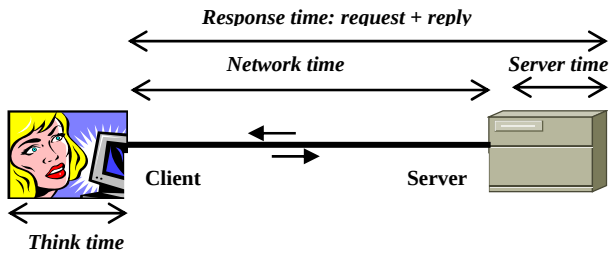



Figure 3: Client-Server Interaction

The notion of response time can be viewed from two different perspectives in client-server applications: *end user response time* and *server response time*. The server response time captures the duration between the instant when a request arrives at the server until the request processing is completed and the response is ready to be sent back. In that case only the server is taken into account in the measurement effort. End user response time captures the duration between the instant from which the request is issued at the client workstation until the result appears on the user's screen. The system's components involved in this scheme include the user's workstation, and the servers accessed from that workstation and the network segments linking them. The end user response time, in client-server application, can be broken down into the network (service) time and the server time as shown by Figure 3. The network time corresponds to the time needed by the network to transmit the request to the server and send back the reply to the client. The delay between the time a request is submitted and the start of the response is the *reaction time*. The *think time* is the average time between two consecutive requests submitted by a client.

### *B. Performance Models and Metrics*

Performance models come in two flavors: system-level models and component-level models. System-level models view the system as a black box while component-level models consider explicitly the resources involved [2], [5]. Choosing either models depend mostly on the availability of detailed information about the system parameters. Component level models are based on queuing theory. Each resource is represented as a queue. The interconnection of the different queues involved in the system gives a queuing network model (QNM). Several assumptions are made in the definition of performance models. Some of these assumptions include the following:

- *Infinite versus finite population* assumption: an infinite population model assumes that the user population size is unknown, and as such it may be considered infinite. In contrast, a finite population model assumes that the user population is known in advance.
- *Infinite versus finite queue* assumption: an infinite queue model assumes that all arriving requests at the queue will eventually be processed, i.e., no request is refused. In contrast, a finite queue model assumes that there is a maximum limit on the queue size, i.e., any request that arrives on top of this limit is rejected.
- *Operational equilibrium* assumption: the number of requests present in the system at the start of an appropriate observation period should be the same as that at the end of the period, even though the number may vary in between.
- *Single-class (*also called *homogeneous)* assumption: workload units with the same

statistical characteristics can be combined into a single class and analyzed as a single workload.

Real-life systems typically involve several classes of requests, so they can be studied using multi-class models. Moreover, single-class models are particular cases of multi-class models.

Multi-class models allow the prediction of performance for systems involving different workloads or different classes of workloads. A separate class represents each unit of work. Assume that we have $m$ queues and $n$ classes. Given a workload $i$ $(1 \le i \le n)$ and a queue $j$ $(1 \le j \le m)$, the utilization of resource $j$ created by workload $i$ is given by:

$$U_{ij} = \lambda_i D_{ij}$$

where $\lambda_i$ and $D_{ij}$ stand for the workload intensity for class $i$ and the service demand of workload i at resource $j$, respectively.

The total utilization $U_j$ of resource $j$ is computed as the sum of all the utilizations created by the different workloads as follows:

$$U_j = \sum_{i=1}^{n} U_{ij}$$

The residence time $R_{ij}$ for workload $i$ at resource $j$ is defined as follows:

$$R_{ij} = \frac{D_{ij}}{1 - U_j}$$

The response time $R_i$ of workload $i$ is computed as the sum of the residence times over all the resources:

$$R_i = \sum_{i=1}^{m} R_{ij}$$

In the above formulas, $n$ and $m$ denote the total number of workload classes and the total number of resources respectively.

## V. UML-Based Performance Modeling

The *UML Profile for Schedulability, Performance, and Time* [8] is a section of the UML standard, which supports the specification of performance and real-time requirements, via the following activities:

- Performance requirements capture.
- Integration of the performance requirements in (functional) UML models.
- Specification of execution parameters used by performance analysis tools to evaluate and predict performance measures.
- Communicate performance measures generated by the performance tools.

The produced specification can be submitted to existing performance analysis tool for performance evaluation activities. The main interest in defining performance parameters in baseline UML models is to establish a direct link between performance analysis and baseline model improvement. Performance issues identified during the performance analysis step, such as bottlenecks, can be traced back directly to the baseline model, which can be revised and improved accordingly. This process can continue iteratively until a satisfactory baseline model fulfilling all the QoS requirements is obtained. In the sequel, we describe the components of the above-mentioned UML profile that deal specifically with performance modeling.

### A. Modeling Concepts

System analysis and design may proceed typically by identifying and implementing the different scenarios that characterize the operation of the system. Performance analysis activities can also be built around the same concept. Performance modeling in UML is carried around the concept of *scenario*. Performance relevant scenarios identified during the functional analysis step (see Figures 1) serve as building blocks for the UML performance modeling. During functional analysis, scenarios are represented using UML collaboration or activity diagrams. The performance modeling task simply consists of annotating these diagrams by specifying the values of relevant basic performance parameters. Key performance concepts in UML include the notions of performance context, scenarios, and scenarios steps. We define these concepts as follows.

**Performance Context:** The highest-level performance concept in UML is the notion of performance context. A performance context can be defined as a collection of scenarios, which are used to describe specific dynamic situations involving specified resources. For instance, a performance context can be defined for system usage under particular loads such as peak loads. Performance modeling starts by identifying performance contexts, and by associating scenarios with each context. A performance context is characterized by a set of resources, a set of scenarios, a set of workloads applied to the scenarios in this context, and some quality of service (QoS) requirements.

**Scenarios:** Scenarios represent externally visible response paths characterized by response times and throughputs. In UML, a scenario is defined within a performance context. QoS requirements are associated to scenarios. Each scenario is characterized by a workload, also called job class or user class, which is applied with some workload intensity. A workload can be either open (open workload) or closed (closed workload).

**Scenario Steps:** Scenarios are basic execution sequences during the lifetime of the system. Those execution steps are called scenario steps. A step

represents an increment in the execution of a scenario. A scenario step may correspond to an activity, an elementary operation, or a sub-scenario. The first step of a scenario is called the root step.

Scenario steps are executed on *host resources*. A host resource is defined for a scenario only if all its sub-steps execute on the same host. A step is characterized by its host execution (service) demand. A scenario step is characterized by a mean execution count (i.e., mean number of times it is repeated during execution) and the host execution demand (i.e. the service demand created by the step on its host device). Resource demands may also be specified for sub-steps of the scenario step, and for external resource operations (e.g., disk I/O, CPU, etc).

### *B. Notations and Definitions*

Scenarios are modeled in UML using collaboration or activity diagrams. Performance concepts can be captured using the same modeling paradigms. However, activity graphs are more appropriate for modeling complex hierarchical scenarios, which can be described by decomposing sub-activities into lower-level activity graphs. The UML performance profile defines a set of standard stereotypes that can be used to express performance concepts and annotate UML models accordingly. The stereotypes are the same regardless of whether we are using activity or collaboration diagrams. However, there are some modeling exceptions that are specific to activity diagrams.

Now, we present an overview of common stereotypes and briefly discuss the specifics for activity modeling. The annotation of a collaboration or activity diagram involves the specification of the performance context, the scenario, and scenario steps.

**Performance Context and Scenario:** A *performance context* is modeled as a stereotype <<*PAcontext*>> of a UML collaboration or activity diagram. The stereotype is placed on top of the diagram.

A *scenario* maps to an interaction. It is explicitly represented by its first step also called root step, to which corresponding performance characteristics are associated as a UML note. The note may include performance characteristics such as workload type (open or closed), workload intensity, population size, and think time (in case of closed workload).

**Steps and Performance Steps:** A *step* corresponds to an execution of some action. A step is defined by associating a step stereotype <<*PAstep*>> with an action execution model element or with a message causing the action execution. The root step may optionally be associated with a stereotype describing the workload class: <<*PAopenLoad*>> for an open workload or <<*PAclosedLoad*>> for a closed workload.

*Performance steps* can be defined for all execution steps in the scenario. However, in order to avoid clutters, it is recommended to define them explicitly only for a limited number of steps, which carry useful or significant performance information that can be specified along with the step. Performance information includes performance characteristics such as response time, throughput requirements, service demands, etc.

**Processing and Passive Resources:** *Processing resources* can be specified separately in a deployment diagram by using the <<*deploys*>> relationships, and by showing which processor resource runs specific classifier role or instance. Processing resources can be shown in a collaboration diagram in the rare case where each classifier role or instance executes on its own host. In that case, the classifier role or instance is associated with a stereotype <<*PAhost*>>.

*Passive resources* can be defined by associating a stereotype <<*PAresource*>> to classifier roles or instances representing them. Associating one or several *PAextOp* tagged values with the steps using those resources may also represent them.

### *C. Example*

To illustrate the above notions, let us consider the example of "a deposit scenario at an ATM cash machine".

Figure 4 shows an example of performance annotated *sequence diagram* specifying a deposit scenario at an ATM cash machine. The performance context is shown with the stereotype <<*PAcontext*>>. Scenarios and scenario steps are represented using UML tagged expressions. The meanings of these expressions will be provided later in this Section. The root step shows that a closed workload model is used for this scenario.

Figure 5 shows an example of performance annotated *deployment diagram* for the cash machine example. The diagram shows the allocation of the objects involved in the scenario to the different nodes of the hardware infrastructure. The relationship between an object and associated node is stereotyped using keyword <<*GRMdeploys*>>.

Figure 6 shows an *activity diagram* representation for a deposit scenario in the cash machine example. The swim lanes of the activity graph correspond to the resources or the object instances involved in the scenario. An activity graph can be used to model only one scenario per context. Sub-scenarios are represented by using sub-activities linked to an activity. Only the top most performance context can define a workload (e.g., <<*PAopenLoad*>> or <<*PAclosedLoad*>>).

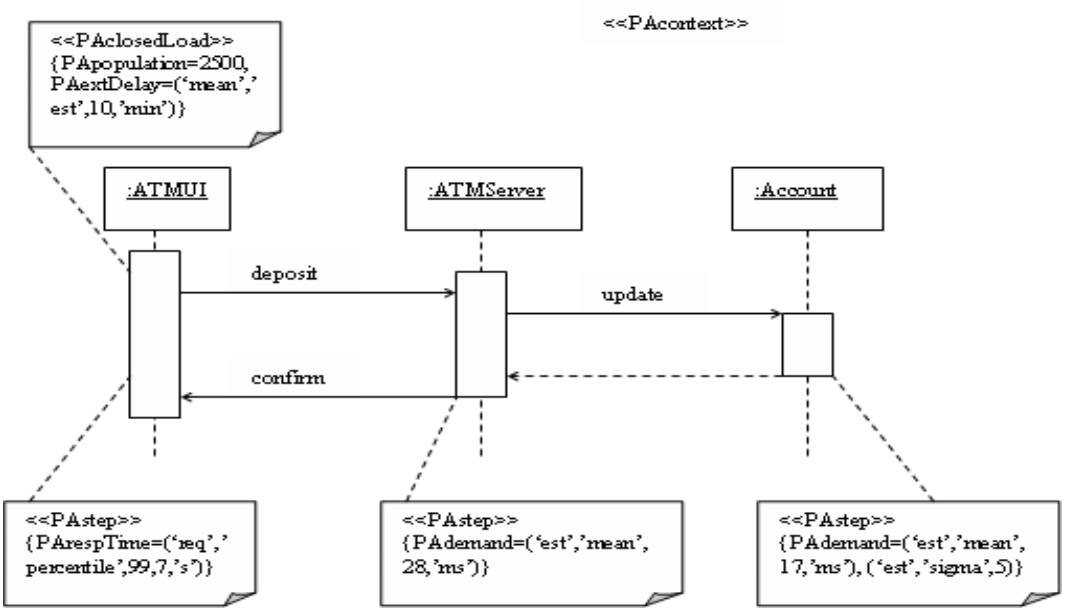



Figure 4. Performance annotation for a deposit scenario at a cash machine - Sequence diagram.

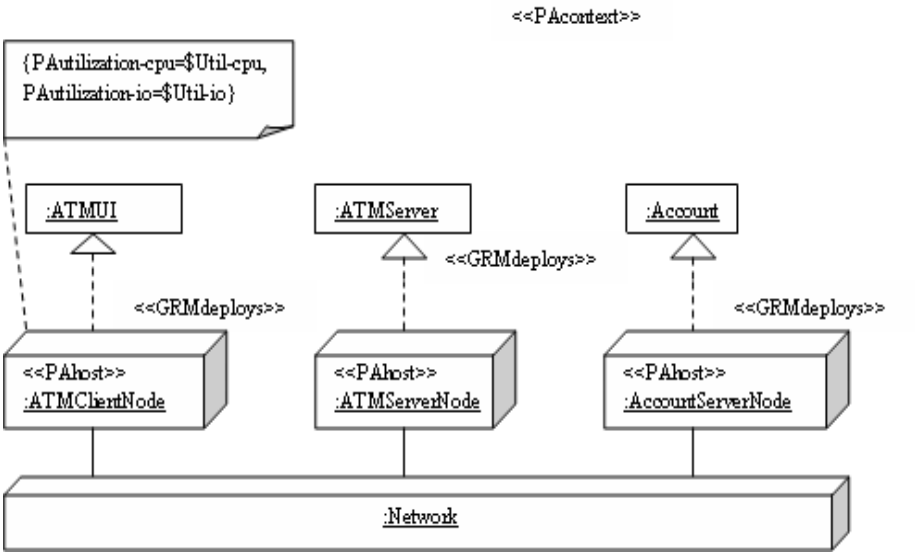



Figure 5. Performance annotation for a deposit scenario at a cash machine - Deployment diagram.

### *D. Performance Values*

For numerical results obtained from the performance analysis step to be meaningful, the meaning or interpretation of those results should also be provided. Numerical values are expressed as UML tagged values using the above mentioned performance stereotypes. These numerical results may correspond to some measures or some predictions, or some mean, average, maximum values, etc.

Performance values can be expressed in UML in a meaningful way by using the following template:

***<performance-value> ::= <source-modifier><type-modifier><time-value>***

A performance value can also be expressed in a UML note as an array in the following format:

***"("<source-modifier>"," <type-modifier>"," <time-value>")"***

where:

- <*source-modifier*> specifies whether the value is *required*, *assumed*, *predicted*, or *measured*.
  ***<source-modifier> ::= 'req'|'assm'|'pred'|'msr'***
- <*type-modifier*> specifies whether the value is *average*, *variance*, $k^{th}$ *moment*, *maximum*, $k^{th}$ *percentile*, or *probability distribution*.
  ***<type-modifier>::= 'mean'|'sigma'|'kth-mom,<Integer>'|'max'|'percentile," <real>|'dist"***
- <*time-value*> corresponds to the actual performance value.

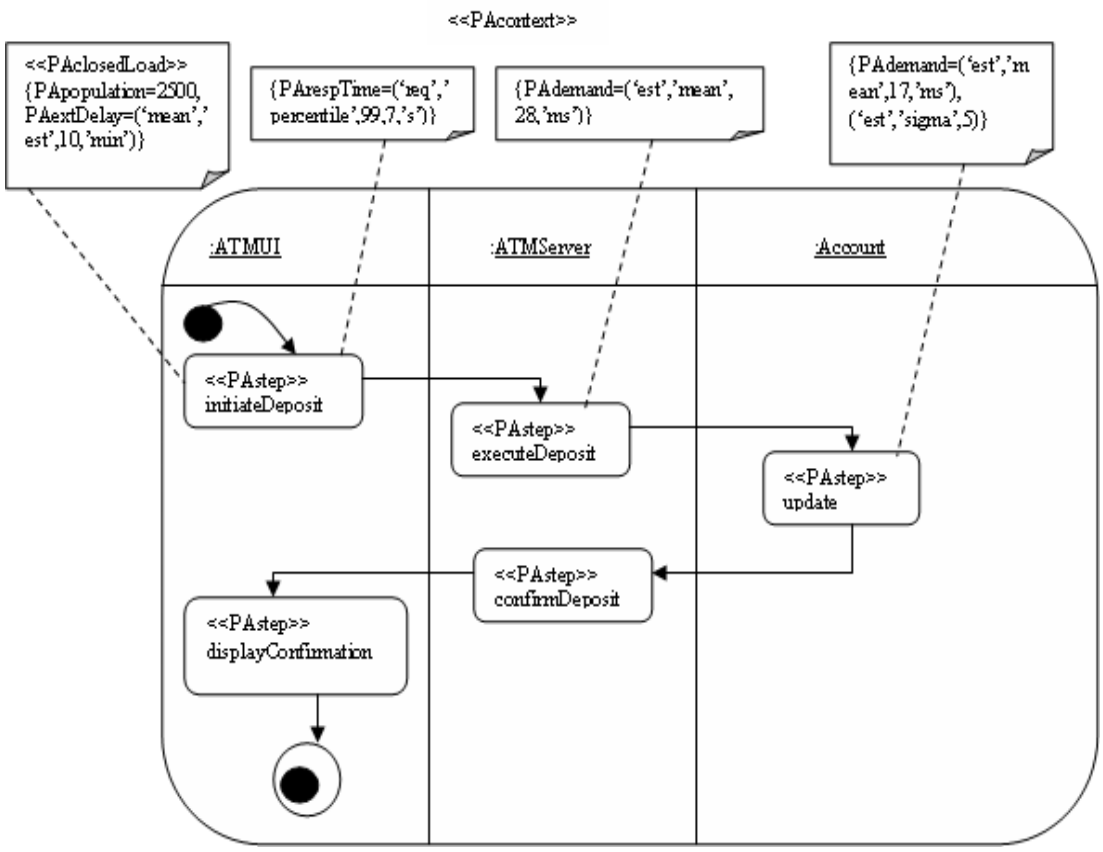



Figure 6. Performance annotation for a deposit scenario at a cash machine - Activity diagram.

Table 1: Examples of performance modeling stereotypes and tagged expressions [1].

| **Stereotypes** | **Tags** | **Notes** |
|---|---|---|
| *<<PAcontext>>* | | |
| *<<PAclosedLoad>>* | *PArespTime*<br>*PApopulation*<br>*PAextDelay* | *PApopulation* corresponds to the user population; *PAextDelay* corresponds to the think time |
| *<<PAopenLoad>>* | *PArespTime*<br>*PAoccurence* | *PAoccurence* corresponds to the workload intensity. |
| *<<PAhost>>* | *PAutilization*<br>*Pathroughput* | Expresses the utilization and throughput |
| *<<PAresource>>* | *PAutilization*<br>*PAcapacity*<br>*PArespTime*<br>*PAthroughput* | *PAcapacity* expresses the resource capacity |
| *<<PAstep>>* | *PAdemand*<br>*PArespTime*<br>*PAprob*<br>*PArep*<br>*PAdelay*<br>*PAextOp*<br>*Painterval* | *PAdemand* expresses the service demand; *PAprob* expresses the probability of execution for the step; *PArep* expresses the repetition of the Step; *PAextOp* identifies an external operation. |

For instance, the tagged value expression *{PArespTime=('req','max',(5,'s'))}* represents a maximum response time requirement of 5 seconds in a scenario or scenario step. The tagged value expression *{PAdemand=('msr','mean',(23,'ms'))}* represents a measured mean service demand of 23 milliseconds for a given scenario step. It should be noticed that the parenthesis around the time value can be removed without changing the semantic of the expression. Hence, *{PAdemand=('msr','mean',(23,'ms'))}* is equivalent to *{PAdemand=('msr','mean',23,'ms')}*.

The above expressions can be used to describe multiple or complex values for a single performance characteristic. For instance, the tagged value expression *{PApopulation=1000, PAextDelay=('est','assm',10,'min')}* represents a scenario in which the maximum number of concurrent users in the system is limited to 1000, and the average *think time* is assumed to be 10 minutes.

Table 1 gives a list of commonly used stereotypes and tagged expressions. The reader is referred to paper [8] for a more complete list of available tags.

## VI. CASE STUDY

To illustrate the concepts introduced above, we present in this section a case study based on the performance requirements of a reservation system for a hotel chain.

### *A. Performance Requirements*

A major hotel chain is replacing its old reservation system by a new Web-based system. Customers can search for rooms that match their preferences by filling a form, in which they specify the following information: bed size, smoking or non-smoking preference, start and end dates of stay, and city where the hotel is located.

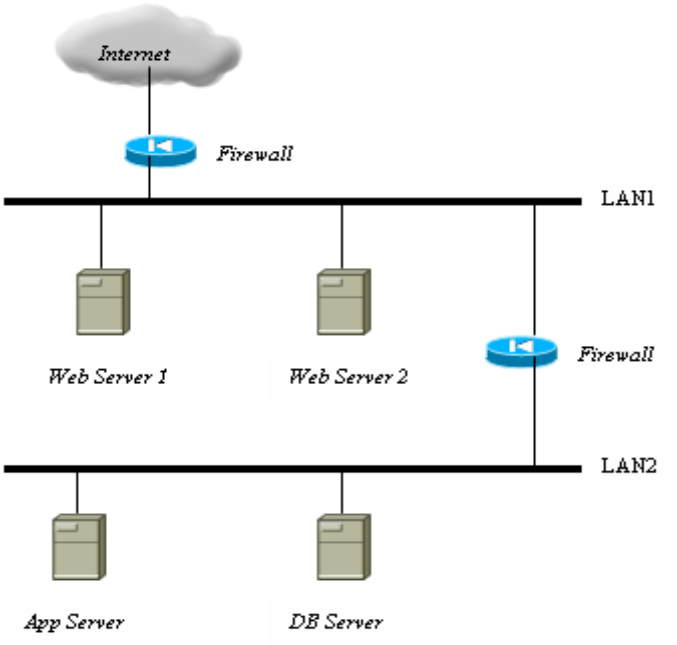



Figure 7: System architecture for the Hotel Reservation System.

After submitting the search request, the system checks the room availability and returns either a match and daily price or a non-match. In the case where a room is available, the customer can confirm the booking by providing his personal information (i.e., name, address, phone number, and email) and his credit card information (i.e., type, number, and expiration date) for billing purposes. After booking the room and storing the customer's information, the system returns a confirmation number to the customer that the customer will use for check-in. A customer can cancel a reservation by simply specifying the reservation number. In this case, he/she must receive a cancellation notice. In order to ensure customer satisfaction, the company requires that the average response times for checking room availability and confirming a reservation should always be less than 7 seconds. No particular restriction is put on the reservation cancellation.

After monitoring the system during a peak period of 4 hours, the following measurements were taken: (i) 120,000 search requests were processed by the system, (ii) 30% of these requests resulted in actual reservations, (iii) 5,000 reservation cancellation requests were processed.

Figure 7 represents the information technology infrastructure on which the system is deployed. It consists of two identical Web servers, one application server, and a powerful database server. The network architecture involves two 100 Mbps Ethernet LANs named LAN1 and LAN2. LAN1 is connected to the Internet via a firewall, and contains two Web servers. LAN2 is connected to LAN1 via a firewall and contains the database and application servers. All servers use Microsoft NT as operating systems. The web server runs IBM WebSphere as HTTP server. The application server runs IBM Application Server, and the database server runs IBM DB2 for database management. Using appropriate measurements tools (e.g., LAN analyzer, NT performance monitor, Web Server Logs, etc.) the service demands at network segments, the processors and disks of the Web servers, the applications servers, and the database servers are computed. Tables 2 and 3 provide these values.

Table 2: Service demands (in msec) for the Hotel Reservation System.

| | Search Room | | Confirm Reservation | | Cancel Reservation | |
|---|---|---|---|---|---|---|
| | CPU | I/O | CPU | I/O | CPU | I/O |
| Web servers | 8.5 | 12.5 | 8.1 | 12.5 | 3.1 | 8.5 |
| Application server | 18.0 | 15.1 | 13.4 | 12.5 | 8.9 | 17 |
| Database server | 19.7 | 88.8 | 17.5 | 85.2 | 35.5 | 98.9 |

Table 3. Network service demands (in msec) for the Hotel Reservation System.

| | Search Room | Confirm Reservation | Cancel Reservation |
|---|---|---|---|
| LAN1 | 0.81 | 0.22 | 0.11 |
| LAN2 | 0.42 | 0.80 | 0.17 |

### B. Functional Analysis

Architecture analysis for the above software system involves conducting a functional analysis for the system and identifying major use cases and scenarios. Quantitative performance characteristics such as workload intensity for corresponding scenarios can then be computed. Using the computed performance measures, an annotated software architecture model can be derived based on the UML profile for performance and schedulability.

A complete use case analysis of the system reveals the main functionalities carried by the system. The brief requirements provided in the previous section characterize mainly a *room reservation* use case, which involves three scenarios, namely *room search, reservation confirmation* and *reservation cancellation*. In order to illustrate the introduced concepts, we will analyze the two first scenarios. In the UML notation, scenarios are modeled using either interaction or activity diagrams. In order to illustrate both approaches, we describe the room search scenario using a sequence diagram and the reservation confirmation using an activity diagram. This allows us to illustrate how performance annotations can be carried in each of these notations.

The sequence diagram characterizing the room search scenario is shown in Figure 8. Using a Web browser, the customer first fills a form by specifying his/her preferences and then submits a search request. The Web server receives and forwards the search request to the application server, which delegates the search to a *Reservation Manager* object. The Reservation Manager then checks for a room availability at an appropriate hotel by searching the database.

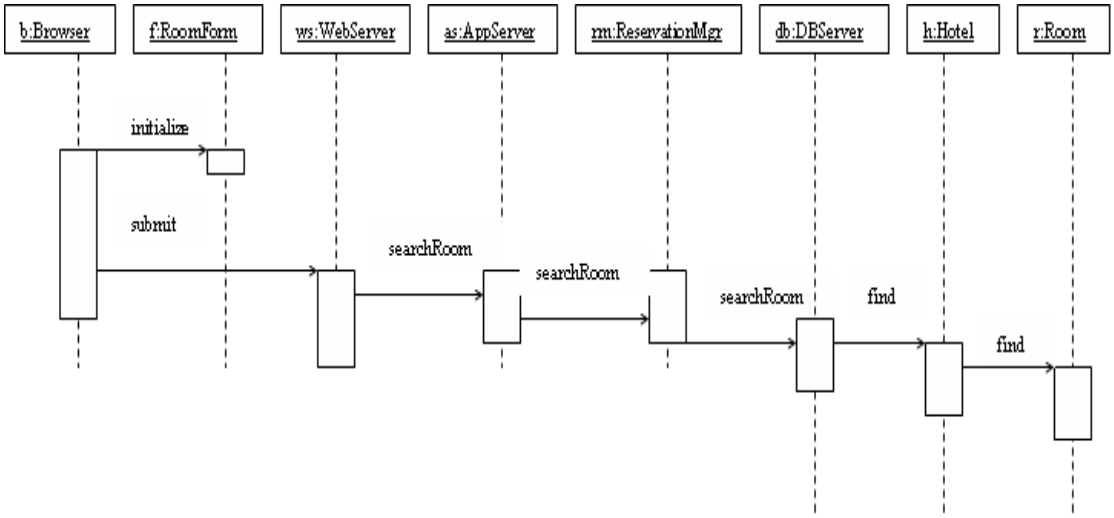



Figure 8. Sequence diagram of the Reservation System - Room search scenario.

Figure 9 depicts a hierarchical *activity diagram* representation of the Reservation *Confirmation* scenario (here, only one representation is needed, either the sequence diagram or the activity diagram).

A request to book one of the rooms returned after the room search is submitted by the customer through the Web browser. A customer information form is then created. After filling and submitting the customer form, the Web server transmits the booking request to the *Reservation Manager*, which processes and returns the result to the customer. The result can be either a successful booking with a confirmation number or an unsuccessful booking.

The sub-activity diagram shown in Figure 10 further details the processing performed by the Reservation Manager. This diagram is a refinement of the sub-activity entitled "*handle booking*" in Figure 9.

In the sub-activity diagram shown in Figure 10, the reservation manager initiates the booking process by creating or updating the customer's personal information collected from the information form initially submitted with the request. Then, the effective booking takes place by finding the requested hotel with an available matching room. The room is booked and a reservation record with a number is issued and a confirmation is sent back to the customer. In the case where no room is available, a message is sent back to the customer.

Figure 11 shows the *deployment* of the objects involved in the reservation system across the underlying hardware and networking infrastructure. The Web browser is deployed on the client's workstation. The Web server machine deploys the Web server program, and the customer and room forms (objects), which may be implemented, for instance, using JSP or ASP technologies. The application server program and the Reservation Manager object are deployed on the application server node. The Database Manager is deployed on the database server node. The database server node deploys also the data tables, which are not represented in Figure 11 to avoid clutter. Instead, data tables are represented in Figure 12.

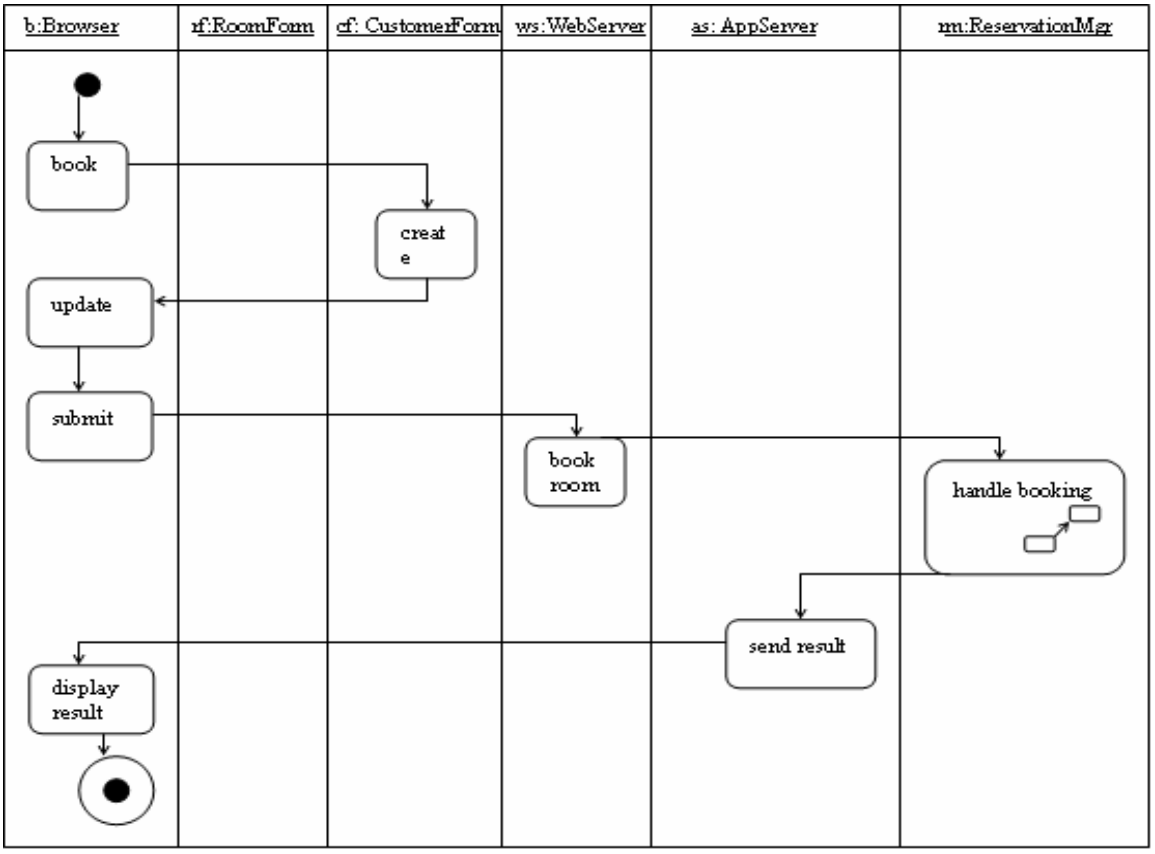



Figure 9: Activity diagram for the Reservation System - Reservation confirmation scenario.

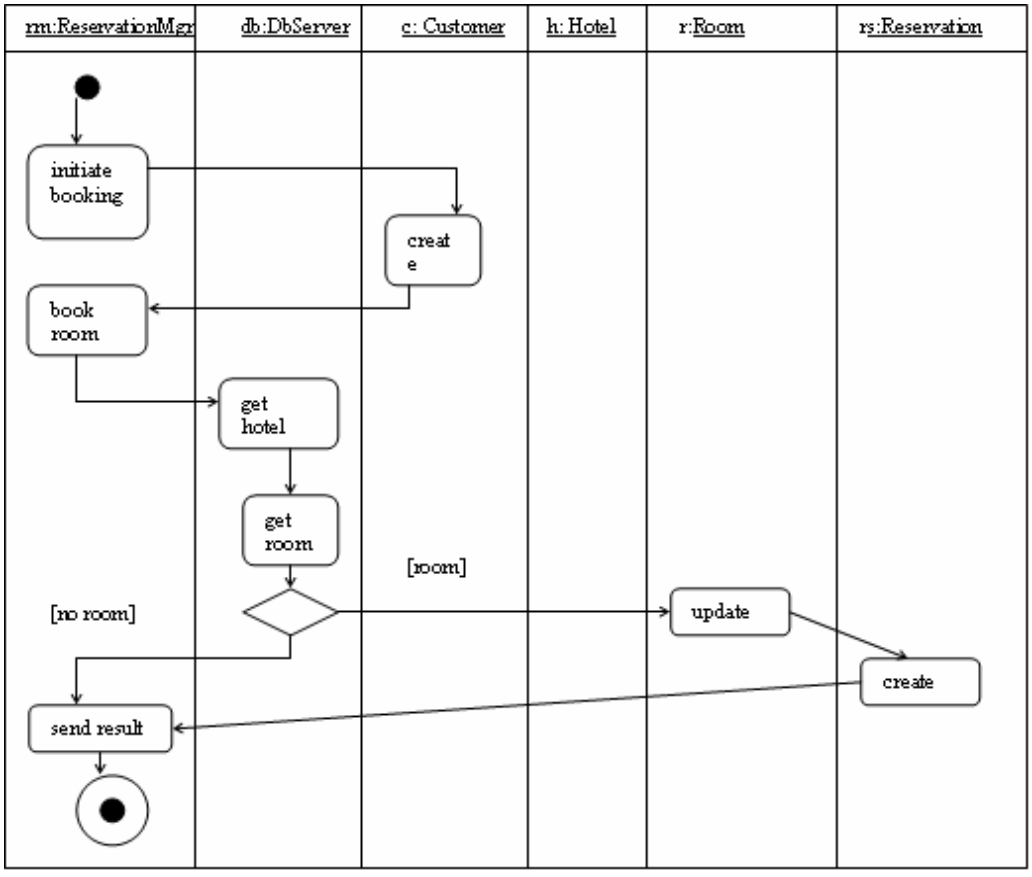



Figure 10: Activity diagram for the Reservation System -"Handle Booking" sub-activity.

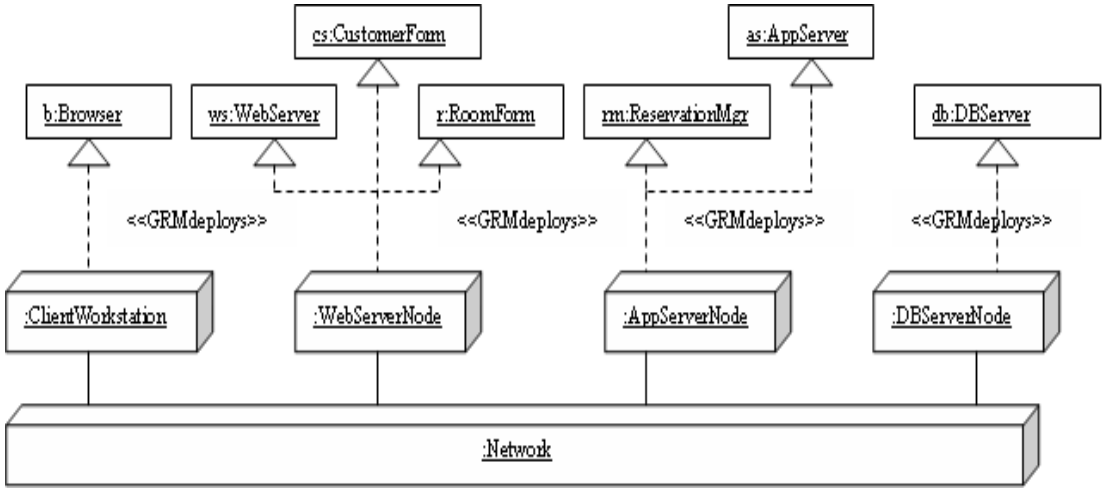



Figure 11. Deployment of the logical elements on the hardware infrastructure for the Reservation System.

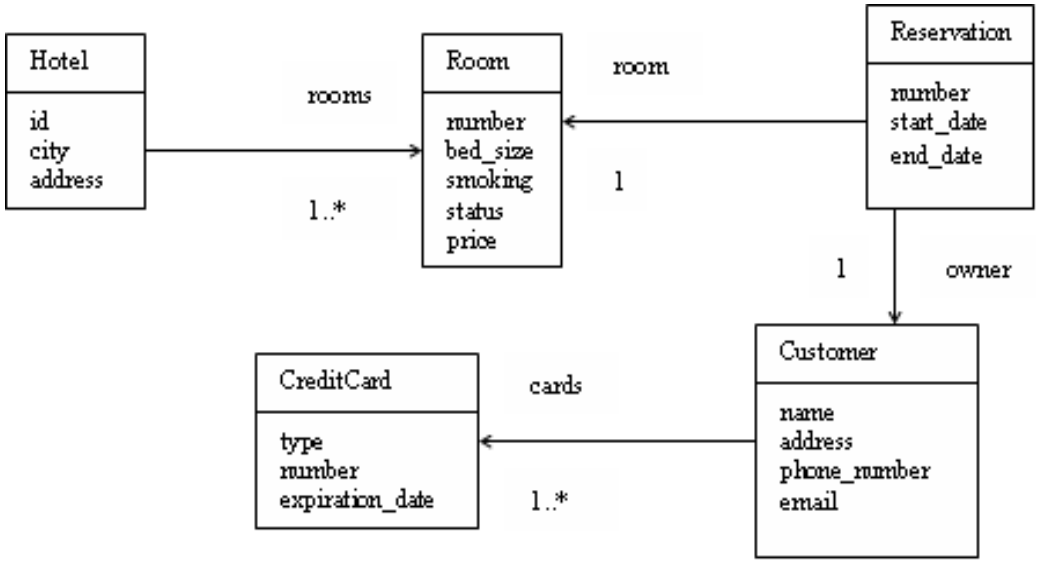



Figure 12: Data Model for the Reservation System.

Figure 12 describes the data model, which is quite simple. A hotel is supposed to contain at least one room. A reservation is linked to exactly one room and is issued in the name of a single customer, even though the room may host several persons. A customer is required to provide at least one credit card. A hotel is described by specifying a unique identifier, the city where it is located (we assume that the hotel is named after the city), its address, and the set of available rooms. A room is described by specifying a unique number, the bed size (double, single, etc), whether the room is for smoking or non-smoking customer, the room's price and status (available or non-available). A reservation is described by providing a unique confirmation number, the start date, the end date and the corresponding room number. The customer data include its name, address, phone number, email and credit card information. The credit card information includes the type (e.g., visa, master card, etc), the number, and the expiration date.

### *C. Performance Measures*

Measurements taken during a peak period of 4 hours showed that 120,000 search requests were processed by the system, and 30% of these requests resulted in actual reservations. Also, 5000 reservation cancellation requests were processed. Thus, the workload arrival rates for these transactions can be calculated as follows:

$$\lambda = \frac{number\ \ of\ \ incoming\ \ requests}{monitoring\ \ \ time}$$

$$\lambda_{search} = \frac{120000}{4 \times 3600} = 8.33 \quad req/sec$$

$$\lambda_{confirm} = \frac{120000 \times 0.3}{4 \times 3600} = 2.5 \quad req/sec$$

$$\lambda_{cancel} = \frac{5000}{4 \times 3600} = 0.347 \quad req/sec$$

*D. Annotated UML Model*

Performance requirements and measures for the hotel reservation software system are shown in annotated models using standard UML stereotypes and tagged expressions. We assume open workload in this case.

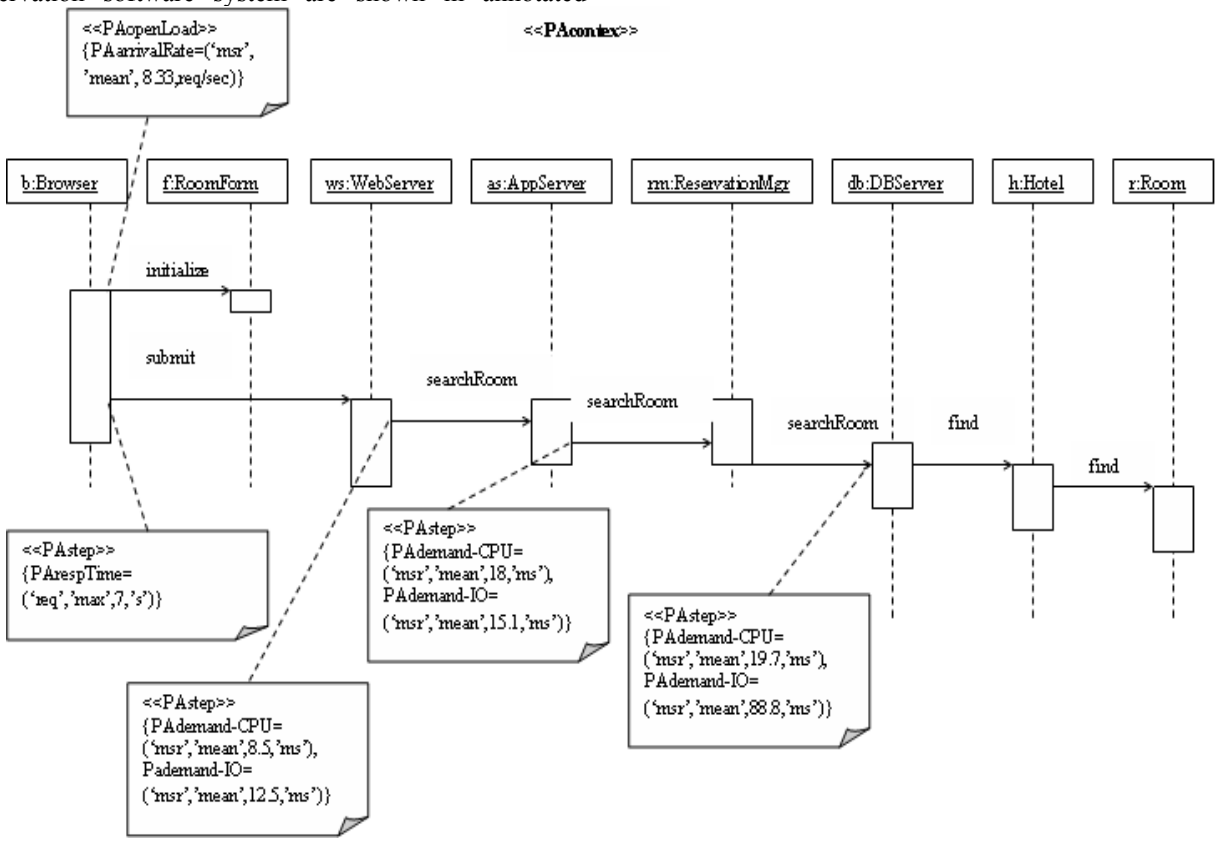



Figure 13. Sequence diagram of the Reservation System – Room search scenario.

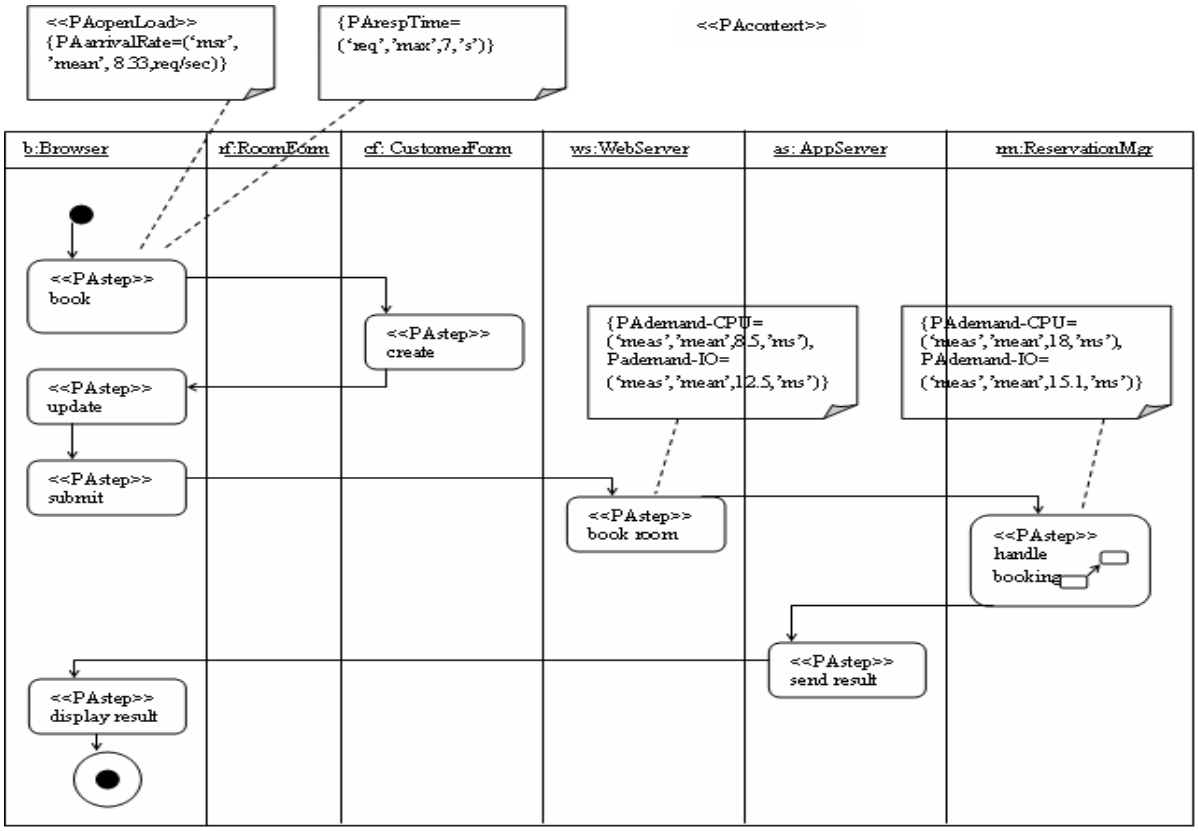



Figure 14. Annotated activity diagram for the Reservation System- Reservation confirmation scenario.

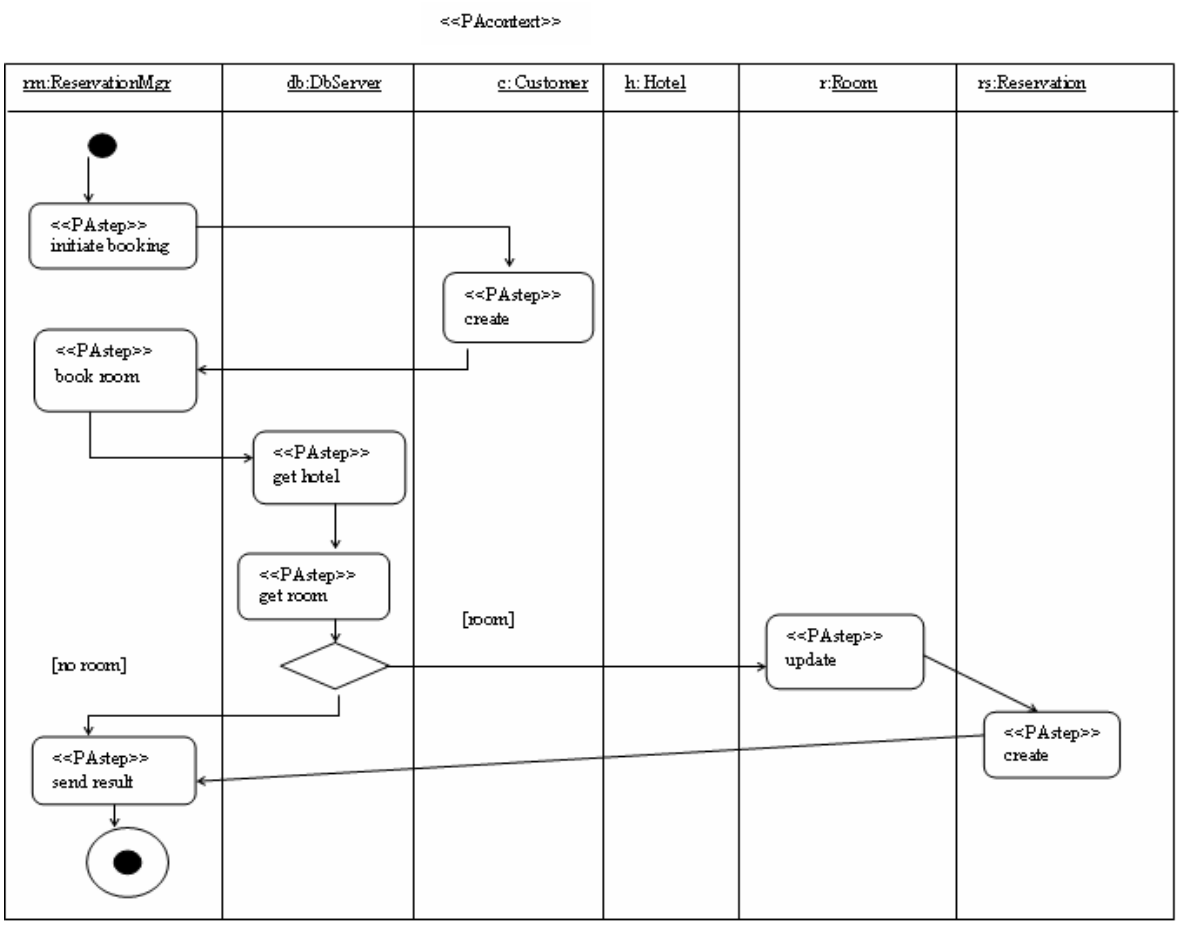



Figure 15. Annotated activity diagram for the Reservation System-"Handle Booking" sub-activity

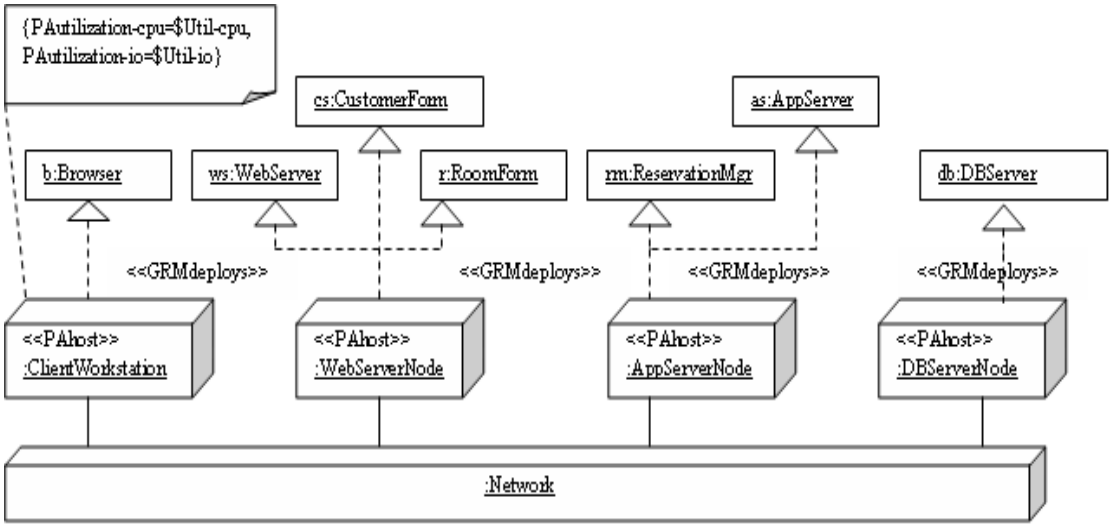



Figure 16: Annotated deployment diagram for the Reservation System.

The annotated diagram for the *sequence diagram* is shown in Figure 13. Similar diagrams for the *activity graphs* are shown in Figures 14 and 15.

For instance, in Figure 13, the first note (see upper corner left side of Figure 13) indicates that the scenario is represented as an open workload model, in which the mean workload arrival rate is measured as *8.33 req/sec*. The symbol "*msr*" indicates that the corresponding values are also measured.

The second note specifies that response time for search transactions should be no more than 7 seconds. Service demands are specified in appropriate steps as well. For instance, the second note specifies mean service demands of *8.5 msec* at the CPU and *12.5 msec* at the disk in the

Web Server node. An annotated diagram is also given in Figure 16 for the *deployment diagram*. The annotation includes the nodes' relevant performance characteristics such as utilization, scheduling policy, etc. For instance, utilizations are computed using some performance models and tools.

## VII. PERFORMANCE TESTING

Performance testing can be conducted by evaluating the system against some published benchmarks or by using testing environments. Benchmarks provide a synthetic environment that mimics the real environment in which the system may be deployed [23, 24]. On the other hand, testing tools are used to simulate realistic workloads using load generators and usage scenarios, by generating scripts. Executing the scripts allows one to test the system response under specific workloads.

Performance testing consists of evaluating the system performance using realistic workloads and under particular usage scenarios. Possible workloads include steady-state workloads and peak-usage workload. According to the load used, three kinds of performance testing may be conducted:

1. Load testing: to evaluate the system under specific workload. The system is submitted to various workloads representative of real or projected usage scenarios.
2. Stress testing: which consists of testing the system under extreme workloads, heavier than what would be expected (i.e. worst-case scenarios).
3. Spike testing: which consists of testing the system by simulating the potential load spikes (e.g.: heavy load for a short time length).

Since most realistic systems are complex, it is more convenient and practical to automate the testing process. There is a variety of performance testing tools available on the market. Examples of tools include Rational Performance Tester [24] and Mercury Load Runner [25]. A typical performance tester consists of two basic elements: *controller* and *virtual users* (VU). VUs simulate the workload to which the system under test (SUT) is submitted to. As such, they play the role of real users. The controller, as the name indicates, controls the load generation process. A performance testing process involves several activities as illustrated in Figure 17.

- ***Defining the test objectives:*** At the beginning of the test process, the goals of the testing must be clearly established. For instance, testing may be conducted to identify the maximum throughput that the system can achieve or to determine the number of concurrent users that the system can support under peak load with acceptable response time.
- ***Analyzing the application workload and environment:*** This step consists of establishing a clear understanding of the workload under which the system is submitted to, as well as the services it provides, and the hardware and software environment in which it is running.

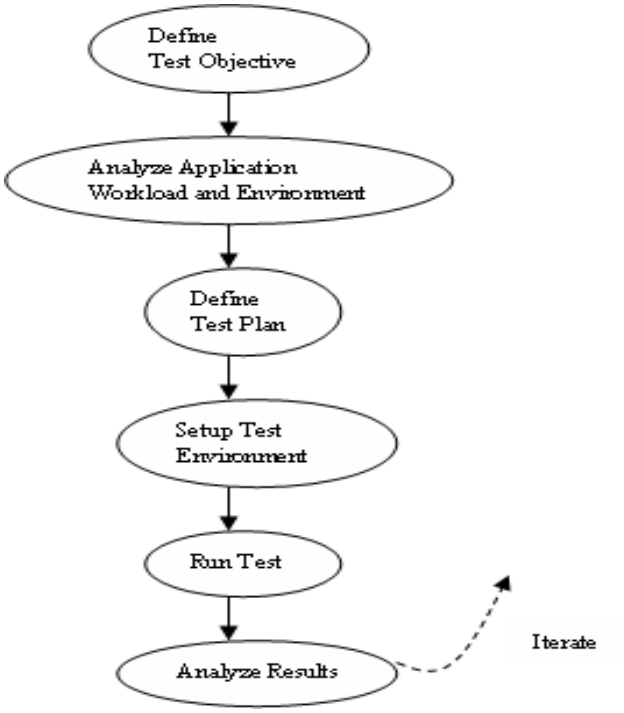



Figure 17: Major Phases of the Performance Testing Process.

- ***Defining the test plan:*** In this step, the test plan typically outlines the functions and services that are targeted by the testing process, the workload and usage scenarios, and the quality of service (QoS) requirements that should be met. The test plan must also include a schedule specifying in which order the services and functions will be tested, as well as the list of deliverables.

- ***Setting up the test environment:*** This step involves initializing the persistent data, for instance, by populating the database and generating test scripts based on the different scenarios.

- ***Running the test:*** This step consists of executing the developed test scripts. The same test must be executed several times in order to obtain statistically significant results.

- ***Analyzing the results:*** At this step, the test results are collected and analyzed. For instance, by identifying bottlenecks and by taking proper actions to improve the performance of the system when necessary. This may lead to several iterations of some major phases of the testing process.

### *A. Example*

We consider the same Case Study introduced in Section 5. Assuming that the hotel reservation system can support up to a maximum of $N = 1500$ concurrent users, we can determine, as explained in the following, the

number $N_{vu}$ of concurrent virtual users (VUs) needed to stress test *search* transactions.

The maximum response time requirements set for *search* transactions is $R_{max}$= *7 sec*. During stress testing, a virtual user (VU) responds directly when it receives a request; thus the think time is $Z_{vu}$ = *0 sec*. The goal of using virtual testers is to simulate the real environment so that the throughput and response time remain the same in both the real system and test environments. Thus, by applying the response time law in both cases, we obtain the following:

$$X_{search} = \frac{N}{R_{search} + Z} = \frac{N_{vu}}{R_{search} + Z_{vu}}$$

$$\Rightarrow (N - N_{vu})R_{search} = (N_{vu}Z - NZ_{vu})$$

$$\Rightarrow R_{search} = \frac{(N_{vu}Z - NZ_{vu})}{(N - N_{vu})} \leq R_{\max}$$

$$\Rightarrow (N_{vu}Z - NZ_{vu}) \leq (N - N_{vu})R_{\max}$$

$$\Rightarrow N_{vu} \leq \frac{NR_{\max} + NZ_{vu}}{Z + R_{\max}}$$

$$= \frac{1500 \times 7 + 1500 \times 0}{45 + 10} = 190.91 \approx 191$$

Therefore, the required number of VUs to stress test *search* transactions is $N_{vu}$ = *191*. The project manager or the test engineer should ensure that the licenses for the targeted performance testing tool can accommodate at least 191 concurrent VUs.

Based on the maximum number of VUs, several runs of the test are conducted by increasing the number of VUs. During each run, performance measures such as response times and utilizations are collected. Repeating the test over several iterations allows one to obtain statistically significant measures. Statistically stable performance measures can be computed by averaging the measurement outputs over the different runs and by computing the corresponding confidence intervals. At the end of the test, the collected data can be analyzed in order to identify potential bottlenecks. The analysis consists of studying trends versus performance requirements, correlating and interpreting performance results. In the considered example, a typical trend is that increasing the number of VUs leads to an increase in response time. In principle, the utilization should increase as well. If the resource utilization levels off while the number of VUs is increasing, this means that the system has reached a bottleneck. Further analysis would then be required in order to identify this bottleneck.

## VIII. CONCLUSION

Identifying and addressing quality concerns as early as possible in the software development cycle is a well-established and accepted wisdom in the software community. Software performance is one most of the essential quality criteria for most of today business applications. In this paper, we have introduced a SPE process that assists in addressing performance concerns throughout the software development lifecycle. We have shown that by using the recently proposed UML Profile for Schedulability, Performance, and Time, our proposed model-driven SPE process can effectively deal with building annotated UML models for performance, which can thereby be used for the performance predictions of software systems. A case study illustrating our approach has been presented, along with the performance annotation steps. As future work, we intend to explore the implementation of the MDSPE process and its applications to case studies from the industry from a performance study viewpoint.

## REFERENCES

[1] I. Traore, I. Woungang, A. A. El Sayed Ahmed, and M. S. Obaidat, " UML-Based Performance Modeling of Distributed Software Systems", Proc. of IEEE International Symposium on Performance Evaluation of Computer Telecommunication Systems (SPECTS 2010), July 11-13, Ottawa, Canada, pp. 119-126, 2010.

[2] C. U. Smith, "Performance Engineering of Software Systems", Addison-Wesley, 1990.

[3] S. Balsamo, M. Marzolla, "A simulation-based approach to software performance modeling", In Proc. of ESEC/FSE 2003 (Paola Inverardi, Editor), Helsinki, FI, Sep. 1-5, 2003. ACM Press.

[4] D. E. Geetha, R. M. Reddy, T. V. S. Kumar, K. R. Kanth, "Performance Modeling and Evaluation of e-commerce Systems Using UML 2.0", In Proc. of the 8th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD2007), Qingdao, China, July 30-Aug. 1st, 2007.

[5] S. Balsamo, A. Di Marco, P. Inverardi, M. Simeoni , "Model-based performance prediction in software development: a survey", In IEEE Trans. on Software Engineering, May, vol. 30, Issue 5, pp. 295-310, 2004.

[6] E. Dimitrov, A. Schmietendorf, R. Dumke, "UML-Based Performance Engineering Possibilities and Techniques", IEEE Software, vol. 19, no. 1, pp. 74-83, Jan-Feb, 2002.

[7] R. Mirandola and V. Cortellessa, "UML Based Performance Modeling of Distributed Systems", In LNCS, Springer Berlin/Heidelberg, vol. 1939/2000, pp. 178-193, 2000.

[8] OMG, "UML Profile for Schedulability, Performance, and Time", Sept. 2003, Online access at: http://www.omg.org (Last visited Feb. 25, 2010).

[9] I. Traore, I. Woungang, A. A. E. Sayed Ahmed, and M. S. Obaidat, "Performance Analysis of Distributed Software Systems: A Model-Driven Approach". Proc. of IEEE Intl. Symposium on Performance Evaluation of Computer Telecommunication Systems (SPECTS 2010), July 11-13, Ottawa, Canada, pp. 111-118, 2010,

[10] Object Management Group, http:/www.omg.org, OMG Unified Modeling Language specification, March 2003, version 1.5 (Last visited Nov. 15th, 2010)
[11] J. A. Street, R. G. Pettit IV, "Lessons Learned Applying Performance Modeling and Analysis Techniques", Proc. of the 9th IEEE Intern. Symposium on Object and Component-Oriented real-Time Distributed Computing (ISORC'06), Gyeongju, Korea, pp. 208 -214, Apr. 2006.
[12] V. Garousi, "UML model-driven detection of performance bottlenecks in Concurrent Real-Time Software", In the Proc. of IEEE International Symposium on Performance Evaluation of Computer Telecommunication Systems (SPECTS 2010), July 11-13, Ottawa, Canada, pp. 317-324, 2010.
[13] V. Cortellessa, A. Di Marco, R. Eramo, A. Pierantonio, C. Trubiani, "Digging into UML models to remove performance antipatterns", In Proc. of the 2010 ICSE Workshop on Quantitative Stochastic Models in the Verification and Design of Software Systems (Quovadis'10), Cape Town, South Africa, Mar. 2010.
[14] M Gogolla, M. Kuhlmann, F. Buttner, "A Benchmark for OCL Engine Accuracy, Determinateness, and Efficiency", In Model Driven Engineering Languages and Systems, LNCS, 2008, Vol. 5301/2008, pp. 446-459, 2008.
[15] A. AL Abdullatif and R.J. Pooley, "UML-JMT: A Tool for Evaluating Performance Requirements", In Proc. of the 17th IEEE Intern. Conference and Workshops on the Engineering of Computer-Based Systems, Oxford, UK., March 22-26, pp. 215 - 225, 2010.
[16] S. Distefano, A. Puliafito, M. Scarpa, "Implementation of the Software Performance Engineering Development Process", Journal of Software, Vol. 5, No. 8, pp. 872- 882, Aug. 2010.
[17] Y. Zhang, T. Huang, J Wei, "Declarative Performance Modeling for Component-Based System using UML Profile for Schedulability, Performance and Time", In Proc. of the 4th Intl. Conference on Software Engineering and Formal Methods (SEFM '06), Pune, India, Sept. 11-15, 2006.
[18] H. Koziolek, " Performance Evaluation of Component Based Software Systems: A Survey", In Performance Evaluation, vol. 67, Issue 8, pp. 634-658, Aug. 2010.
[19] S. Bernardi, J. Merseguer, "Performance evaluation of UML design with Stochastic Well-formed Nets", Journal of Systems and Software 80 (11), pp. 1843-1865, 2007.
[20] V. Garousi, "Experience and Challenges with UML-Driven Performance Engineering of a Distributed Real-Time System", In Information and Software Technology 52, Elsevier, pp. 625–640, 2010.
[21] M. Marzolla, "Simulation-Based Performance Modeling of UML Software Architectures", Ph.D. Thesis: TD-2004-1, Dipartimento di Informatica, Universita Ca' Foscari di Venezia, Italy, 2004.
[22] S. Balsano and M. Marzolla, "Performance Evaluation of UML Software Architectures with Multiclass Queueing Network Models", Proc. of the 5th intl. Workshop On Software and performance (WOSP'05), Palma, Illes Balears, Spain, July 12-14, 2005.
[23] System Performance Evaluation Corporation, http://www.spec.org (Last visited Nov. 23, 2010).
[24] Transaction Processing Performance Council, http://www.tpc.org (Last visited Nov. 23, 2010).
[25] Mercury LoadRunner, http://www.mercury.com (Last visited Nov. 23, 2010).

# User-Centric Mobility for Multimedia Communication: Experience and Evaluation from a Live Demo

Raffaele Bolla, Riccardo Rapuzzi
Department of Communications, Computer and System Sciences (DIST), University of Genoa, Italy
Email: {raffaele.bolla, riccardo.rapuzzi}@unige.it

Matteo Repetto
Italian National Inter-University Consortium for Telecommunications (CNIT), Parma, Italy
Email: matteo.repetto@cnit.it

***Abstract*— Nowadays people claim and expect pervasive communications, with continuous and seamless media access; that requires new communication paradigms beyond the legacy network- and device-centric approaches, and leads to the user-centric concept.**

**Mobility is a key issue in pervasive communications, and session migration is the most related aspect with the user-centric vision; however, despite of the fact that the user should be at the center of the system, no evidence of user involvement in the design and evaluation phase can be found in the literature for this topic.**

**In this paper we describe the user evaluation we carried out by a live demo open to a large heterogeneity of potential users at a national science exhibition. Our purpose was twofold: on the one hand, to evaluate users' feeling with our user-centric networking mobility framework based on the concept of Personal Address and, on the other hand, to figure out general indications for the whole research community about user's expectations and requirements for session migration.**

***Index Terms*— Session migration, integrated mobility, user-centric communication, personal address, user evaluation.**

## I. Introduction

Modern communication paradigms have gone well beyond the raw packet transport service for which the Internet was originally designed. The latter has radically evolved since its beginning; this change concerns the nature and composition of the traffic but also – and mainly – the way the Internet is perceived by users: today most of the people expect to "interact" with the Internet, not only to retrieve HTML pages or to transfer files.

This evolution has had a deep impact on the masses, with significant social implications, yielding new paradigms of communication in which users are at the center of the network. Nowadays people expect communications to happen in a way that is more tailored to humans instead of machines; the trend is to move from legacy device- and network-centric approaches to the user-centric paradigm. Roughly speaking, technology should go towards users and adapt itself to them, unlikely what has been happening until recently.

One of the key aspect in this evolution is transparent and seamless access to network content and services by users, regardless of their location and the terminal device(s) they are using. That rises new expectations about an effective and flexible mobility support in the current Internet.

There has been an ever increasing interest in bringing mobility into the Internet architecture during the last 20 years, which somehow resembles the evolution discussed so far: initially, the effort was mainly devoted to make devices mobile (terminal handover), whilst in the past decade the focus was on session and service mobility (session migration and service portability). The latter are essential in order to build user-centric pervasive communication environments; however, despite of the large numbers of algorithms and protocols for terminal handover, few proposals are available for session migration.

The main performance issue of mobility frameworks concerns the "seamless" property, which demands for fast and timely execution of the migration procedure. Usually, an upper bound on the communication gap during the handover procedure can be easily derived depending on the application; for example, it is well known that a few hundreds milliseconds of voice conversation can be lost without significantly affecting the understanding of the whole dialog, while for data traffic few seconds of delay may be tolerated before muddling the TCP congestion control up.

The above consideration is certainly true for terminal handover, but things are more complicated in case of session migration: the quality of service perceived by the user depends not only on the media disruption but also

This paper is based on "User-Centric Mobility for Multimedia Communication: Experience and Evaluation from a live demo" by R. Bolla, R. Rapuzzi and M. Repetto, which appeared in the Proceedings of the 2010 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS 2010), Ottawa, Canada, July 11-14.


doi:10.4304/jnw.7.1.21-36

on the current user behavior and context. Indeed, when an active session is migrated from one terminal to another one, depending on the relative position of the user and the two devices the user may have to turn his head, to pick up the new device or to move towards the new terminal (if the two devices are not close each other); in all cases, his behavior may hide in part or completely the transition delay.

The main outcome from this reasoning is that session migration should be evaluated both quantitatively (in terms of migration delay) and qualitatively (in terms of user satisfaction). Unfortunately, the latter aspect is very often (if not always) neglected. Nevertheless, user evaluation is crucial while developing user-centric systems, where the interaction between the user and the device is as important as technological issues. Further, quantitative measurements are almost impossible with automatic session migration: it is not possible to say when session migration should start and the time by which it should be completed, as that would require to know where the user is looking at and whether the devices could be seen at the same time.

In this paper we discuss our experience with the evaluation of a user-centric mobility framework. Such framework is built around the concept of *Personal Address* (PA) [1], i.e., a network address logically associated to the user rather than to a physical device; it accounts for personal mobility, terminal handover and session migration.

We carried out user evaluation in a live demo at a national science exhibition, by asking visitors to compile a short questionnaire after they had tried the demo themselves. We did not limit to gather feedback about our framework, but we also asked more generic questions that was used to figure out *what* users expect from a pervasive system and *how* they are willing to interact with it. We think our work may be a reference for whoever would deal with mobility issues in user-centric systems.

The paper is organized as follows. Section II provides a brief overview about mobility, in particular for what concerns session migration. Section III explains the PA concept and the user-centric framework for mobility, while Section IV discusses the current architectural solution. Section V describes how the mobility framework was used to build a Video and Voice over IP application (VVoIP) for user evaluation and Section VI describes the set up of the live demo. User evaluation is discussed thoughtfully in Section VII and final remarks are given in Section VIII. Finally, our conclusions are derived in Section IX.

## II. Related work

Applications establish communication sessions over the network and maintain a status and a context for each of them. The migration of an on-going session requires the transfer of its status/context to another instance running on a different host, therefore this kind of mobility always requires support at the application layer. Some applications have very complex status information and so migration is not a trivial task; thus, this topic has not been considered in the literature as much as terminal mobility [2], [3].

Session migration may be implemented by means of specific middleware, working for all applications. The Adaptive terminal Middleware (AMID) [4] provides an architecture for network monitoring, device discovery and session migration. The middleware scans for devices on each network and makes decisions on the basis of resource availability. The migration architecture requires the user to have a mobile device with him: this device is supposed to follow the user during his movements, it maintains the current context and controls the session for the whole duration; it can either redirect media from the source to the new local device or act as a proxy to deliver media to the current local device, without any update to the source. This last scheme is useful when the new local device is not addressable from the source, e.g., it lies on a local network without a direct Internet connection.

The most recent paradigms for session migration aim at maintaining the same IP address for the whole session duration, VNAT [5] and DIP [6] first introduced this concept. Both of them envision a sort of “loan” of the original IP address coupled with a MIPv6 architecture; the application and transport protocols at the new terminal use the borrowed address, which is translated into a usable real address through specific mechanisms.

Virtual Network Address Translation (VNAT) with MIPv6 [5] relies on the network address translation (NAT) function to map a virtual address (the first used for the session) into a real one (that used at the current host) at both the local and the remote peer.

Delegated IP (DIP) [6] exploits the Return Routability of MIPv6 to redirect packets towards a target node. The original host “delegates” the use of its Home Address to the target node; DIP provides a DIP IP Adaptation Layer (DIAL) and a DIP Transport Adaptation Layer (DTAL) at the target node to receive packets addressed to a third-party from the MIPv6 infrastructure and to deliver them to a local socket.

VNAT and DIP share the risks of using the same address at the transport layer of two different hosts. For example, in VNAT the local host does not keep track of session migrated elsewhere; port number conflicts are possible at the corresponding host whether this latter tried to set up another session before the previous were closed [7] (this is quite likely, as each application usually uses the same port numbers). On the other hand, in DIP the corresponding host cannot reach the local host for the whole duration of the migration because of a route update towards the target node (see [6] for protocol details).

Other issues about VNAT and DIP concern their applicability limited to IPv6 networks. Moreover, VNAT requires its framework to be present in the corresponding node as well and does not manage the simultaneous movement of the two endpoints. On the other hand, DIP only permits one single migration per session; moreover DIP violates the usual division in layers, as it requires the network entities to inspect packets for higher-layer

TABLE I.
CLASSIFICATION OF PROTOCOLS FOR SESSION MIGRATION

| *Protocol* | *Layer* | *Type* | *Applicability* | *Notes* |
|---|---|---|---|---|
| AMID | Application | Middleware | Generic | Deals with network monitoring, device discovery and session migration. The user must keep a device with him. |
| VNAT/MIPv6 | Network/Application | Address preserving | | Two (or more) devices use the same address; port conflicts are possible. |
| DIP | | | | Two (or more) devices use the same address; port conflicts are possible. Only one migration is possible. |
| PA | | | | Cross-layer approach. The address is not shared among devices. Requires at least one address per user. |
| 3PCC | Application | SIP extension | SIP | The original device keeps the control of the session. Session may be adapted to the capability of the new terminal. |
| SH | | | | Session may be adapted to the capability of the new terminal. |

information (i.e., source and destination ports), and this results in increasing the complexity of the solution.

The Personal Address (PA) [8] enriches the above protocols by bringing the user-centric principles and paradigms into the network as well, pursuing an approach where users have the leading role. Users are the session endpoints whilst devices only act as the physical terminals. To this aim, users are assigned network addresses, which are used by hosts and applications on behalf of the user.

Another approach to session migration is building specific (optimized) frameworks for each application. The Session Initiation Protocol (SIP) [9], chosen for controlling interactive multimedia sessions, supports all aspects of mobility [10].

SIP provides two migration schemes for mid-call mobility (session migration), namely *Third Party Call Control* and *Session Handoff* modes [10]. In Third Party Call Control (3PCC), the current terminal transfers the media to a new device, but retains the control of the session until its termination. On the contrary, Session Handoff (SH) transfers the whole session to a new device by notifying the remote peer. SIP also enables advanced features related to session migration, as connection splitting on different terminals [11].

SIP has a very good and complete mobility framework, but this has three main drawbacks: i) it works only for the specific application, ii) it is not integrated with link- and network-layer mechanisms to detect link failures and iii) it requires both peers to implement the framework.

Table I provides a brief comparison of the mechanisms available for session migration.

Rather surprisingly, many papers dealing with session migration only describe mobility protocols and pervasive environments, without any evidence of performance measurements from working testbed [12], [13], [14], [15], [16], [17], [18], [19] or even any proof of implementation [5], [6]; when some numerical results are available, they only concern quantitative measurements in few and very simple scenarios [4], [11], [20], [21]. A more accurate performance analysis is only available for the PA, in different networking scenarios and for different multimedia applications [8].

User evaluation is not taken into account in papers dealing with session migration; however, user preferences should be a leading factor in designing multimedia pervasive systems.

## III. A USER-CENTRIC MOBILITY FRAMEWORK

A lot of mechanisms and protocols for mobility are available, but most of them only cope with one specific aspect, mainly handover and terminal mobility. We cannot find a generic and flexible framework suitable for any kind of application.

Apart for the SIP protocol, no example of user-centric approach is known. However, a slight evolution has taken place during the years from a prominent network-centric paradigm to a more recent device-centric approach. Indeed, most of the oldest mechanisms are mainly focused on solving the problem from a network perspective. One solution was to keep invariant the IP address of devices and to deploy mobility infrastructures to trace their position within the network; this avoids maintaining host-specific information in the routing tables for each mobile host, which would not scale for the whole Internet. Anchors, proxies, multicast and end-to-end signaling have been used to this aim [2], [22], [23], [24], [25].

Recently, new approaches have been proposed that hide the changes in network/terminal to the applications without requiring any network infrastructure. We denote them as device-centric just because of this. We briefly discussed two interesting examples of this approach in Section II based on an invariant IP address, namely VNAT and DIP.

However, to fully implement the user-centric vision, the main principles of this paradigm must be brought into the network as well, in order to build communications around the users. That means users should be the session endpoints (sources and/or destinations), at least in principle, whilst devices act as the physical terminals. Such idea was the starting point to derive a general mobility framework at the network layer. Briefly, we assign network identifiers (addresses) to users; these identifiers will then be used by hosts and applications on behalf of the user for networking issues. The basic idea is

simple and straightforward, yet very powerful: however, at the best of our knowledge, it has never been used before.

This approach is not far from the principles behind VNAT and DIP; however, users have their own identifiers and we do not need using any device's address. That enables to overcome the main limitations of these two approaches.

Starting from the considerations above, we derived a mobility framework to address all aspects of mobility. It inherits the user-centric characteristic from the basic definition of the approach; moreover, we aimed at making it transparent to third parties and at keeping it independent of any specific network protocol.

The core of this framework works at the network layer and it deals with both terminal handover and session migration in a uniform way. The main idea is to maintain the same network address for the whole session duration, while the network and/or the device change. Obviously, we need some interaction with the application for handling its context during session migration and thus our framework spans across multiple layers.

### A. The concept of Personal Address

We think users may be assigned static and invariant identifiers, likely in the form of *Universal Resource Identifiers*[1] (personal mobility). User identifiers are then translated in temporary Personal Addresses depending on the underlying network technology.

We define the Personal Address (PA) as "a network identifier dynamically assigned to a user for a specific communication session." Personal Addresses are exploited to identify users instead of their terminals. Actually, we may use any kind of network identifier (for example, those provided by HIP [27]); until now, we have been focusing on IP addresses in order to deploy our framework in the current Internet.

The PA is specific for each communication session[2]; that prevents the risk of having address/port conflicts whether multiple sessions were initiated by the same user at some corresponding node.

The corresponding nodes involved in the session see at any time the same address, independently of the user movements and the devices used. Any migration (handover or session transfer) is transparent to remote applications and these latter are not required any specific functionality. This is one of the main assets of the PA scheme.

Figure 1 depicts the basic idea behind the PA concept. The user is assigned a network identifier (the IP address 1.1.1.1 in this example), this address is used to manage multimedia content (like a VVoIP session) on a small portable device (for example, a handheld), independently

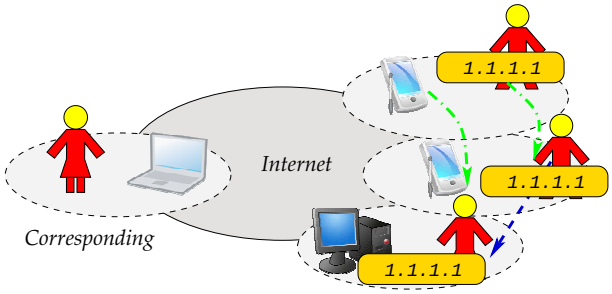



Figure 1. The Personal Address concept.

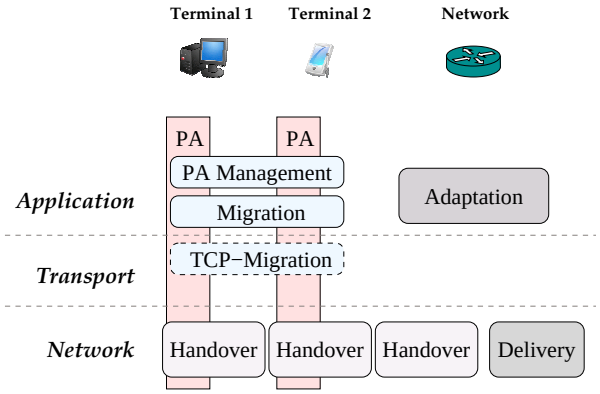



Figure 2. The architecture of the user-centric mobility framework based on the Personal Address.

of user movements across different networks (dash and dot arrows). The same network identifier continues to be used when the session is transferred to a different device, for example a television with a bigger monitor (dash arrow).

### B. The mobility framework

The mobility framework based on the Personal Address accounts for both terminal handover and session migration. A cross-layer approach is the most suitable solution to handle all of these issues efficiently. Our mobility framework mainly works at the network layer, although some operations are still necessary at the application layer to migrate the session context. Figure 2 depicts the architecture of the mobility framework.

The *PA Management* function retrieves a PA for the current user and makes it available for the application. In practice, this is accomplished by adding the Personal Address to the network interface of the current device (say Terminal 1 in Fig. 2); only one application at a time can bind to it.

The PA must be a topologically independent address, as it does not reflect the current point of attachment to the network. Thus, the network must provide a *Delivery* function to locate the user and to deliver packets to the current terminal. When the network or the device change, the PA remains the same, but the Delivery function must be updated with the new location.

The *Handover* task deals with two main issues: movement detection and handover management. It works at

[1] Actually, the user identifier may be a structure including information for different networks (see, for example, Chapter 15 in [26]).

[2] The idea of using a fixed IP for each user is not feasible in IPv4, which currently lacks addresses for devices, although it would be possible in IPv6. However, to maintain the approach as general as possible, we keep in mind the limitations of IPv4 and think at the PA as a dynamic address.

the network layer because the PA is physically configured as a network address of the host; moreover, handover is usually implemented in a more effective way at this layer.

The *Migration* function moves the user PA and the application context from one terminal to another and updates the Delivery function about the new location; this last task is essentially the same as in terminal handover, thus it can be accomplished by the same mechanism.

Usually, UDP is used to transmit multimedia traffic; however, for the sake of generality, we should also envision a *TCP-Migration* function whether TCP connections were used (this mainly happens for signaling).

The migration may involve heterogeneous devices or even different implementations of the same application. In order to preserve transparency for the corresponding peer, the network should provide an *Adaptation* function, which adapts the session to the new device. This function may deal with bandwidth requirements and different device capabilities; for example, for multimedia streams it involves transcoding of codecs, frame rates, aspect ratio, etc.).

The concept of PA and the structure of the mobility framework lead to three great advantages with respect to previous works (i.e., VNAT, DIP). First, the device-independent nature of the address enables an arbitrary number of migrations. Second, the presence of a Delivery function in the network also makes it possible to account for simultaneous mobility of both peers. Third, corresponding nodes are completely unaware of any mobility issue.

## IV. Implementing the Mobility Framework

The cross-layer architecture depicted in Fig. 2 allows to split the implementation into two main components, namely core functions common to all applications and application-specific functions.

### A. Core functions

The core of our mobility framework is the part which is common for every application. It includes all the functions working at the network and transport layers.

Summing up the main design guidelines discussed in Section III, we need suitable mechanisms to address the following issues:

- finding the user's current device(s);
- forwarding packets towards the user's current device(s);
- managing the change in network and/or device at the user side;
- updating the forwarding at the network side when the migration occurs.

Instead of thinking at entirely new mechanisms for the Handover and Delivery functions, we searched the scientific literature for architectural schemes that manage topological-independent network addresses. We found three alternatives: multicast, anycast and Mobile IP.

Multicast [28] provides an architecture to deliver packets to IP addresses (class D) that do not lie on the same network; unfortunately, at present multicast infrastructures are available only within few administrative domains.

Anycast [29] uses the same unicast IP address for multiple hosts, delivering packets only to one of them through standard routing mechanisms. However, routing is known to be slow to converge and the user client would unlikely be allowed to propagate its own routing information.

Mobile IP (MIP) [30] enables mobile nodes to use a fixed network address (the Home Address, HoA) independently of their location. It only requires a minimal infrastructure: one Home Agent (HA), owned by the user himself or some service provider, and optional Foreign Agents (FAs) in local networks for the sake of improving performance (only for IPv4 networks). This protocol registers a dynamic IP address (Care-of Address, CoA) for the Mobile Node (MN) with the Home Agent, so the latter can forward packets addressed to the HoA to the current host location. Moreover, it is transparent for Corresponding Nodes (CNs). MIP includes security mechanisms to prevent most attacks concerning flow redirection and spoofing.

MIP is a suitable solution for our framework, at least at the current implementation stage. In the MIP architecture, packets addressed to a Mobile Node (MN) are routed towards its Home Network, which is the network that the HoA topologically belongs to. As shown in Figure 3, if the MN does not lie in this network, packets are captured by the Home Agent, which answers to ARP Request packets on behalf of the MN with its own MAC address. The HA records the current position of the MN in terms of Care-of Address (CoA), which is the address packets must be sent to. This forwarding is made through tunneling mechanisms. The CoA is either the address of a local Foreign Agent (FA-CoA) or a temporary address assigned to the MN (Co-locate CoA, CoCoA); in both cases, packets finally reach the MN (see [30], [31] for details about MIPv4/6).

The same structure can be used in our mobility framework for the Handover and Delivery functions: MIP clients accomplish the tasks of the former, while the HA acts as the latter. However, we formally consider the HoA assigned to the user, rather than to one device of his; the device is simply a physical mean to use the address and so the HoA can be considered in every respect a Personal Address.

As a matter of fact, MIP was developed for terminal mobility, but its framework does not require the registration to be updated by the same host. We exploit this consideration to update the Delivery function from a different host in case of session migration[3]. From the HA perspective, this appears as a standard terminal migration, thus packets are redirected to the new terminal. Figure 4 sketches this mechanism; note that the FAs are not mandatory, although they are useful in the IPv4 version to speed up the process of detecting a migration whenever

[3]Obviously, the hosts migrating the session must share the same secret material requested by MIP security mechanisms.

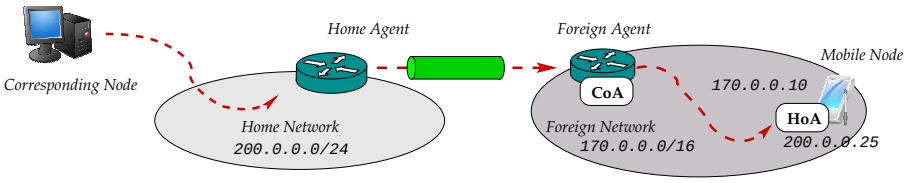



Figure 3. The basic operations of Mobile IP.

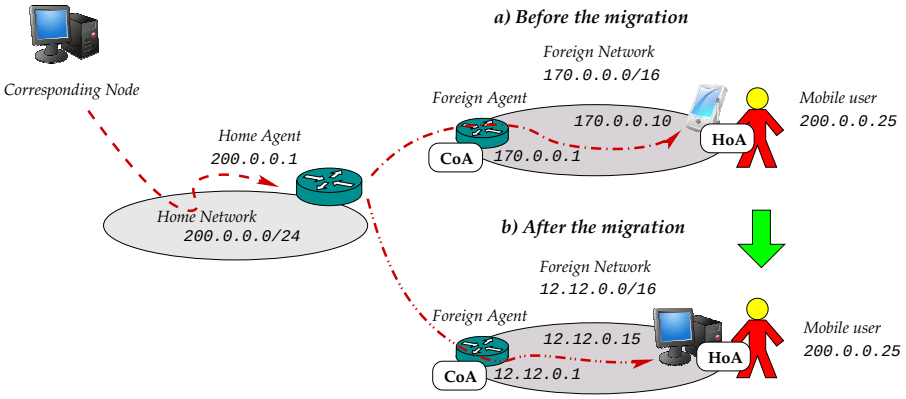



Figure 4. The implementation of the core functions of the mobility framework.

it occurs.

The last core function in our framework is TCP-Migration. It is only needed for applications relying on connection-oriented services from the transport layer. We have not still taken into account this issue in our implementation; however, we are currently thinking at splitting the connection at an intermediate agent, like MSOCKS [32]. This approach maintains transparency with respect to third parties (the Corresponding Nodes); moreover, the intermediate agent may be co-located with the HA.

### B. Application-specific functions

Four tasks are needed to handle application-specific issues of the PA Management and Migration functions:

- retrieving the Personal Address;
- setting up the Personal Address;
- saving, transferring and restoring the application context;
- removing the Personal Address.

A generic and common protocol may be considered to accomplish the first and third tasks; however, in the current implementation we delegated the application to take care of that. This way, we avoid adding new architectural elements; anyway, the two approaches are substantially equivalent. This issue will be covered in details in Section V.

Adding or removing an IP address to a network interface (the second and fourth tasks) is a trivial task and can be done by the network APIs of the operating system. It is worth noting the four tasks must be triggered by the application in any case.

Finally, we envisioned the presence of an adaptation server (e.g., transcoding). It may be located at the Home Agent, because all traffic is forced to cross this point. Our focus was mainly on the mobility infrastructure, so we did not consider either any solution nor implementation for the Adaptation function.

## V. Automatic Session Migration for Interactive Multimedia Sessions

For the purpose of evaluation, we need to select a specific application which exploits the PA mobility framework. Among several alternatives concerning multimedia transmission, we found interactive communications the most challenging and suitable example, as of their hard requirements for real-time operations. Further, to make the whole framework more appealing for user evaluation, we also decided to account for automatic migration by exploiting information about the user position [33].

We selected SIP as the signaling protocol for the interactive multimedia communication framework; thanks to its flexibility, this protocol can easily integrate session control (initialization and migration) and user localization.

This Section describes the localization system and the SIP messages used to implement the application-specific part of the PA mobility framework.

### A. User localization

User localization is made by two components: the Localization Server and a localization system.

The Localization Server gathers information by heterogeneous localization sources and computes the user position. In order to abstract from a particular localization system, the Localization Server was designed on top of the SAIL architecture [34], to enable the abstraction and integration of heterogeneous sensors within a common context-aware framework. In SAIL a localization system monitoring a set of mobile devices is abstracted as a set

of logical devices (one for each mobile device) providing their localization information. SAIL also provides a simple way for allowing multi-protocol network access to the data, which becomes useful for integrating the Localization Server with the Personal Address Mobility Framework.

Behind the Localization Server there is the actual localization work. This is done by sensor networks, a low-cost wireless technology that is expected to be largely deployed in the near future. Sensors are tiny devices with limited computing capabilities, integrating hardware for environmental monitoring (typically temperature, brightness, humidity, position, speed, acceleration) and short-range radios; these devices have networking capabilities as well.

Localization establishes which terminal the user is close to; this happens by evaluating whether he is inside the "usage range" (Area of Interest, AoI) of a terminal. Such an estimation of position is quite raw, but it is enough for session migration. To this purpose, fixed sensors (called anchors) are placed around each terminal, whose position is known "a priori"; the user carries with him a mobile sensor.

The mobile sensor, which needs to be localized by the system, periodically emits a beacon packet containing its identifier. As the anchor sensors receive the beacons, they compute the corresponding RSSI and send to the Localization Server all the pairs *<RSSI, anchor id>*. The Localization Server accumulates all the pairs and, using a couple of thresholds evaluated during a training phase[4], estimates the mobile position.

In our testbed the mobile position are obtained by means of a network of MicaZ sensors[5], which operate at the 2.4 GHz ISM frequency band and adopt the IEEE 802.15.4 communication protocol.

### B. Automatic migration

Proxy and Registrar servers are used in SIP to locate the user's current device. We co-locate the HA and these functions in a single element and extend its interface. Personal Addresses are taken from the Home Network address space and dynamically assigned by the Proxy as detailed in the following.

Figure 5 shows the signaling flow for setting up the session and for migrating it from one local terminal (LT1) to another (LT2); we extended the basic SIP signaling (see [9]) to account for migration-specific issues, i.e., retrieving the PA and transferring the session context.

The PA is assigned during SIP registration, but it will only be used when the session starts. There could be objections about the fact the address is assigned and perhaps never used; we argue this is our implementation choice for the sake of simplicity, but other solutions can be easily integrated, as providing extensions to the registration message to get the address when really needed (see, for example, the procedure outlined in [8]).

The registration makes the Proxy aware that the user is "on-line" and his location needs to be tracked in order to know his current device (the Local Terminal, LT). Thus, the user's Proxy subscribes the location service at the Location Server of the domain where the registration came from (SIP provides SUBSCRIBE/NOTIFICATION messages). We did not explicitly address the mechanism used to find the Location Server for a domain; however, that might happen through standard TXT or SRV resource records; moreover, the domain name could be retrieved by a reverse query for the source IP address of the registration message. After the subscription is completed, the Location Server starts immediately updating the user position and the closest available terminal (LT1 in the example).

Requests of setting up a session coming from Corresponding Nodes (CNs) are forwarded by the Proxy to the current device LT1. Before answering the INVITE message, this terminal adds the PA to its network interface and runs the MIP client registering the PA as the HoA; from now on, the SIP user agent begins using the PA just set up and all signaling and media are routed within the MIP architecture through the HA[6]. The same mechanism also applies if the mobile user's terminal initiates the session; the only difference is that the PA is set up before sending the INVITE message.

When the user moves closer to a new terminal (LT2 in Fig. 5), the Localization Server notifies the Proxy of the change. The latter updates the registration with the Registrar server. The Proxy is stateless, thus it does not know whether there is any active session; nevertheless it sends a REFER message to the previous device LT1. If a session is active on LT1 for the migrating user, this terminal initiates an INVITE/OK/ACK exchange to transfer the current session context to LT2. The INVITE message contains the session description, including the PA to use. During this phase, the MIP client is stopped on LT1 (after the OK is received) and the PA is removed from its network interface; then it is added on LT2 and another MIP instance is started with the same PA (after the ACK message). That updates the location of the PA inside the network; this sequence of operations avoids the duplication of the IP address on the two terminals. Note that the remote peer CN is completely unaware of the migration procedure as the IP address used in the session does not change[7].

## VI. The live demo

The live demo was organized at an Italian national science exhibition, named "Science Festival", held in

[4]The deployment of the localization system requires a training phase, which consists in configuring and calibrating the sensors providing localization information for the AoI. The calibration enables the sensors to recognize when a person equipped with a localization sensor (the mobile sensor) enters or exits the AoI.

[5]Crossbow, MicaZ Specification, http://www.xbow.com.

[6]SIP provides the current IP address of the remote terminal in the "Contact" field of the headers, thus the CN knows the PA to use after receiving the OK message.

[7]The presence of an adaptation server on the Home Agent would take care of transcoding, if needed. We did not deploy it in our testbed.

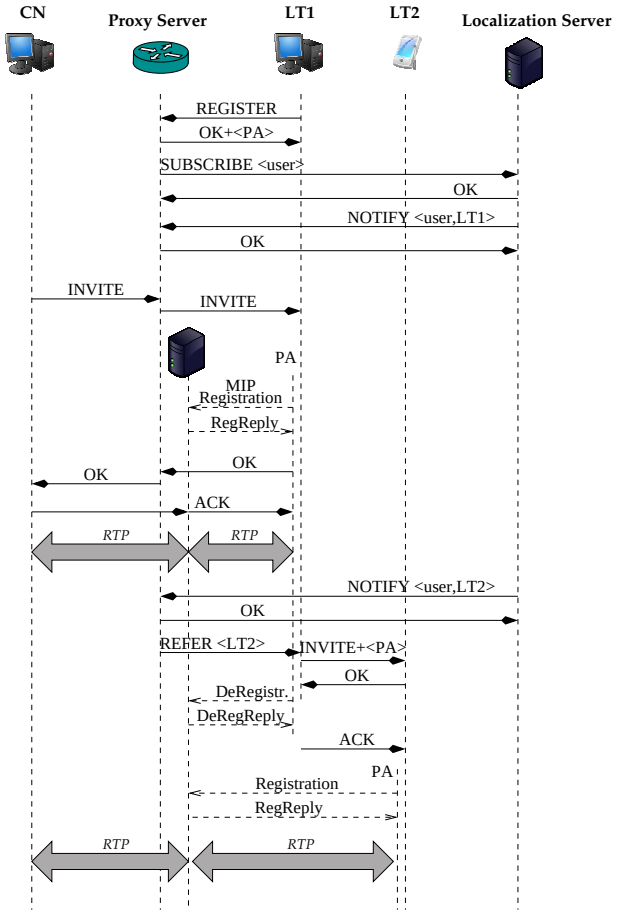



Figure 5. SIP signaling for session set-up and migration in the mobility framework.

Genoa on October 23rd – November 1st 2009. The Science Festival[8] is an important Italian dissemination initiative where researchers in different science and technology fields meet a large audience, ranging from business men, to students and families. The Festival saw 200.000 visitors that year: 160.000 people visited exhibitions and laboratories whilst 40.000 people attended conferences, shows and free-access events. Indeed, we only stayed at the Festival two days and got feedback from 101 users.

Visitors of the Science Festival are very heterogeneous and usually do not have in-depth technical knowledge; therefore they are more inclined to give plain and unconditioned reviews of demos and applications. Following these considerations, we decided to organize a live demo showing dynamic networking in action and to get user feedback about the PA framework.

Unskilled people have little or no knowledge at all about networking and related issues. It is also very difficult to let them try networking, as this latter is usually hidden behind applications and services of different nature. To this aim, we built a simple Video and Voice-over-IP (VVoIP) application based on the mobility framework with automatic session migration we have described in Section V. The same software also allows a manual control of the migration; it was already used for quantitative performance measurement in both local and Internet scenarios (see [8]).

The testbed was composed by three parts: the SIP agents (one Proxy server and the clients on each terminal), the localization infrastructure (wireless sensors and one Localization Server) and the mobility infrastructure (one MIP Home Agent, one MIP Foreign Agent and the clients on the mobile user's terminals). The demo followed the architectural scheme depicted in Figure 6: three terminals were used, one for the corresponding (fixed) user and two for the mobile user. All network elements (terminals and agents) were deployed in the same room, with direct connection among them (i.e., no Internet links); this corresponds to the scenario called "local" in our previous quantitative analysis shown in [8]. Two sensors were put near each terminal and an anchor sensor was tied to the wrist of the mobile user by a strip of velcro.

The users began a VVoIP conference; each user could see himself and the other person in the graphical user interface. This is only a minimal VVoIP application: the main window contains stored profiles and provides options to manage profiles, to set connection parameters and codecs, to start calls; the call window also enables to manage profiles and settings. Two rendering boxes are available in the call window: the big box displays the video of the remote user, the little box plays the video of the local user. Screenshots of the application are given in Figure 7.

The mobile user was then asked to move to and fro between his terminals, so he could evaluate the responsiveness of the automatic session migration (there was no way to separate localization and session migration); the corresponding users saw a freezing image during the migration and could assess the nuisance value of this interruption.

## VII. User Evaluation

User evaluation was conducted in three steps:

- **Presentation**: Users were presented a brief introduction about the user-centric vision and the VVoIP application they were going to try.


Figure 7. Screenshots of the VVoIP application. Upside the main and the call windows are shown; items under menu entries are shown below each window (the Tools entry is currently empty).

[8]Festival della Scienza, web site: http://www.festivalscienza.eu.

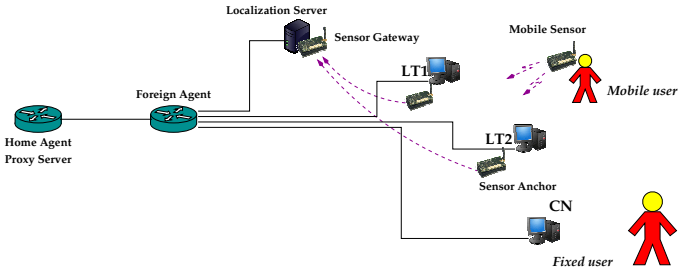



Figure 6. Set up for the live demo. All terminals (LT1, LT2, CN), the Foreign Agent and the Localization Server lie on the same local network; the Home Agent is co-located with the Proxy and is one hop away the Foreign Agent.

- **Live demo**: Users were invited to try the automatic migration during a video call. We prepared three different locations with a laptop each. Each demo session was attended by two users, the first user made a video call from one location, and the second one moved between the other two locations to test the migration. Then the roles were inverted.
- **Assessment**: After they had tried the demo, users were asked to compile a questionnaire. The assessment phase was not limited to that issue; indeed, it was also extended to the previous two phases by observing commonly asked questions from users, their difficulties while using the migration service, their comments and suggestions for improvement.

The questionnaire was proposed to potential end-users to assess their interest in automatic session migration and to gather their feeling about the usefulness and applicability of such feature; questions were prepared with the support of a psychologist and took into account background skills, familiarity with technology, personal assessment of the system, social impact of this kind of technology.

The questionnaire was organized in three parts:

- First part concerned the user's profile. We were interested in knowing age, gender, education, work, familiarity with and use of technologies in daily life.
- Second part focused on the assessment of the live demo. After they had tried the automatic migration of a video call, we asked users to assess their experience in terms of usability and responsiveness of the migration service.
- Last part concerned the investigation of user preferences about physical and ergonomic characteristics of sensors, the possible contexts where the migration may be used and potential issues such as security and privacy.

### A. *User's profile*

The questionnaire was filled in by 101 people (48 females, 53 males), aged between 10–69. In the analysis of results we divided the subjects based on their age, as shown in Table II. Most results later on are expressed as percentages and are relative to each of these classes; many questions allowed multiple answers.

The Science Festival especially draws the attention of people with a natural gift and practice for technology, usually the younger generation; moreover, many teachers bring their students to this event to get them in touch with applied sciences and future technologies. For this reason, most people who completed the questionnaire were schools, high schools and university students (69 people); Table III shows the average of years they had spent in education, together with other statistical parameters as standard deviation and minimum and maximum values. The remaining part of the sample was rather heterogeneous in terms of education and employment; they were teachers, employees, professional men, housewives, unemployed people and pensioners.

TABLE II.
NUMBER OF QUESTIONNAIRES FOR DIFFERENT AGE GROUPS

| Age | Males | Females | Total |
|---|---|---|---|
| <14 | 14 | 11 | 25 |
| 15–19 | 20 | 16 | 36 |
| 20–29 | 9 | 6 | 15 |
| 30–50 | 7 | 9 | 16 |
| >50 | 3 | 6 | 9 |
| **Total** | 53 | 48 | 101 |

TABLE III.
YEARS SPENT IN EDUCATION

| Age | Average | Standard Deviation | Min | Max |
|---|---|---|---|---|
| <14 | 6.04 | 0.45 | 5 | 7 |
| 15–19 | 9.94 | 1.01 | 9 | 12 |
| 20–29 | 14.73 | 3.99 | 8 | 18 |
| 30–50 | 14.92 | 2.53 | 13 | 18 |
| >50 | 16.75 | 2.26 | 13 | 18 |

Regarding the use and familiarity with technology, most of the sample feels skilled with technology and uses several devices every day, without distinction between genders. Evaluation of familiarity with technology is placed on a Likert scale from 1 up to 5, where 5 indicates a great familiarity and 1 no familiarity; moreover, visitors were asked to tick off equipment they usually use on a list of 12 technological devices quite common in everyday life. Table IV analyses the number of devices used and the familiarity with technology in terms of mean value, standard deviation, minimum and maximum value.

Detailed statistics about usage of each of the 12 devices in the list are given in Figure 8, classified according to the age classes identified in Table II. The most commonly

TABLE IV.
NUMBER OF USED DEVICES PROPOSED IN THE QUESTIONNAIRE AND FAMILIARITY WITH TECHNOLOGY

| | Avrg | Std Dev | Min | Max |
|---|---|---|---|---|
| Number of devices (¡ 12) | 7.35 | 2.07 | 1 | 11 |
| Familiarity with technology | 3.92 | 0.74 | 2 | 5 |

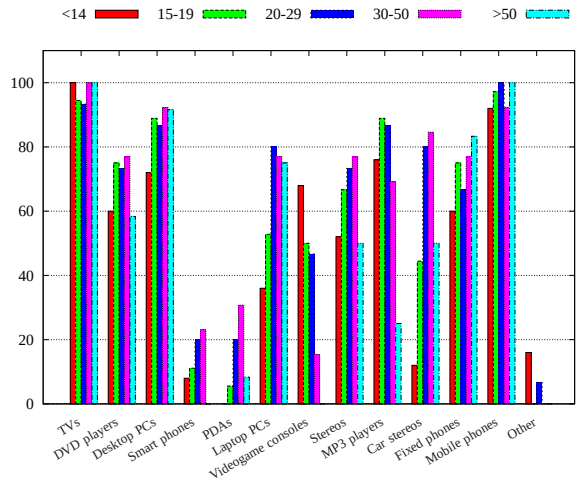



Figure 8. Statistics about device usage for different age groups.

used devices are televisions (98%), mobile phones (97%) and PCs (86%), which are used by almost all people regardless of their age and gender. Other kinds of devices are most suitable to different age ranges: video consoles and MP3 players are used by younger subjects, whilst car stereos and laptops are used by adults. Expensive and niche devices are currently less used, examples are PDAs (10%) and smart phones (12%).

### *B. Assessment of the demo*

After they had tried the automatic migration of the video call, we asked users their feeling about the application. The main purpose was to check the intuitiveness of the framework, the performance of the underlying network mechanism and the usefulness of the migration feature. The first question was the effort in understanding the migration feature (immediateness of use, i.e. *Rapidity*), which only means the level of difficulty in learning how to use the migration service (in practice, tying a sensor-wristband) and not technical details. The second question concerned performance, and thus how quick the migration happened (*Speed*). The following questions were about the usefulness of session migration among multimedia devices: how much the user had liked this feature (*Pleasant*), their assessment about its usefulness in everyday life (*Utility*) and how much they would have spent to use it (*Value*). Users answered these questions on a Likert scale from 1 to 5, where 5 is the more positive and 1 is the more negative opinion. Each score corresponds to a meaning adjective, specific for each question; for example, the judgment about migration usefulness was proposed as follows: “very useful” (5), “useful” (4), “not so useful” (3), “not useful” (2), “harmful” (1). The assessment of the economic value of the feature was proposed in Euros according to the following arbitrary scale: “above 20” (5), “5 up to 20” (4), “less than 5” (3), “nothing, I would only use it whether it were free” (2), “nothing, I would not use it” (1). Mean values for each age group are shown in Table V.

TABLE V.
USER ASSESSMENT OF THE MIGRATION FEATURE FOR EACH AGE RANGE

| Age | Rapidity | Speed | Pleasant | Utility | Value |
|---|---|---|---|---|---|
| <14 | 4.16 | 4.24 | 4.76 | 4.48 | 4.12 |
| 15–19 | 3.94 | 4.14 | 4.22 | 3.89 | 3.97 |
| 20–29 | 4.13 | 4.21 | 4.50 | 4.00 | 3.31 |
| 30–50 | 3.92 | 4.08 | 4.46 | 4.00 | 3.42 |
| > 50 | 3.92 | 4.08 | 4.75 | 4.08 | 4.08 |

As the results show, the effort to understand the user interface and the migration feature was acceptable, even for older and unskilled people; further, we may note younger generations required less effort, as probably they are friendlier and more used to modern technologies than eldest people, which often are less prone to learn new technologies. The rapidity of migration mainly depends on the Personal Address framework as the delay introduced by media codec in video acquisition and rendering is almost negligible; we got a good score here, thus we can take the quantitative analysis for the local testbed given in [8] as a good benchmark for assessing the effectiveness of session migration.

The second part of the evaluation shows a substantial interest by users towards the demo scenario and their willingness to accept the migration feature in the next future; this feedback motivates our work and future research in this field. Users liked the feature of migrating an interactive video session among devices, they considered the service useful and they would have spent some money for it. A MANOVA [35] analysis was conducted to check if there were differences in the answers by different age groups; no relevant variation has arisen among those groups in assessing usability and suitability of the migration service to the needs and interests of potential users.

Finally, innovation was evaluated by asking users whether they had ever found session migration in any application. Most people (86% of interviewed) considered the migration service innovative, as they had never seen before this functionality. A small percentage (9%) said they had already seen similar application, but oral interviews following the compilation of the questionnaires pointed out that most of them referred to side aspects of the demo, which are not related with session migration, as the use of webcams and VVoIP calls. Finally, few users (about 3%) found the migration service similar to other kinds of functionality: the GPS localization available in the iPhone, the automatic re-tuning to a different frequency providing the same station when the first signal becomes too weak (e.g., when moving out of range) usually found in car stereo systems (AF function

of the RDS[9] system), the handover mechanism of cellular networks.

### C. User preferences

Although the migration framework mostly works at the network layer, it implements a service that directly interacts with the user, thus it is important to keep into account the user's needs and to involve people in the development phase. Hence the last part of the questionnaire investigates how users perceive our technology and their feeling with related ethical issues; in particular, we were interested in understanding whether they found the migration framework intrusive, whether they were afraid about their privacy to be violated and which kind of sensors they would have been willing to interact with.

Session migration is always related to applications, as each of them has its own context to be transferred; however, the migration is not meaningful for every possible application (a file transfer is a typical example where the migration is not useful), thus it is important to find out for what applications users expect the feature to be available.

We selected a list of session-based networked applications, considering interactive sessions (multimedia, chat), content access (broadcast and on demand media, Internet browsing), entertainment (videogames) and generic work applications. Users checked off those they would find our service most useful (multiple selections were allowed); indeed, our demo falls within the most rated topic. The full classification, in decreasing order of preferences is: "phone calls" (74%), "watching TV" (53%), "listen to music" (49%), "Internet browsing" (40%), "videogames" (38%), "chat" (34%), "office applications" (23%), and "other" (2%).

Figure 9 shows the preferred user applications for each age group; in this case there are significant differences. For example, 80% of users aged under 14 would like to use the migration service to play videogames, whilst the corresponding percentage for the other groups is significantly lower (range 15–19=31%, range 20–29=33%, range 30–50=0%, >50=17%). Note that the youngest people always have higher percentages than other groups; this means they checked off a larger number of items for this question; the only exception is the "office applications" item, as users under 14 are students and are not involved with such activity.

From a technological point of view, many features can be implemented in a easier way on certain devices: writing software for general purpose PCs is much simpler than developing Symbian[10] applications for smartphones or firmware for televisions. Unfortunately, users expect the migration feature on most of their daily equipment: "TVs" (83%), "mobile phones" (71%), "desktop PCs" (67%), "laptop PCs" (51%), "MP3 players" (33%), "stereos" (26%), "fixed phone" (24%), "DVD players" (23%), "PDAs" (20%), "smart phones" (19%), "car stereos"

[9]Radio Data System, http://www.rds.org.uk/.
[10]The Symbian Fundation, http://www.symbian.org/.

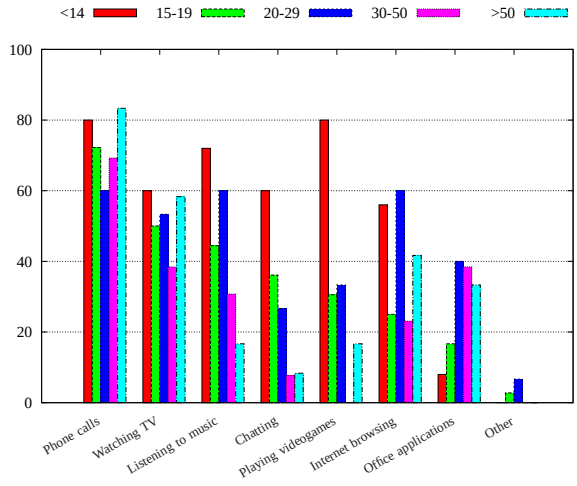



Figure 9. Statistics about which applications users would like session migration to be available for.

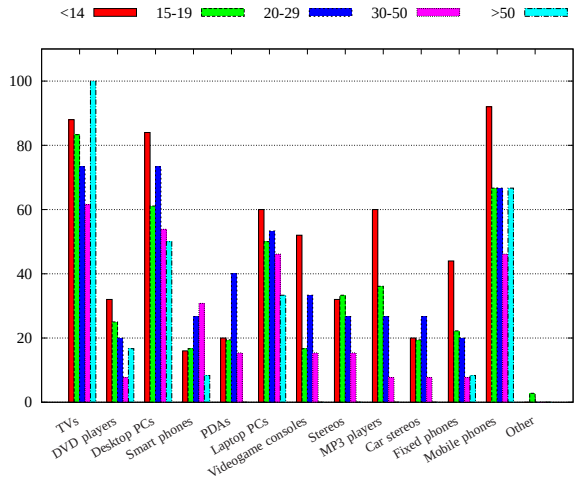



Figure 10. Statistics about which devices users would like session migration to be available on.

(17%) and "others" (1%). This implies the algorithm must be kept simple enough to be ported on a wide range of different devices. Again, users were allowed multiple selections.

The preferred devices vary with age (see Figure 10): 100% of the oldest users (above 50) checked off television, while 92% of youngest people (under 14) selected the cell phone. Other devices voted by a large number of people are desktops and laptops.

Session migration is a component of pervasive communication. The latter relies on complex frameworks which may be difficult to deploy in certain scenarios [20]; however, users may not need pervasive communication everywhere. Indeed, user feedback was quite surprising for us: they mainly expected session migration at home (which is the preferred answer of eldest people), and only in lower percentage everywhere (which is the preferred answer of youngest users). The full classification is: 50% "at home", 32% "everywhere", 26% "at school/work", 15% "in street", 4% "nowhere" and 2% "other". Figure

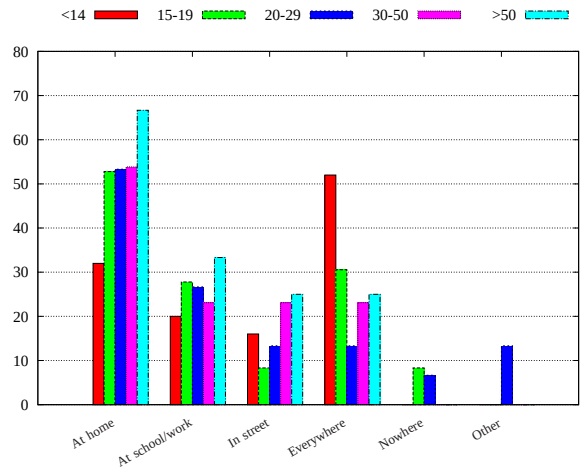



Figure 11. Statistics about where users would like session migration to be available.

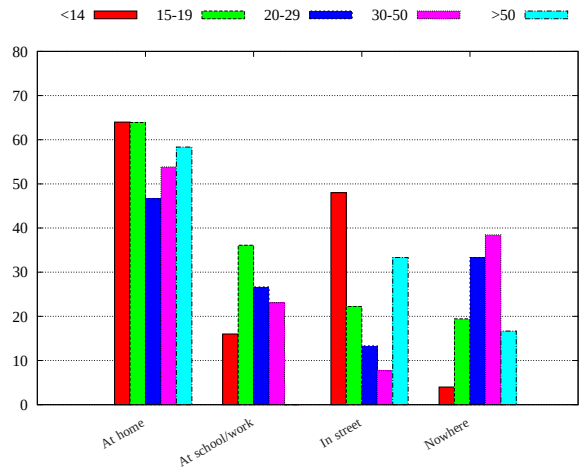



Figure 12. Statistics about where users would be willing to be tracked.

11 shows the detailed answers for each age group.

As a side effect of automatic session migration, users' movements have to be tracked and this may concern privacy issues for many people, hence the natural question for users was where they would be willing to be located by sensors. Most users checked off their own home, whilst other options got far less preferences: 60% "at home", 27% "in street", 24% "at school/work" and 20% "nowhere". The large gap among "home" and other options sounds quite strange; perhaps people take privacy for granted in personal environments, i.e., a tracking system working at home keeps all data on private equipment and does not allow anybody to access such information.

Taking into account the behavior according to age, all groups agreed that home is a perfect place to locate sensors, while disagree in the other responses (see Figure 12). People aged 20–29 and 30–50 are less inclined to be located with respect to other groups, people under 14 and above 50 are more willing than others to be also located "in street" and people aged 15–19 have a higher percentage "at school/work" than other groups. This last fact is quite curious as well, as teenagers often care about letting their parents know they are (or are not) at school! Another consideration regards the fact many people above 50 have already left work.

Just to be sure our users were aware of the relationship with current technologies, people who did not like to be located anywhere were asked if they knew cellular systems indeed maintain information about the cell of their phone (and implicitly of their position); everyone said yes to this question and thus we argue people are willing to postpone their qualms about privacy whether they are really interested in the service.

Another issue in pervasive communication is the presence of public, shared and private devices in the environment. That poses security concerns around who is allowed to use what. From the user's perspective, the problem is twofold: which devices the user would use among those available and whether the user would share

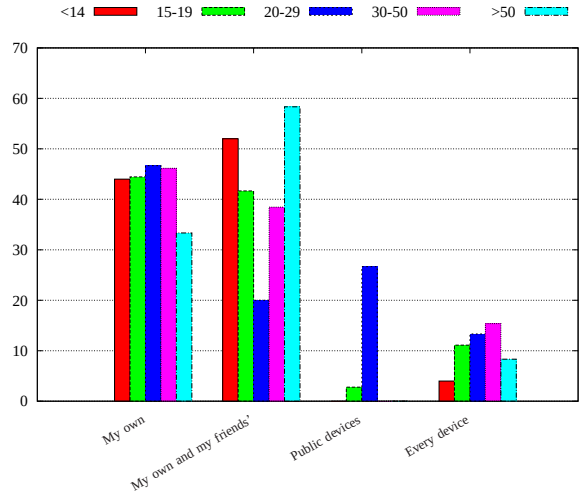



Figure 13. Statistics about which devices people would migrate their sessions to.

his own devices with other people.

The first side of the problem concerns the use of equipment by the user. To this aim, we identified three classes: own devices, devices belonging to people the user knows and public/third parties' devices. Most people would use their own devices and those of their friends, but few users are interested in other devices; the classification we got from the questionnaire is: own devices (44%), own devices and devices of my friends (43%), all devices (10%), public devices (5%). Note this time only a single answer was allowed.

Figure 13 shows the results for the different groups of users. Only people aged between 20–29 are interested in using public devices; indeed, this group includes university students and young workers which are usually more used to share computers and other devices with their colleagues (at the university, at office); on the other hand, teenagers and elder people are more bound to the concept of personal computers and often do not know features as centralized profiles and authentication management.

The other side of the problem concerns sharing of user's devices: if the owner is not currently using them,

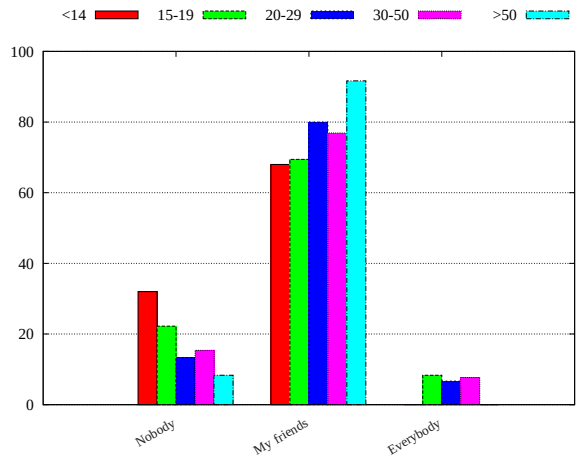



Figure 14. Statistics about willingness of users to share their personal devices with other people.

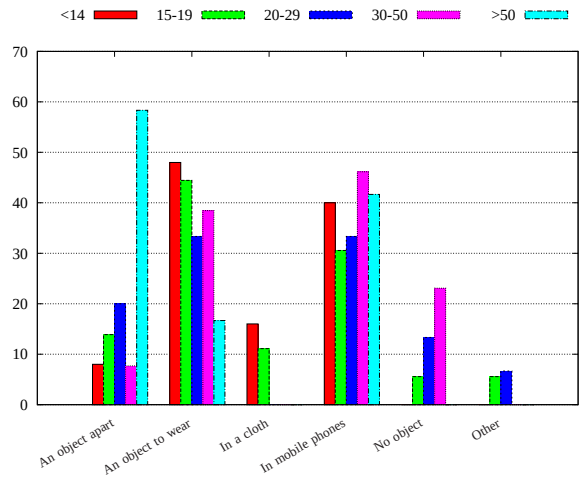



Figure 15. Statistics about user's preference on the placement of the sensor they have to bring with them.

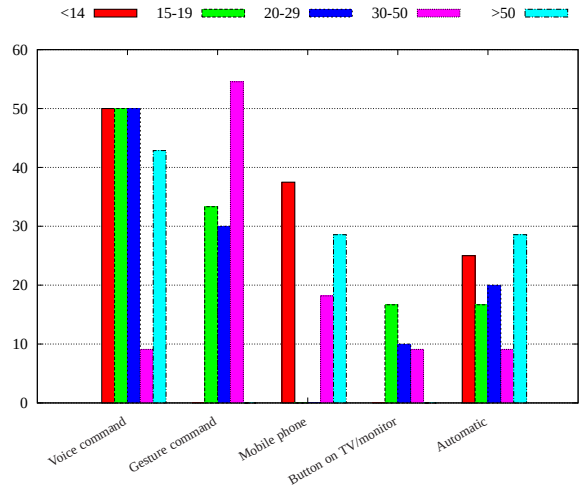



Figure 16. Statistics about user's preference on the control of session migration.

they may be used by other people to enrich the environment pervasiveness. The result is congruent with the previous question: people are not inclined to share devices with third parties. Indeed, most of them would only let their friends to use their own devices (75%), some people would not lend devices to anyone (10%) and very few users would make them available to everybody (5%). Figure 14 shows a slight trend for older users to share their devices with friends.

Coming back to more technical issue, some questions were devoted to sketch enhancements and guidelines for the future development of personal mobility infrastructures. These questions concerned the interaction of users with session migration, mainly integration of sensors in daily life and alternative forms of control of the migration process.

Sensors represent the most intrusive part of the system, as one of them needs to be carried by the user. Other techniques might be used for localization, but sensor networks are currently low-cost and tiny devices, which are expected to be easily spread in most environments in the near future. As of these characteristics, sensors may be integrated in several objects users usually bring with them, and the main question here is what kind of object the users would like.

We proposed a list of items in the questionnaire; 40% of users would prefer sensors as an object to wear, 37% would like it were integrated in their mobile phone, 18% would bring it as an object apart, 8% would integrate it in an article of clothing, 7% would not like any object, 3% would choose other options. Many differences arose among answers from the different groups (see Figure 15). Users who chose an "object to wear" or "other" specified that it could be a clock.

An automatic migration system must take care of selecting the right device to use among those available; obviously, it needs to account for user preferences and impairments, security requirements, and so on.

Despite of the good logic it can implement, an automatic decision maker may not reflect the current user's need. Thus, some form of direct user control is needed in addition to the fully autonomic feature. Given the nature of the system (pervasive communication, multiple heterogeneous devices, distributed computation), it is necessary to find the most appropriate way for users to interact with the mobility framework.

Several alternatives were envisioned and users gave their preferences as follows: 38% of users would control migration through a voice command, 26% through a gesture command, 19% would prefer automatic migration, 17% would use the mobile phone to control the session, 7% would like a button on the devices (TV or PC monitor). There are many differences among the different age groups (see Figure 16).

## VIII. Remarks on the Output from the Evaluation

The results coming from questionnaires, the observation of the user interaction with the service and the

analysis of type and number of errors allowed us to give a positive judgment about usability of the migration service (conclusions are drawn using the ISO 9241 standard [36]). This evaluation takes into account three parameters:

- **Effectiveness**: The level of achievement of the objectives. The first and simplest effectiveness index is the achievement of the objective: a product is effective if it carries out its task. Otherwise, if the objective is not achieved, the effectiveness can be measured in terms of number of operations towards its completion state. The migration service has been evaluated as "effective" because all users achieved the goal in the live demo, i.e. they migrated a video call from one computer to another one, without they were required to take any control action.
- **Efficiency**: The effort required by the user to achieve the goal. The migration service has been evaluated "efficient" because users easily learned how it works and they quickly began to use it.
- **User Satisfaction**: The perceived usefulness of the service by users. The service has been evaluated "useful" by users and they talked positively about the migration concept.

More feedback was collected by analyzing answers, comments and critics from the users during the demo. This information provided us useful indication about aspects that should be taken into account in developing user-centric systems. For example:

- **Security**: Users are interested in security and privacy issues involved in using devices owned by other people.
- **Human-Machine Interface** (HMI): Many users, especially the youngest, underlined the importance of improving the service interface and physical aspect of sensors; obviously these are minor remarks for our purposes, as our framework works at the network layer and the VVoIP application was only developed to set up a live demo, while at the current stage sensors are only prototypes and are far from being a real product. About control of migration, a clear and unique trend does not appear from users; indeed, answers from users suggest that different solutions could be integrated, according to different user profiles and preferences.

Finally, the last remarkable aspect to be considered is the tendency of adult users to perceive the migration service as a futuristic technology, while younger users seem more inclined to use this technology in daily life straightaway.

## IX. Conclusions

In this paper, we have applied the user-centric paradigm to dynamic networking. We have discussed the concept of Personal Address and we have described a cross-layer framework which accounts for different aspects of mobility. This framework exploits a cross-layer architecture, which brings together efficiency at the network layer with flexibility at the application layer. The other important benefit is transparency for unaware corresponding applications.

This paper extends our previous work about this topic by presenting the user evaluation we carried out at a national science exhibition. A live demo was built by using the Personal Address framework for a VVoIP application with automatic session migration. SIP was used as the session control protocol at the application layer, MIP implements the mobility framework at the network layer, sensor networks were used for tracking the user position and the SAIL framework was used for the Localization Server; only few extensions to SIP were necessary to account for migration-specific issues at the application layer.

Visitors of the exhibition were invited to try the live demo; evaluation was done by written questionnaires and by direct interview, with the support of a psychologist skilled in this field. Outcomes from the evaluation has given positive feedback about the effectiveness of our framework and outlined general indications for designing pervasive communication systems. To the best of our knowledge, no trials of this kind have been ever carried out before.

Users liked the automatic migration feature and they positively assessed its effectiveness in terms of timeliness and speed. Moreover, they found the application easy to use, that especially thanks to the user-centric approach followed by the architecture design.

As general concerns, users would have expected to keep more control over their sessions, mostly by advanced interaction interfaces as voice and gesture recognition; fully automatic migration was seen as something that may elude what they really mean to do. Automatic migration requires locating and tracking the users, and thus privacy issues must be taken carefully into account before considering applicability of such kind of system in environments other than private (especially home). Finally, there is not a large willingness in sharing devices, although there are significant differences among different age groups.

## Acknowledgment

The authors would like to thanks Stefano Chessa and his staff at the Institute of Information Science and Technologies (ISTI) of the Italian National Research Council (CNR) in Pisa for their invaluable help in providing the localization by sensor networks framework used in the live demo. The authors would also like to thanks the psychologist Ludovica Primavera for his assistance in preparing and analyzing the questionnaires.

This work was partially funded by the EU 6th framework program, contract no. 38419 (Intermedia NoE).

## References

[1] I. G. Niemegeers and S. M. H. D. Groot, "Research issues in ad-hoc distributed personal networking," *Wireless Personal Communications*, vol. 26, no. 2–3, pp. 149–167, 2003.

[2] D. Le, X. Fu, and D. Hogrefe, "A review of mobility support paradigms for the Internet," *IEEE Commun. Surveys Tuts.*, vol. 8, no. 1, pp. 38–51, 1st Quarter 2006.
[3] N. Banerjee, W. Wu, S. K. Das, S. Dawkins, and J. Pathak, "Mobility support in wireless Internet," *IEEE Wireless Commun. Mag.*, vol. 10, no. 5, pp. 54–61, October 2003.
[4] K. Ohta, T. Yoshikawa, and T. Nakagawa, "Adaptive terminal middleware for session mobility," in *Proceedings of the $23^{rd}$ International Conference on Distributed Computing Systems Workshops (ICDCS 2003)*, Providence, Rhode Island, USA, May 19–22, 2003, pp. 394–399.
[5] W. Lu, A. Lo, and I. Niemegeers, "Session mobility support for personal networks using Mobile IPv6 and VNAT," in *$5^{th}$ Workshop on Applications and Services in Wireless Networks (ASWN05)*, Paris, France, June 29 – July 1, 2005.
[6] R. Kohn, "Delegated IP: A Mobile IPv6-based protocol to support session delegation," in *IEEE International Conference on Communications (ICC'08)*, May 19-23, 2008, pp. 3279–3285.
[7] G. Su and J. Nieh, "Mobile communication with Virtual Network Address Translation," Columbia University, Technical Report CUCS-003-02, February 2002. [Online]. Available: http://www.cs.columbia.edu/techreports/cucs-003-02.pdf
[8] R. Bolla, R. Rapuzzi, and M. Repetto, "An integrated mobility framework for pervasive communications," in *IEEE Global Communications Conference (IEEE Globecom 2009)*, Honolulu, Hawaii, USA, November 30 – December 4 2009.
[9] J. Rosemberg, H. Schulzrinne, G. Camarillo, A. Johnston, R. Sparks, A. Handley, and E. Schooler, "SIP: Session Initiation Protocol," RFC 3261, June 2002. [Online]. Available: http://www.ietf.org/rfc/rfc3261.txt
[10] H. Schulzrinne and E. Wedlund, "Application-layer mobility using SIP," *Mobile Computing and Communication Review*, vol. 4, no. 3, pp. 47–57, July 2000.
[11] M.-X. Chen, C.-J. Peng, and R.-H. Hwang, "SSIP: Split a SIP session over multiple devices," *Computer Standards & Interfaces*, vol. 29, no. 5, pp. 531–545, July 2007.
[12] R. Shacham, H. Schulzrinne, S. Thakolsri, and W. Kellerer, "The virtual device: Expanding wireless communication services through service discovery and session mobility," in *IEEE International Conference on Wireless And Mobile Computing, Networking And Communications (WiMob'2005)*, Montreal, Canada, Aug. 22–24, 2005, pp. 73–81.
[13] H. Schulzrinne, X. Wu, S. Sidiroglou, and S. Berger, "Ubiquitous computing in home networks," *IEEE Commun. Mag.*, vol. 41, no. 11, pp. 128–135, November 2003.
[14] S. Berger, H. Schulzrinne, S. Sidiroglou, and X. Wu, "Ubiquitous computing using SIP," in *Proceedings of the 13th international workshop on Network and operating systems support for digital audio and video (NOSSDAV'03)*, Monterey, CA, USA, Jun. 1–3, 2003, pp. 82–89.
[15] K. Kaneko, H. Morikawa, and T. Aoyama, "Session layer mobility support for 3C everywhere environments," in *Proceeding of the Sixth International Symposium on Wireless Personal Multimedia Communications (WPMC 2003)*, vol. 2, Yokosuka, Japan, October 2003, pp. 347–351.
[16] M. Hasegawa, H. Morikawa, M. Inoue, U. Bandare, H. Murakami, and K. Mahmud, "Cross-device handover using the service mobility proxy," in *Proceedings of 6th International Symposium on Wireless Personal Multimedia Comunications (WPMC2003)*, vol. 2, Yokosuka, Japan, October 2003, pp. 357–361.
[17] H. Song, H.-H. Chu, and S. Kurakake, "Browser session preservation and migration," in *The 11th International World Wide Web Conference (WWW 2002)*, Honolulu, Hawaii, USA, May 7–11, 2002, pp. 7–11, poster Session.
[18] A. Dutta, J. Chen, S. Das, M. Elaoud, D. Famolari, S. Madhani, A. McAuley, M. Tauil, S. Baba, T. Maeda, N. Nakajima, Y. Ohba, and H. Schulzrinne, "Implementing a testbed for mobile multimedia," in *IEEE Global Telecommunications Conference (GLOBECOM '01)*, vol. 3, San Antonio, TX , USA, Nov. 25–29, 2001, pp. 1944–1949.
[19] S. Salsano, A. Polidoro, C. Mingardi, S. Niccolini, and L. Veltri, "SIP-based mobility management in next generation networks," *IEEE Wireless Commun. Mag.*, vol. 15, no. 2, pp. 92–99, April 2008.
[20] R. Shacham, H. Schulzrinne, S. Thakolsri, and W. Kellerer, "Ubiquitous device personalization and use: The next generation of IP multimedia communications," *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 3, no. 2, May 2007, article No. 12.
[21] N. Banerjee, A. Acharya, and S. K. Das, "Seamless SIP-based mobility for multimedia applications," *IEEE Netw.*, vol. 20, no. 2, pp. 6–13, March-April 2006.
[22] T. R. Henderson, "Host mobility for IP networks: A comparison," *IEEE Netw.*, vol. 17, no. 6, pp. 18–26, Nov. - Dec. 2003.
[23] P. Reinbold and O. Bonaventure, "IP micro-mobility protocols," *IEEE Commun. Surveys Tuts.*, vol. 5, no. 1, pp. 40–57, Third Quarter 2003.
[24] A. T. Campbell, J. Gomez, S. Kim, C.-Y. Wan, Z. R. Turanyi, and A. G. Valko, "Comparison of IP micromobility protocols," *IEEE Wireless Commun. Mag.*, vol. 1, no. 2, pp. 72–82, February 2002.
[25] D. Saha, A. Mukherjee, I. S. Misra, and M. Chakraborty, "Mobility support in IP: A survey of related protocols," *IEEE Netw.*, vol. 18, no. 6, pp. 34–40, Nov.-Dec. 2004.
[26] V. Kumar, M. Korpi, and S. Sengodan, *IP Telephony with H.323*. Wiley, 2001.
[27] R. Moskowitz and P. Nikander, "Host Identity Protocol (HIP) architecture," RFC 4423, May 2006. [Online]. Available: http://www.ietf.org/rfc/rfc4423.txt
[28] J. Mysore and V. Bharghavan, "A new multicasting-based architecture for Internet host mobility," in *Proceedings of the 3rd annual ACM/IEEE international conference on Mobile computing and networking (MobiCom'97)*, Budapest, Hungary, 1997, pp. 161–172.
[29] C. Partridge, T. Mendez, and W. Milliken, "Host anycasting service," RFC 1546, November 1993. [Online]. Available: http://tools.ietf.org/rfc/rfc1546.txt
[30] C. Perkins, "IP mobility support for IPv4," RFC 3344, August 2002. [Online]. Available: http://www.ietf.org/rfc/rfc3344.txt
[31] D. Johnson, C. Perkins, and J. Arkko, "Mobility support in IPv6," RFC 3775, June 2004. [Online]. Available: http://www.ietf.org/rfc/rfc3775.txt
[32] D. A. Maltz and P. Bhagwat, "MSOCKS: An architecture for transport layer mobility," in *Proceedings of the $17^{th}$ Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'98)*, vol. 3, San Francisco, California, USA, March 29 – Apr 2 1998, pp. 1037–1045.
[33] R. Bolla, R. Rapuzzi, M. Repetto, P. Barsocchi, S. Chessa, and S. Lenzi, "Automatic multimedia session migration by means of a context-aware mobility framework," in *The International Conference for Mobility Technology, Applications and Systems (ACM Mobility Conference 2009)*, Nice, France, Sep. 2–4, 2009.
[34] S. Chessa, F. Furfari, M. Girolami, and S. Lenzi, "SAIL: a sensor abstraction and integration layer for context aware architectures," in *34th EUROMICRO Conference on Software Engineering and Advanced Applications - Special Session on Software Architecture for Pervasive Systems (SAPS)*, Parma, Italy, Sep. 3–5, 2008, pp. 374–381.
[35] J. P. Stevens, *Applied multivariate statistics for the social sciences*. Mahwah, NJ, USA: Lawrence Erblaum, 2002.

[36] ISO 9241-11:1998, "Ergonomic requirements for office work with visual display terminals (VDTs) – part 11: Guidance on usability," 1998. [Online]. Available: http://www.iso.org/iso/catalogue_detail.htm?csnumber=16883

**Raffaele Bolla** received the Ph.D. degree in Telecommunications in 1994 from the University of Genoa. Since November 1996 he is a researcher in the Department of Communications, Computer and Systems Science (DIST) at the University of Genoa, and since September 2004 he is associate professor in the same Department. He is also a member of CNIT, the Italian inter-university consortium for telecommunications.

He acts as reviewer for many different international magazines and participates to technical committees of international congresses (SPECTS, QoS-IP, Globecom, . . .). He has co-authored over 140 scientific publications in international journals and international conference proceedings.

Prof. Bolla's current research interests are in: i) mechanisms and techniques for energy consumption reduction in IP networks, ii) modeling and design of Service Specific Overlay Networks, iii) advance platform for Future Internet nodes (Flexible Software Router), iv) advance mobility management in "user centric" networking approaches.

**Riccardo Rapuzzi** received his "laurea" degree cum laude in Computer Science from University of Genoa in 2004. In 2009, he obtained the Ph.D. degree in Electronic, Computer Engineering and Telecommunications. Since 2006, he has been involved in both Italian and European research projects.

He is co-author of several reaserch papers, which have been published in proceedings of international conferences or international journals. His main research interests include the Internet traffic classification and characterization, with a particular focus in peer-to-peer applications, the IP mobility issues in wireless networks and pervasive environments and the green networking. Since 2009, he is holder of research grants at the Department of Communication, Computer and System Sciences (DIST) of the University of Genoa.

**Matteo Repetto** received his "laurea" degree cum laude in Telecommunication Engineering in 2000 and the Ph.D. degree in Electronics and Informatics in 2004 from the University of Genoa. Currently, he is a researcher at CNIT, the Italian inter-university consortium for telecommunications.

He has co-authored over 20 scientific publications in international journals and conference proceedings, and he has cooperated in many different national and European research projects in the networking area. His current research interests are in wireless networks, estimation of freeway vehicular traffic, pervasive communications and mobility management.

# Tracking Per-Flow State – Binned Duration Flow Tracking

Brad Whitehead, Chung-Horng Lung
Department of Systems and Computer Engineering
Carleton University
Ottawa, Canada
{bwhitehe, chlung}@sce.carleton.ca

Peter Rabinovitch
Alcatel-Lucent
Ottawa, Canada
peter.rabinovitch@gmail.com

***Abstract*—Recent advances in network monitoring have increasingly focused on obtaining per-flow information, such as flow state. Tracking the state of network flows opens up a new dimension of information gathering for network operators, allowing previously unattainable data to be captured. This paper presents a time efficient novel method – Binned Duration Flow Tracking (BDFT) – of tracking per-flow state by grouping valid flows into "bins". BDFT is intended for high-speed routers where CPU time is crucial. BDFT is time efficient by adopting Bloom filters as the primary data structures. Simulation results show that BDFT can achieve over 99% accuracy on traces of real network traffic.**

***Index Terms*—network monitoring; flow tracking, Bloom filter, high-speed networks**

## I. INTRODUCTION

The continual evolution of the Internet creates a dynamic environment for ISPs to operate their network within. Traffic patterns have changed dramatically in recent years, from a history of a few heavily used protocols, the Internet today contains hundreds. Identifying and managing the transmission of these protocols is critical for ISPs, as better network control leads to better utilization which leads to lower costs.

The operation of many network measurement applications can be abstracted to a requirement to store some amount of state about individual flows. Tracking per-flow state on high-speed routers requires an approach that is customized to the demands of an embedded environment, typically limited memory and processing capability. In this environment, monitoring applications such as NetFlow are not able to scale to the ultra-low per-flow resources available; typically NetFlow requires a minimum of 21 bytes per flow. Packet sampling (e.g., 1 in 20 sampling), on the other hand, normally returns low accuracy. For long duration flows that are also high bandwidth flows, due to the fact that high bandwidth means the probability of them being sampled is high, naive implementation does manage to track some long duration flows. However, the memory usage could be very high [17].

This paper proposes a time and space efficient method of tracking per-flow state by grouping flows into bins called Binned Duration Flow Tracking (BDFT). The "binning" concept was motivated by design tradeoffs that ensure that efficient operation in terms of accuracy, computational resources, and memory resources, can be attained.

Flow state is an abstract concept that is best discussed in terms of an example, in this paper we focus on tracking the duration of flows with the BDFT concept. In brief, BDFT operates by placing individual flows into "bins" which represent the current state of the flow. BDFT assigns time ranges to each state (bin) (e.g., 0-15 sec, 15-45 sec, 45-75 sec, 75-105 sec), and moves flows to the next time range (state) on a periodic basis. BDFT inherits much of its time and space efficiency from the use of counting Bloom filters [3] as the data structure which represents the bins. Symmetric Connection Detection (SCD) [18] (described in Section III) is used for BDFT to pre-filter incomplete connection attempts and reduce the total loading on the methods.

Flow duration is a useful metric in real-world situations. For example, determining the application-level content of a network flow normally requires Deep Packet Inspection (DPI) of critical packets in the flow. Unfortunately, DPI is not available on high-speed routers due to intrinsically high requirements for processing cycles and memory accesses per packet. An alternative to DPI is to classify traffic based on transport level network flow information, such as flow duration, average packet size, and fan in/fan out [10][14].

The rest of the paper is organized as follows: Section II describes the background. Section III highlights SCD. Section IV explains tracking the state with bins. Section V presents basic BDFT. Section VI discusses BDFT extensions. Section VII describes false positive removals. Section VIII demonstrates the experiments and results for memory usage and accuracy. Section IX presents detailed computational analysis of BDFT. Finally, Section X concludes our study.

## II. BACKGROUND

Tracking per-flow state is a relatively new area with little work that is directly related to BDFT. However, the use of


doi:10.4304/jnw.7.1.37-51

Bloom filters and Bloom filter variants such as counting Bloom filters [2] has become wide-spread in network monitoring. The main reason for the popularity of Bloom filters is their ability to provide a time and space efficient data structure to represent a set of items when some errors are acceptable [2]. A good survey of Bloom filters in network applications is [4]. Some other Bloom filter variants are Space-Code Bloom filters [13], and Time-Decaying Bloom filters (TDBF) [7] which can track flow duration with medium accuracy as shown in [17]. Attig and Lockwood have shown that a Bloom filter can be implemented in hardware and can scale to OC-192 (10Gbps) speeds [1].

Bonomi et al. [3] presented methods of tracking the state of network flows. They described the use of Bloom filters and d-left hashing to enable per-flow tracking of state in a network. They presented two variations of a state tracking system using Bloom filters. Their first method uses a single counting Bloom filter to store a set of <flow, state> pairs. Their second approach uses counting Bloom filters to store both a count and a state in each cell corresponding to a flow's hashes. Although both of these approaches rely on Bloom filters, they are significantly different from our approach to per-flow state tracking using “bins”. This paper introduces the concept of using multiple bins (and therefore multiple Bloom filters) to achieve high accuracy with low computational requirements.

Bonomi et al. [3] found that the accuracy of both approaches was low enough to motivate a third method which is based on d-left hashing [5] and fingerprints called Fingerprint-Compressed Filter Approximate Concurrent State Machine (FCF ACSM). Each cell in the d-left hash stores the fingerprint and the state of the flow. FCF ACSM was found to be very accurate and to have good memory efficiency when compared to the Bloom filter based approaches. Its accuracy remains good up to ~80% memory efficiency (depending on table/bucket/cell configuration), after which bucket overloading can become a problem.

FCF ACSM is memory-efficient but has higher computational cost than BDFT. Computational performance is a metric that is rarely analyzed in determining a method's performance in this area. Computational analysis is an important part of the overall performance picture of each method, because CPU time is critical for high-speed routers.

Cuckoo hashing is a combination of multiple-choice hashing with the ability to move elements. Cuckoo hashing requires only a constant number of items to be moved for each insertion, depending on the load of the hash table. However, standard cuckoo hashing suits software applications, not high-speed routers [11]. Cuckoo hashing combined with insertion queue was proposed in [11]. In [12], the authors designed a scheme that allows at most one item to move during insertion, which results in higher space utilization. Comparison with other schemes was conducted and reported in [12]. Both [11] and [12] consider the availability of content addressable memory that allows parallel lookups.

## III. Symmetric Connection Detection

SCD [18] is method of filtering network traffic such that only fully established TCP (the protocol of concern) flows will pass through the filter. To establish a TCP connection a three-way handshake process takes place; each computer sends a SYN, and the initiating computer sends an ACK to complete the connection. Once the ACK is received, the connection process is completed and the TCP session is fully established. Tracking the establishment of a TCP connection therefore requires keeping track of all three states. However, this can be simplified to two states with the following observation. From a point in the middle of the route between the computers the receipt of SYN packets from both sides of a connection implies that both computers can reach each other and want to establish a connection, strongly indicating that the connection will be established with a completing ACK. SCD makes use of this observation and defines an established connection as one where both sides have received a SYN from the other side but not necessarily an ACK. Therefore, SCD processes only TCP SYN packets, or an average of about 1 in 20 packets. In typical Internet traffic, TCP accounts for the most of traffic (could be as high as 95%), of which 5-10% is SYN packets [15].

SCD is used for BDFT to pre-filter incomplete connection attempts and reduce the total loading on the methods. SCD stores the state of all connection attempts and performs a comparison on the connection state to determine when a connection has been established. SCD can report the current connection status in real-time, every time the state of a flow changes. The connection status is reported as a Boolean value; true if the flow is now established, and false if it is not yet established. Connection information can then be used to filter or pass packets for that flow to a higher level monitoring system.

The operation of SCD can be summarized as follows: TCP SYN packets are associated to flow identifiers using two Bloom filters for both space and time efficiency. We employ two Bloom filters, one filter for each SYN direction. Once a TCP SYN has been detected from both sides of a connection, SCD will report that the connection was successfully established. The problem of tracking connection establishment can now be defined as the following question: when a TCP SYN is received from one side of a connection has the other side already sent a SYN? If so, then the connection is established; if not, then store the fact that this side of the connection has sent a SYN. To answer this question, SCD keeps state on all SYN packets that have been sent and the direction that they were sent in. Direction is determined by comparing the source and destination IP addresses, e.g., if source IP is greater than destination IP then the packet is assigned direction 1, and if

source IP is less than destination IP the packet is assigned direction 2.

Network monitoring applications such as tracking the duration of TCP flows can be significantly improved using the pre-filtering provided by SCD to filter out flows which are never fully established. In [18], it was shown that filtering those unsuccessful flows can reduce the processing requirements by 95%. In this paper, SCD is adopted to pre-filter incomplete TCP flows for BDFT. Section IX.A highlights the effect of SCD parameters on accuracy and memory usage.

## IV. Tracking State With Bins

Network operators desire to track the state of network flows to extract additional information about the traffic on their network. This information can be varied, and the required state transitions for a flow can be equally varied. Providing a truly flexible and open definition of the state transitions for a flow requires that the state tracking method be able to update the state of a flow based on arbitrary packets and times. Such a state tracking method can meet any demands placed on it. Flexibility comes with a price, however, and a state tracking implementation which processes every packet and can allow arbitrary state transitions may not be practical on today's router.

Finding an implementable solution for tracking per-flow state is possible if the scope of the problem is narrowed. For some classes of information, such as flow duration, a limited state tracking method may meet both the information needs of a network operator, and be practical for current router hardware. Narrowing the scope of the problem, accepting a less precise state, and trading off accuracy for some errors, are several general ways to gain efficiency in a system. Following these principles leads us to define a class of flow states with the following characteristics:

- Many flows share a common state
- State transitions happen for many flows at the same time
- State transitions are singly-linked

These state characteristics can be intuitively thought of as grouping flows into “bins”. A bin represents a group of flows that are all in the same state. And since all state transitions are unitary and happen at the same time for all flows in a bin, state updates can be performed by simply moving all of the flows in one bin to another. Such a simple model of state transitions creates some advantages for a method which tracks flow state.

The bin grouping allows a potentially large set of flows to be treated as a single entity for operations such as state transitions, flow removal, and information queries. Memory efficiency can also be increased, since all flows in a bin are known to have a specific state there is no need to have a per-flow state indicator. In addition, each bin can be represented using an arbitrary data structure, which allows a designer to choose the best data structure that meets the performance requirements specific to each bin.

Using bins can increase state-tracking efficiency dramatically, but only for some applications. This is the main disadvantage of grouping flows into bins. There is a potentially large array of state machines which cannot be translated into a grouping of bins. For instance, state machines which require processing of every packet in the network and allow per-flow state transitions at any time are not easily translated into a binning approach.

Tracking the duration of flows is one example of state tracking which translates well into a binning approach.

## V. Binned Duration Flow Tracking

BDFT is a method of tracking the duration of network flows on a per-flow basis, for every flow that passes through a network device. This section presents a high-level overview of BDFT, followed by a detailed description of permissible BDFT operations. Section VI discusses the extensions to BDFT which can improve the accuracy of expected BDFT performance.

BDFT is a data structure and algorithm designed to track the approximate duration of all TCP flows seen on a high-speed router. The BDFT-style binning method can also be extended to any network measurement application where minimal state is required, and where the operations required are adding flows, removing flows, and querying for a flow's state. Using BDFT as a classier allows the long duration flows to be aged for further processing by a DPI device to determine if they are P2P traffic or other traffic types.

Compared to NetFlow, BDFT reduces memory requirements in two novel ways. First, the flow identification information is simply not stored in memory. This technique of network monitoring, not storing the flow identification information, is one of the primary contributions of BDFT, and is explained further below. Second, the per-flow state information is stored by splitting the flows into a number of bins, each bin is associated with some state, so all flows in a bin share some common characteristic. Bins are the only data storage component of BDFT. In BDFT, each bin represents an independent and arbitrary length of time, therefore the bin that a flow is in corresponds to its current duration. A bin can be represented and stored using an arbitrary data structure; however selection of an appropriate data structure is a critical design decision. The selected data structure must allow the flow's state to be saved, queried, and removed, without any requirement to store flow identification information.

In BDFT, counting Bloom filters were selected as the default bin data structure. This selection allows the intrinsic operation of Bloom filters to be used to our advantage, by replacing the flow identification information with hashes. This replacement takes advantage of the fact that counting Bloom filters use a number of independent hash functions

which are used to index counters in an array, incrementing them on insert, and decrementing them on delete. When a packet is received the hashes are calculated based on various flow identification information contained in the header. As a result, the hashes are the only information required to index the flow's location in the filter, and all flow identification information is calculated on-the-fly from the current packet and the hash values, not stored in memory.

BDFT's bins are its only data storage component, so the execution of BDFT involves operations that modify, or lookup, the data stored within the bins in response to external inputs. These operations are triggered by TCP packets received in the data path of a router (or other networking device) with the SYN, FIN, or RST flags set. When a TCP SYN packet is received, the corresponding flow is entered into the first bin, and is then automatically aged to successively older bins until the flow is removed when a FIN or RST packet is received. To determine the current or end-point duration of a flow, BDFT determines the bin number that the flow is in, which is then translated into a range of time (or the midpoint of the bin) and returned to the requester. The next sub-sections describe the operations of BDFT in detail.

### A. BDFT Components

Figure 1 shows the basic operations that define how BDFT maintains duration information for all flows. For applications requiring higher performance in terms of accuracy, there are ways to increase the accuracy of BDFT, which is described in Section VI. Many of these enhancements come at very little cost in terms of implementation complexity or computing requirements, and therefore should be employed in most BDFT implementations. The basic operations are depicted as follows.

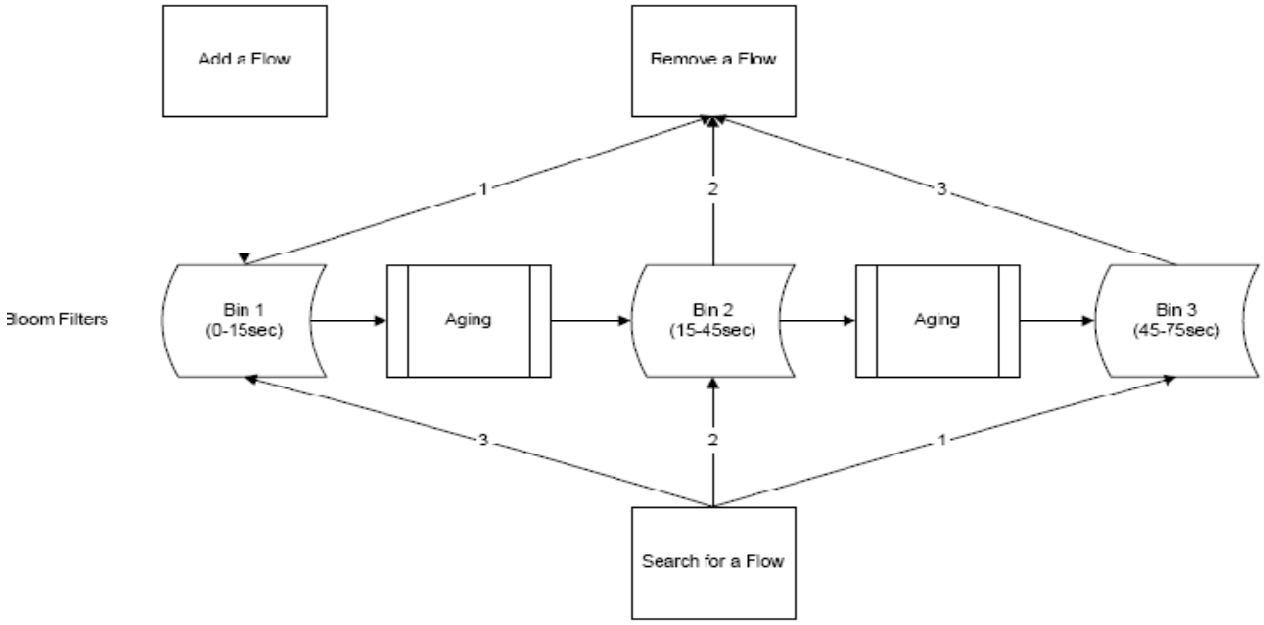



Figure 1. Diagram of BDFT operations

#### 1) Add a Flow

Flows are added to Bin #1 when they enter a "partially established" state, which we define as receipt of a SYN packet from either side of a connection (1st or 2nd step of the TCP 3-way handshake). Flows are added by creating $k$ hashes from the flow identification information, searching all bins to see if the flow already exists; and if not, incrementing the counters in Bin #1 corresponding to those hashes. Searching all bins can be avoided by using SCD pre-filtering as described in Section III, so flows are only added on the 2nd step of the handshake.

When adding a flow to a counting Bloom filter, it is possible that one or more counters are already at their maximum value. In this case, counters which are at the maximum value should be left at the maximum and all other counters incremented. Ideally each flow should only be added once and flows which are never established (e.g. port scans) should not be added. Flows which never complete the establishment phase must be removed from the filter.

#### 2) Remove a Flow

TCP packets containing a FIN or RST flag signal the end of a flow, at which point the flow is removed from its bin. Flows are removed by searching from the shortest-duration bin to the longest. When the flow is found the counters corresponding the flow's hashes are decremented. The counters corresponding to the flow must be decremented

every time a FIN or RST is received, until one of the counters reaches zero. This operation results in an aggressive removal of flows, e.g., some flows may be removed prematurely, due to multiple FIN packets being sent. A solution for multi-removal of flows is presented in Section VI. Bins are searched starting with the youngest based on the observation that 40% to 70% of flows last less than 2 seconds [6], so the flow will most likely to be found in the youngest bin.

*3) Aging*

BDFT maintains its per-bin state by "aging"–the process of moving all flows in a shorter-duration bin to the next longer duration bin. When a bin is in a state where it needs to be aged we say that it is "expired". Each bin represents a time range (duration) for flows. The time range for each bin must be selected based on the accuracy required vs. memory requirements. TABLE I shows an example of BDFT array configuration that uses 180,224 bytes (or 1,441,792 bits) of memory. (There are 720,896 entries in total for this example; each entry has a 2-bit counter, so the total memory size is 720,896×2 = 1,441,792 bits). The expected number of flows in each bin is based on the *C_04* [16] trace flow duration which will be presented in detail in Section VIII.

TABLE I. EXAMPLE BDFT ARRAY CONFIGURATION

| Start Time | End Time | Max Entries | Expected # of Flows | # of Hash | Counter Bits |
|---|---|---|---|---|---|
| 0 | 15 | 131072 | 2125 | 3 | 2 |
| 15 | 45 | 131072 | 1692 | 3 | 2 |
| 45 | 75 | 65536 | 929 | 3 | 2 |
| 75 | 105 | 65536 | 565 | 3 | 2 |
| 105 | 165 | 65536 | 620 | 3 | 2 |
| 165 | 225 | 32768 | 301 | 3 | 2 |
| 225 | 285 | 32768 | 228 | 3 | 2 |
| 285 | 405 | 32768 | 323 | 3 | 2 |
| 405 | 525 | 32768 | 189 | 3 | 2 |
| 525 | 765 | 32768 | 208 | 3 | 2 |
| 765 | 1245 | 32768 | 213 | 3 | 2 |
| 1245 | 2205 | 32768 | 183 | 3 | 2 |
| 2205 | 3165 | 16384 | 51 | 3 | 2 |
| 3165 | 3600 | 16384 | 26 | 3 | 2 |

Note that the number of entries for each bin should be configured significantly higher than the expected number of flows. In addition, the array should be designed so that when the flows are moved to the next longer bin, the next longer bin should have enough entries for those flows. Bin sizes should also be a multiple of a base unit to increase the efficiency of the aging algorithm. For instance, bin 1 (0-15sec) and bin 2 (15-45sec), as illustrated in TABLE I, have the same number of entries (131072), as the expected number of active flows in bin 2 (1692) is more than half of that in bin 1 (2125). The size of bin 3 (65536), however, is only half of that of bin 2, since the expected number of active flows in bin 3 (929) is close to half of that in bin 2 and many flows in bin 2 will terminate due to the fact that most flows only last for a short duration. The next subsection discusses bin time ranges and sizes in details.

As time advances during the operation of BDFT, the flows in a bin become older, until the oldest flow in the bin is older than the time range of the bin, at which point the bin must be aged. The aging process is the key to BDFT; it allows the maintenance of the state for all flows. By keeping flows in counting Bloom filters, and aging the filters in time, no flow-specific information such as flow start time needs to be kept. When a bin is expired, the flows that are currently in the bin are moved to the next longer duration bin, as shown in Figure 2.

*4) Search for a flow*

Searches are performed by an external agent submitting flow identification information which can be used to generate the *k* hashes. Searches are performed starting with the oldest bin first, and moving to sequentially younger bins, until a bin is found where all counters are greater than zero corresponding to flow's hashes, at which point the flow is "found" (or a false positive was found). The reason is based on the assumption that the longer duration bins are the least likely to generate a false positive. The duration for the flow is an estimate calculated by determining the midpoint time for the bin the flow was found in. For example, if the flow is in a bin with a time range of 45-75 sec, a duration of 60 seconds would be returned.

Figure 3 depicts an example of the life of a flow as it is inserted, aged, and removed from BDFT. In this example, BDFT has bins with time ranges of 0-15 sec, 15-45 sec, 45-75 sec, and 75-135 sec, and the flow lasts for 55 seconds. The flow arrives just after Bin 1 was aged, and therefore the flow will be in Bin 1 for its full 15-second duration.

During normal BDFT operation, the timer to expire Bin 1 is always running, and therefore, for example, a flow could arrive when Bin 1 is just about to expire or at any other time. In Figure 3, the "Expire Bins" box denotes the continuous process of checking all bins to see if they need to be expired. The following steps illustrate the BDFT operations for this example:

- The new flow arrives; its hashes are calculated based on IP Src/Dst, Port Src/Dst, and protocol type
- The flow is added to Bin 1 by incrementing the counters corresponding to the hashes
- After 15 seconds Bin 1 expires and its flows are moved to Bin 2
- After an additional 30 seconds Bin 2 expires and its flows are moved to Bin 3
- After 55 seconds from the flow start, a TCP FIN is received for the flow, and the removal process begins
- The flow's hashes are calculated as above
- The Bins are searched for the flow's hashes starting with Bin 1
- The flow is found in Bin 3, so the counters corresponding to the hashes are decremented in Bin 3

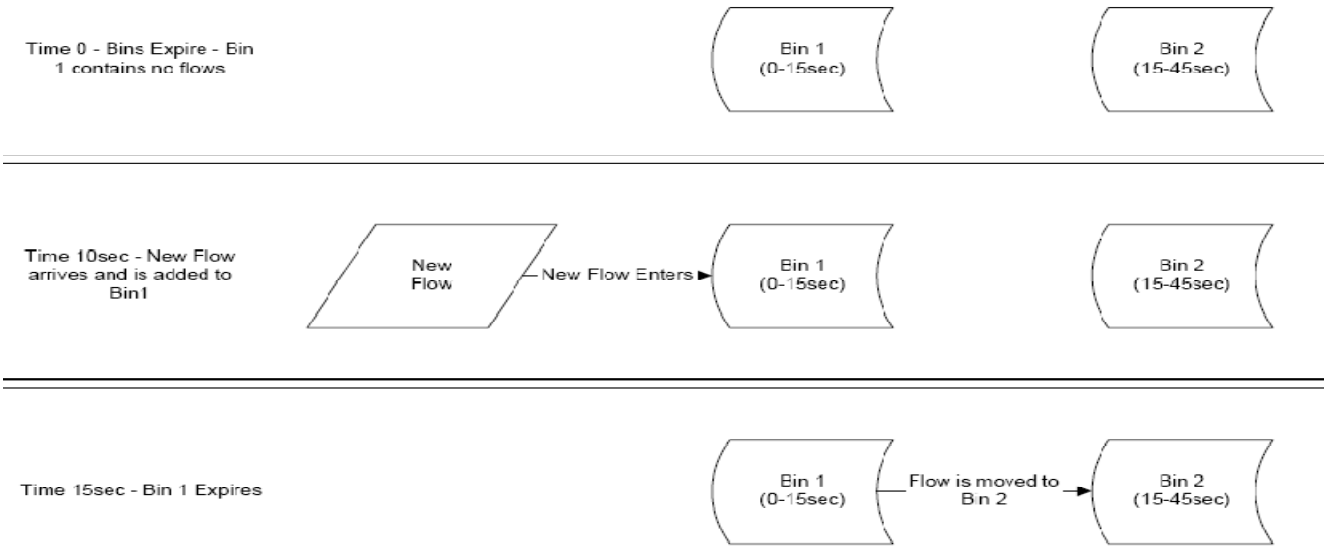



Figure 2. BDFT aging process

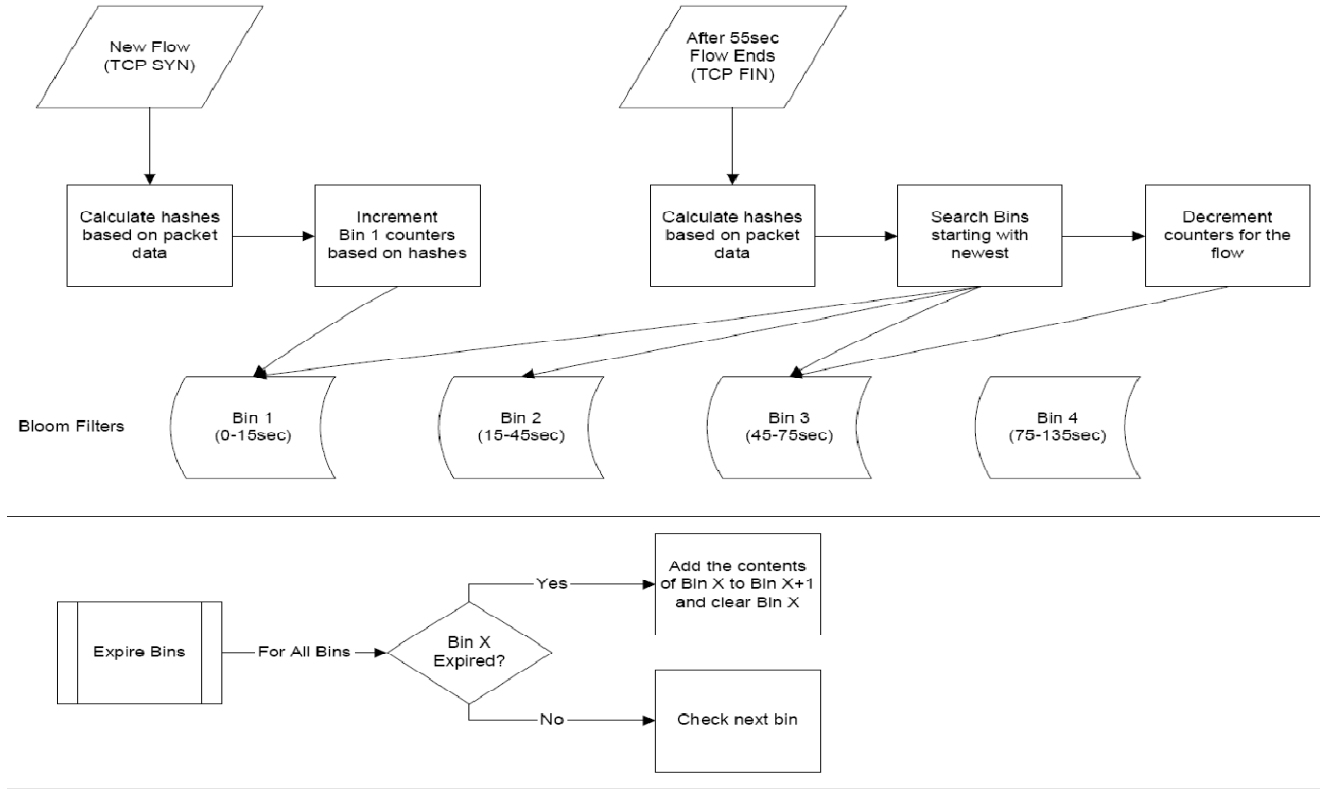



Figure 3. Flow chart of the life of a flow in BDFT

### B. BDFT Parameters

Design and analysis of BDFT involves specific design goals and several parameters which affect algorithm performance. TABLE I shown in the previous sub-section presents an example of BDFT configuration. The key parameters include:

- m – The size of the Bloom filter (in total entries).
- n – The expected number of entries in the filter. Also, the number of items in the set.
- k – The number of independent hash functions per Bloom filter.
- c – The maximum count in an entry of a counting Bloom filter.

In [4], detailed analyses of the Bloom filter were systematically conducted. We have also followed the fundamentals of [4] and conducted thorough mathematical analyses on the expected performance of BDFT based on each of the parameters, taken independently of the others. Independent analysis of BDFT parameters leads us to present a series of basic heuristics for choosing a full set of BDFT parameters, given the goals of the algorithm designer. We choose a heuristic approach due to the BDFT's ability to adapt to a wide range of design goals and network traffic characteristics, based on the decisions of the designer.

Some typical design goals include accuracy (or error probability), probability of a false positive, expected number of overflowed counters for the counting Bloom filter, processor and memory usage. Different design goals and traffic characteristics will affect the selection of BDFT parameters. For instance, if the ratio of the number of entries to maximum entries (n/m) is high, then the probability of false positive will increase. Formally, the probability of a false positive is also determined by the probability of selecting k bits that are set to true, as presented in [4]:

$$\mathrm{P[False\ Positive]} = \left(1 - \left(1 - \frac{1}{m}\right)^{nk}\right)^{k}$$

Figure 4 illustrates the search error probability of a Bloom filter when the ratio of entries to total entries (n/m) and the number of hash functions are varied.

The size of the counter for the counting Bloom filter is another example to consider in parameter selection. The standard Bloom filter with one bit (true/false) per entry does not support removal of items without generating false negatives, see Section VI for details. Counter overflows contributes to false negatives which is a critical issue for accuracy. The expected number of overloaded counters can be calculated as follows. Each entry in a Bloom filter can be considered independently when determining the probability that it will be incremented. To cause an overflow the counter must be selected c+1 times out of n inserts. We are interested in the probability of any overflow of the counter, as counter can overflow multiple times, defined as c+1, c+2, c+3 .... The probability of having x matches in n inserts follows the binomial distribution if the counts are assumed to be independent. For a given maximum count c, the probability of overflowing a single counter/entry is given by [4][8]:

$$\mathrm{P[Counter\ Overflow]} = 1 - \sum_{x=0}^{c} \left[ \binom{n}{x} \bullet \left(\frac{k}{m}\right)^{x} \bullet \left(1 - \frac{k}{m}\right)^{n\ \ x} \right]$$

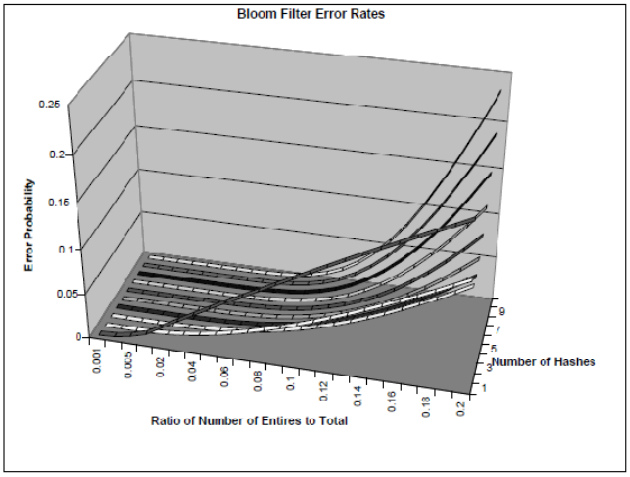



Figure 4. Probability of a false positive w.r.t. varying n/m ratios and the number of hash functions

The number of expected overflows in a filter is then m× P[Counter Overflow]. The probability of counter overflow could be extremely small with 3-bit or 4-bit counters. For instance, the probability of counter overflow could be dramatically decreased ($10^{-5}$ ×) from 2-bit to 3-bit counters and even down to ~ $10^{-15}$ with 4-bit counters [8]. For our experiments, if 3-bit or 4-bit wide counters are used, counter overflows could almost be eliminated entirely.

Bin time ranges and sizes are one of the most complex decisions when designing a BDFT array. This is also a flexible aspect of the design, as BDFT can be customized to give fine-grained or coarse grained feedback on the duration of flows. TABLE I is an example of a fine-grained configuration. A coarse-grained design, for example, could consist of only 3 bins with time ranges of 0-20 sec, 20-110 sec, and 110-3610 sec. The sizing of the bins depends on the flow duration distribution that is expected in the network. The distribution does not have to be exact, but should provide a general guideline for the number of flows expected in each bin.

For example, the flow duration distribution described in Section VIII shows that 75% of valid flows in a traffic race last less than 15 seconds. Therefore, for a network where 100,000 valid flows are expected to be active at any time, a maximum of 75,000 flows would be expected to be stored in a bin containing flows from 0-15 seconds. This bin would need to be sized accordingly to reduce expected error rates. Higher duration bins typically have fewer flows stored, and therefore can be sized smaller to reduce memory usage. Bin sizes should also be a multiple of a base size to increase the efficiency of the aging algorithm.

As a general guideline, to design a BDFT array with a high search accuracy means that the long duration bins should have a search accuracy of at least 99.9+%. The number of active flows in each bin can be estimated from the flow duration distribution. We determine the number of flows in each bin according to the following assumptions:

- There are 1,000,000 terminating flows each hour.
- Flows arrive at a uniform rate.
- The flow duration distribution is uniformly distributed within a bin.

From assumption 3, the average termination time of flows in a bin is half the bin duration (note that assumption 3 is worst-case, in normal traffic the distribution is skewed towards shorter duration flows, resulting in fewer flows in the bin). The number of flows that terminate in a given bin within a certain time can be directly read from the flow duration distribution. Also the flows that are longer duration than the current bin, but are passing through the bin must be taken into account. Given the above assumptions, the number of flows active in bin b is given by Little's law, where the Average Time in Bin is half of the bin duration:

Flows Ending in Bin = Average Time in Bin × Arrival Rate (*ending in bin*)

Arrival Rate (ending in bin) = (1,000,000 flows/hour × P[*Flow Terminating in the Bin*]) / 3600 seconds/hour

Accounting for the contributions of longer duration flows to the flows in the bin requires summing all the flows which are in the bin but will not terminate here, and instead are moving to higher duration bins. Little's equation still applies, but the Average Time in Bin is now the full bin duration. Let n represent the number of bins and c be the current bin;

Arrival Rate (not ending in bin) = (1,000,000 flows/hour × $\sum_{i=c}^{n} P[Flow\ ends\ in\ Bin\ i]$ ) / 3600 seconds/hour

TABLE I shows an example configuration of bin time ranges which match the requirements of the fine-grained design. The number of active flows expected in each bin was calculated using the abovementioned three equations. Once the bin time ranges and number of active flows in each bin has been determined, the bins can be sized. Heuristics for sizing the bins are as follows:

- A short duration (0-15sec) medium accuracy bin (95%) should be used to filter the large number of very short duration flows.
- The short duration bins will not be searched in many cases, and therefore can be designed to have a higher false probability error rate in order to save on memory.
- The long duration bins should have high accuracy (99.9%) for good search accuracy.
- The number of flows normally decreases logarithmically with the duration, so longer duration bins should cover more time and be reduced in size.
- The binning process introduces inaccuracy when the actual duration of the flow is not the average duration of the bin, so longer duration bins should cover more time, and the shorter duration bins less time. This keeps the relative inaccuracy involved in the binning process to a minimum.

In summary, parameter selection depends on various factors, particularly resource constraints and design goals. There are different possible configurations with respect to various scenarios and design goals. Parameter selection heuristics for Bloom filters and counting Bloom filters have been investigated by researchers [3][4][8][17]. This paper focuses on bin time ranges and sizing. Section VIII and IX present some experimental and computational analyses for two specific traffic traces vis-à-vis accuracy, overflow errors, and memory usage.

## VI. BDFT EXTENSIONS

The standard BDFT as described in Section V will provide a basic level of performance suitable for some applications. For applications requiring higher performance in terms of accuracy there are various ways to increase the accuracy of BDFT substantially. This section presents several enhancements to both the accuracy and computational requirements of BDFT. Many of these enhancements come at very little cost in terms of implementation complexity or computing requirements, and therefore should be employed in most BDFT implementations.

### *A. Enhanced Insertion and Removal*

The accuracy and computational performance of BDFT can be improved if it can be guaranteed that for each flow received, the insert and remove functions will be executed once and only once. Relying on SYN/FIN/RST directly can lead to an imbalance in BDFT due to timeouts and lost packets. Since BDFT increments counters on SYN packets, and decrements them on FIN/RST, this imbalance can cause incomplete removal of flows (SYN > FIN/RST) or potential removal of multiple flows which share hashes by setting one of the shared counters to zero (FIN/RST > SYN). Balancing the number of SYN packets vs. the number of FIN/RST packets can be accomplished using pre-filtering mechanisms such as Symmetric Connection Detection (SCD) [17][18]. SCD can be employed to ensure that there is only one insert per flow, so using SCD also has the additional benefit of reducing counter overflows. However, balancing SYN vs. FIN/RST also requires that only one removal notification be sent to BDFT per flow, regardless of the number of actual FIN/RST packets. The functionality required for FIN/RST can also be achieved using SCD or a simple variant. A Bloom filter could be employed to track all of the flows which have already sent a FIN or RST, by adding the flow to the filter and notifying BDFT of the removal if the flow is not already in the filter. BDFT's accuracy can be increased substantially using these techniques, therefore the small

incremental memory cost to implement pre-filtering such as SCD in conjunction with BDFT is a good design tradeoff.

### B. TCP Timeouts - no FIN or RST

BDFT relies on the receipt of FIN or RST packets to signal the end of a TCP connection and remove the flow from its present bin. However, it is possible for a TCP timeout to occur such that no FIN or RST packet is ever sent on the connection. For example, when an Internet connection goes down, or a route changes, packets no longer reach the router running BDFT. In this case the flow is "hung" in BDFT and will never be removed. Eventually hung flows overwhelm the long duration bins resulting in a dramatic loss in accuracy for both long duration and short duration flows. As a result, timeouts must be accounted for in environments where they are possible, typically about 0.1% of Internet flows result in a timeout (see TABLE IV).

## VII. FALSE POSITIVE REMOVALS

A false positive removal (FPR) occurs when a false positive leads to the removal of a flow (that may or may not exist) instead of the correct flow. This problem can affect Bloom filters when multiple filters are searched for removal. The mechanism that leads to a FPR is quite similar for both hash table types, but in both cases the sequence of events is fairly complex. For this reason, to the best of our knowledge, the FPR problem has not been identified before. The sequence of events which leads to a FPR is best explained by examples.

In the case when multiple Bloom filters are searched to find the flow to remove (as in BDFT) a FPR is a serious error that can lead to multiple flows being "orphaned" in the filter(s). For example, a flow with hashes {1, 2, 3} is added to Bin 0. The flows in Bin 0 are then aged to Bin 1 (so Bin 1 now contains flow {1, 2, 3}). Now two flows are added to Bin 0 with hashes {1, 2, 4} and {3, 5, 6}. At this point a search performed on Bin 0 for flow {1, 2, 3} will result in a false positive. If flow {1, 2, 3} ends at this point it will be identified for removal from Bin 0 due to the false positive, and therefore this is a false positive removal. The result would be two orphaned flows in Bin 0 and one in Bin 1. In Bin 0 the flows would now be {-, -, 4} and {-, 5, 6}, and in Bin 1 the flow would still be there as {1, 2, 3}.

These orphans will likely never be removed from the filters, and end up polluting the filters. When flows {1, 2, 4} and {3, 5, 6} are removed they will likely result in false negatives, unless they end up creating a FPR in other bins, which can compound the problem.

FPR is a serious problem. This paper does not deal with FPR. FPR could make a valuable area for future work.

## VIII. EXPERIMENTAL ANALYSIS

### A. Trace Characteristics

We obtained traces of Internet traffic from the well-known networking research organizations CAIDA [16] and NLANR [9], with the two traces hereafter referred to as *C_04* (CAIDA) and *N_12* (NLANR). These two traces are representative of the diverse extremes of Internet traffic. This section highlights the main intrinsic characteristics of these two traces. In general, the *C_04* trace represents normal "dirty" public backbone Internet traffic, with many packets being invalid attempts at port scanning or DDoS attacks. The second trace, *N_12*, represents the other end of the traffic spectrum from *C_04*, being fairly "clean" and containing a low number of active flows and very little or no attack and port scanning traffic.

TABLE II. TRACE CHARACTERISTICS

| | **NLANR 2003-12 (*N_12*)** | **As a % of total** | **CAIDA 2003-04 (*C_04*)** | **As a % of total** |
|---|---|---|---|---|
| Total Packets | 196,956,306 | | 202,510,985 | |
| Total TCP Packets | 56,992,573 | 28.94% | 175,418,691 | 86.62% |
| Total Bytes | 46,472,308,705 | | 95,944,872,321 | |
| Total TCP Bytes | 41,482,633,988 | 89.26% | 91,766,946,651 | 95.65% |
| Avg. Bandwidth | 98.48 Mbps | | 203.33 Mbps | |
| Avg. Bytes Per TCP Pkt. | 727.86 | | 523.13 | |
| Duration | 3600 seconds | | 3600 seconds | |

Both traces represent one hour of Internet traffic. These traces were selected to demonstrate performance over long periods of time TABLE II demonstrates the basic characteristics of each trace. The average bandwidth in both traces is roughly similar at 100Mbps and 200Mbps, sufficient to demonstrate performance on high-speed links. The total number of packets is similar; however, the number of TCP packets is over three times higher in the *C_04* trace.

TABLE III. TRACE CHARACTERISTICS FOR TCP CONTROL PACKETS

| | ***N_12*** | **As a % of total** | ***C_04*** | **As a % of total** |
|---|---|---|---|---|
| Total | 196.9M | | 202.5M | |
| SYN | 732,075 | 0.37% | 15,608,680 | 7.71% |
| FIN | 586,000 | 0.30% | 6,084,826 | 3.00% |
| RST | 52,628 | 0.03% | 3,914,433 | 1.93% |

TABLE III shows significant differences between the *C_04* and *N_12* traces. The percentage of total packets which are TCP packets with one of the main control flags (SYN, FIN, RST) turned on is over ten times lower in the *N_12* trace. Another difference is the percentage of RST packets in the *C_04* trace is over sixty times higher than the *N_12* trace. Given that RST packets typically indicate abnormal connection termination, this large difference indicates that the *C_04* trace has many connections that do not follow normal TCP rules, such as port scanning or DoS attack traffic.

TABLE IV. TRACE CHARACTERISTICS FOR TCP 5-TUPLE FLOWS

| | *N_12* | As a % of total | *C_04* | As a % of total |
|---|---|---|---|---|
| Total Flows | 352,410 | | 11,215,873 | |
| Total Established Flows | 274,473 | 77.88% | 555,927 | 4.96% |
| Ave. Active Flows | 11,284 | | 901,245 | |
| Timed Out Flows | 430 | 0.16% | 4376 | 0.78% |
| Unique IPs | 97,036 | | 2,681,172 | |

TABLE IV is a fine-grained look at the flows in each of the traces, and highlights the substantial differences between the two traces. The large percentage of SYN packets in the *C_04* trance can now be confirmed to be related to the large number of incomplete flows, over 95% in this case. Also, due to the large number of incomplete connection attempts (likely port scanning), there are a very high number of active flows on average in the *C_04* trace compared to the *N_12* trace. For these reasons, the *C_04* trace is considered to be a good example of "dirty" Internet traffic and the N_12 trace is a good example of "clean" traffic (with almost 80% of flows being valid in the *N_12* trace).

The distribution of flow durations is an important factor in the design of a BDFT array, as shown in Section V. The duration distribution allows an estimation of the required size of the bins in BDFT, and therefore an estimation of the total memory usage of the BDFT array. One of the few papers to discuss the number flows that last specific lengths of time is "Dragonflies and Tortoises" [6], where they find that 40% to70% of flows last less than two seconds.

Figure 5 shows the flow duration distribution for the *N_12* trace. Only flows that are fully established within the trace are counted. This distribution shows that 75% of fully established flows are less two seconds long. This duration distribution is characterized by the sharp falloff in the number of flows as the duration increases.

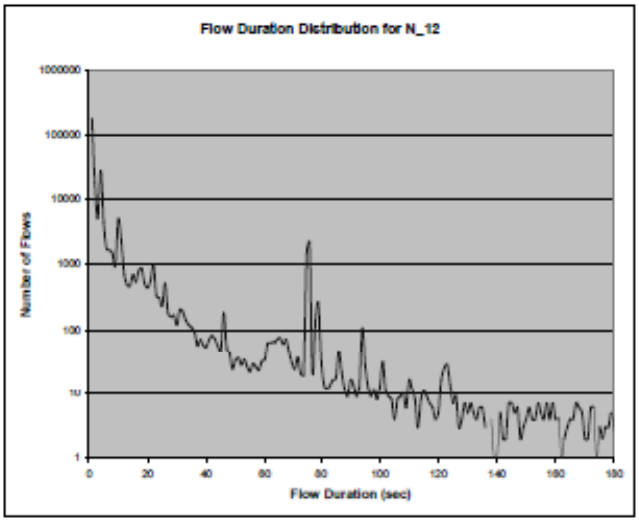



Figure 5. Flow duration distribution for trace N_12

Figure 6 reveals the flow duration distribution for the *C_04* trace. Like the *N_12* trace, this distribution has a large number of flows that last less than two seconds, 50% in this case.

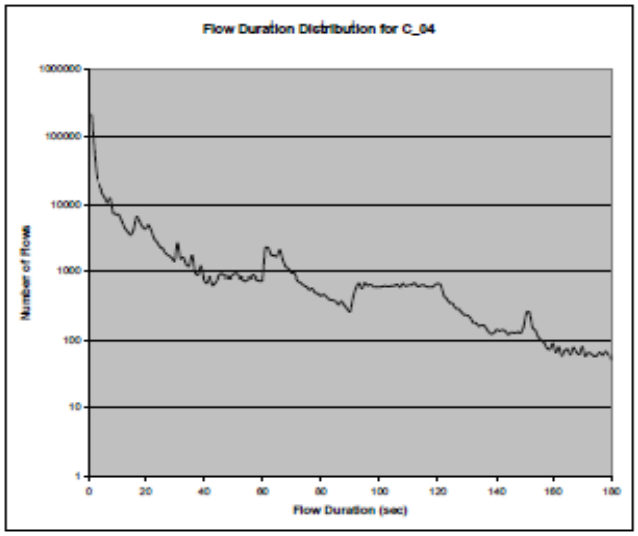



Figure 6. Flow duration distribution for trace *C_04*

## B. Experimental Setup

To determine the accuracy of BDFT, we implemented an experimental framework that keeps track of the duration recorded by a perfect flow tracker with the estimated flow duration reported. The perfect flow tracker was implemented using standard per-flow tracking and measurement techniques. We defined a flow to be the standard 5-tuple of IP source and destination address, TCP source and destination port, and protocol type (TCP only).

As mentioned earlier in Section VI, packets from the trace are also passed to a SCD filter [17][18] to BDFT as well as a simple FIN/RST-checking Bloom filter. The SCD filter was sized large enough to be nearly 100% accurate to eliminate any affect on the performance results from SCD errors. Mitigating timeout effects is still an open area of research, so for our implementation, we automatically remove flows after two minutes with no activity, as adopted in TCP standard.

To present the accuracy of tracking duration with bins, we adopt the following definition of a success. A tracking success is defined to be an estimated flow duration result that is within 50% of the actual flow duration, for flows that are greater than 30 seconds in duration. This definition is based on the design goals of BDFT; it is intended to track long-duration flows to an approximate duration rather than an exact duration. In addition, the duration of actual flows and the duration of bins are not exact, depending on when the flows actually arrive, as explained in BDFT components. Take Figure 3 as a concrete example: if a flow arrives *just after* Bin 1 was aged, the flow will be in Bin 1 for its full 15-second duration. On the other hand, if the flow

arrives *just before* Bin 1 ages, the flow will be moved to Bin 2. Therefore, a tracking success is defined to be the estimated flow duration is within 50% of the actual flow duration. Tracking failures occur if a false negative or don't know is returned, or the duration is outside the acceptable range (mostly caused by a false positive).

The design of the BDFT array is presented in TABLE I for fine-grained tracking of flow duration. The basic bin sizing allocates 128k (128×1024 = 131,072) entries for the first bin, which we refer to as the base filter size. The base size uses 180,224 bytes of memory. See TABLE I for the table configuration and the explanation. We tested BDFT at multiples of the base filter size (and therefore the same multiple of all the other filter sizes) to analyze the performance of BDFT at various memory usage levels. Note that we deliberately use less than ideal number of hashes for the expected loads on the Bloom filters and also use smaller counters than typically suggested. We find in simulation that these choices generate greater *overall* results.

### C. Experimental Results

TABLE V shows the performance of BDFT with various memory configurations. A critical factor for a flow tracking method is the memory used to achieve a desired performance level and the computational requirements of the method. BDFT was able to achieve practical accuracy performance at memory usage levels low enough to be practical for implementation on high-speed routers.

TABLE V. BDFT PERFORMANCE

| Trace | Memory Usage (bytes) | Accuracy |
|---|---|---|
| *C_04* | 90112 | 95.46% |
| *C_04* | 180224 | 99.19% |
| *C_04* | 360448 | 99.87% |
| *C_04* | 720896 | 99.97% |
| *N_12* | 2816 | 96.85% |
| *N_12* | 5632 | 99.79% |
| *N_12* | 11264 | 99.98% |

BDFT has two issues that can lead to buildup of orphan and stale flows in the tables. These issues are false positive removal (FPR) and flow timeouts. For our simulations, the effect of these issues is removed by accurately correcting for FPR and removing flows from the filters that have not seen activity for two minutes in the traffic traces. These corrections provide increased insight into the mechanisms that affect performance for a flow tracking approach. Without the correction, the accuracy reduces significantly for smaller memory usage. As an example, results for BDFT that are not corrected for FPR, but with timeout correction, are provided in TABLE VI.

TABLE VI. BDFT PERFOMRANCE WITH FPR FOR C_04

| Memory Usage (bytes) | # of Overflows | Accuracy |
|---|---|---|
| 90112 | 902 | 79.24% |
| 180224 | 134 | 96.15% |
| 360448 | 16 | 99.59% |
| 720896 | 1 | 99.98% |

Also, note that a simple way to avoid FPR errors is to oversize the tables such that the chance of false positives is small. The use of SCD as a pre-filter reduces the number of flows that are timeouts by 95% on average [18].

BDFT accuracy results are in-line with our expectations. With only 0.257 bits of storage required per flow BDFT achieves 99.87% accuracy on the *C_04* trace, and with only 0.128 bits of storage required per flow BDFT achieves 99.79% accuracy on the *N_12* trace. Bits-per-flow is calculated by dividing the memory usage by the number of established flows. For instance, the third entry in TABLE V has 360,448 bytes (or 360,448×8 bits) for *C_04* at 99.87% accuracy and the total number of flows for *C_04* is 11,215,873 as shown in TABLE IV. Bits-per-flow, therefore, is (360,448×8)/11,215,873 = 0.257 bits. These results also demonstrate an important point; in practice it is not necessary to use 4 bits per counter in a counting Bloom filter, which has been the standard number; in these simulations 2 bits per counter was sufficient to achieve excellent accuracy. However, it should be noted that the primary failure mechanism in overload conditions (e.g., the entry having 90112 bytes for *C_04*) is counter overflow (902 overflows as shown in TABLE VI). These results confirm that BDFT is a good choice to track flow state when CPU time is more expensive than memory.

## IX. COMPUTATIONAL RESULTS AND ANALYSIS

Computational cost plays a crucial role for monitoring of high-speed networks. This section first briefly presents the memory and computational efficiency of SCD. A comparison of BDFT with two other flow duration strategies is then conducted. Third, the computational cost of BDFT is further discussed.

### A. SCD Performance Cost

To perform a computational evaluation of BDFT, the memory usage and computational overhead of SCD are considered, as SCD is used to determine if flows are successfully established. In terms of memory usage, with only 32k bytes, SCD can achieve 99.998% accuracy for the *N_12* trace and 99%+ for the *C_04* trace. By increasing memory usage to 512k, over 99.9% accuracy can be achieved for the *C_04* trace [18]. As demonstrated in VIII.A, the *C_04* trace is an example of a good worst-case test of SCD, as it has high invalid flow ratio combined with the high number of active flows. The *N_12* trace represents the other end of the traffic spectrum, containing a low number of active traces and very little or no attack and port scanning traffic, thus making this trace an example of the best-case performance of SCD. As a comparison of memory usage, the per-flow tracker used 24MB of memory to store about one million flows, or forty-eight times more memory than the 99.9% accurate SCD.

In [18], it also demonstrated the computational efficiency of SCD, which was about 172 seconds to process the 3600-second *C_04* trace. The execution time was an 11× reduction compared to the naïve flow tracking approach.

### B. Other Flow Duration Strategies

The accuracy of BDFT is a clear advantage. However, it is possible that another simpler implementation could achieve similar accuracy with less complexity. For this reason we selected two other strategies to track flow duration and compared the results with BDFT. In this section, we describe the operation of the two competing duration tracking strategies.

To maintain similarity in the comparisons the memory usage and sampling rate were balanced between all three strategies, BDFT, Naïve Sampled, and TDBF [7] (see section II).

#### 1) Naïve Sampled

A simple approach to tracking flow duration is based on the sampling idea used by NetFlow. NetFlow operates by keeping a flow record for every flow that it sees in the network. When a packet arrives, its corresponding flow record is looked up and the appropriate statistics and counters updated. If no flow record exists for the flow then a new one is created. Flow records are removed if no packets are received from that flow for a short period of time, typically 15 seconds. Likewise, flows records are also expired once they reach a certain age, typically 15-30 minutes. When a flow record expires it is exported to an external collector computer for further processing.

The pure naive idea to track flow durations is to process every packet and keep a record of every flow in the network. As part of each flow record timestamps could be maintained for the first and last packet in the flow, and the flow records could be expired after a long interval. Unfortunately this approach would share the same scaling problems as basic NetFlow. So in our implementation of a naïve duration tracker, we make several sacrifices to the overall accuracy to obtain time and space performance that is comparable to BDFT.

For our naïve duration tracker implementation the tracker processes packets based on a sampling rate of one in twenty, which matches BDFT's effective sampling rate. The sampled packets are processed in a similar way to NetFlow. A flow record table is maintained which contains entries consisting of a flow identifier and the start and last packet timestamps. When a packet is sampled the table is searched for the flow and the timestamps updated if the flow is already in the table, or the flow is added to the table. If the table is full then the oldest flow in the table is expired, meaning it is removed from the table.

Accuracy results for the naïve implementation are determined by searching the flow table every time a flow terminates. The estimated duration for the flow returned from the naïve tracker is compared with the actual time using the same comparison methods used for BDFT.

#### 2) Time-Decaying Bloom Filters (TDBF)

The second approach to tracking the duration of flows involves the use of a single Time-Decaying Bloom Filter [7]. A TDBF allows the multiplicity of an item in a set to be tracked, with more recent insertions being weighted higher.

We can modify the basic operation of the TDBF slightly to track the length of time that an item has been in a set. The basic idea is to initialize all counters corresponding to an item's hashes to a large initial value, e.g., 40000. Then, in our implementation, every ten seconds all counters are decremented. When a query is performed to determine how long the item has been in the set the duration is equal to (40000 - current counter value) × 10 seconds. The detailed operation of the modified TDBF can be explained through an explanation of its main operations, insert, search, and removal.

To insert a flow into the TDBF all counters corresponding to the items hashes are set to their maximum value according to how many bits are available in a counter. For our implementation we used 16 bit counters for a maximum value of 65535. Counters are set to the maximum value regardless of their current value.

To remove a flow from the TDBF all of the corresponding counters are set to 0. To determine the duration of a flow the counters corresponding to the flows hashes are read and the lowest value that is not 0 is selected. The duration can be calculated by the formula (65535 - lowest counter value) × 10 seconds. This formula is based on the fact that we decrement all counters every 10 seconds. If all counters are 0 then a failure to find the flow is reported. It is important to note that choosing the lowest counter that is not 0 breaks the fundamental operation of Bloom filters where a 0 would normally indicate that the item is not in the filter. Due to the nature of network traffic and assuming that queries will only be performed for flow IDs that are guaranteed to be in the filter we can assume that if any counters are not 0, then the flow has not been removed. This optimization in our specific case results in a 100% improvement in accuracy. If the above assumption cannot be met, then flows which have one or more counters set to 0 cannot be assumed to be in the network and must be returned as a lookup failure.

In terms of complexity when compared to BDFT, TDBF is somewhat simpler due to the fact that there is only a single Bloom filter instead of one per bin.

#### 3) Performance Comparison

TABLES VII and VIII show the BDFT, TDBF, and Naïve results for comparison. It can be seen that BDFT outperforms both Naïve and TDBF in both traces. With only 0.257 bits of storage required per flow, BDFT achieves 99.59% accuracy on the *C_04* trace, and with only 0.128 bits of storage required per flow, BDFT achieves 99.55% accuracy on the *N_12* trace. The numbers of established flows in an hour are 352,410 for *N_12* and 11,215,873 for *C_04*, respectively, as shown in TABLE IV. The number of

overflowed counters column is higher than the theoretical expectations derived in Section V.B. This is due to real-world factors not taken into account during the theoretical analysis, such as timed-out flows, and errors introduced by the SCD filtering.

The flows which timeout end up "polluting" the higher duration bins with many dead flows. In our case, these bins were not designed to handle the increased number of flows due to the timeouts, and therefore can become overloaded to the point of having overflowed counters. This problem can be easily corrected by increasing the counter width from 2 bits to 3 bits. TABLE IV shows the number of timed out flows for each trace.

TABLE VII. PERFORMANCE COMPARISON FOR *C_04*

| Algorithm | Memory (bytes) | # of Overflowed Counters | Accuracy |
|---|---|---|---|
| BDFT | 90112 | 902 | 79.24% |
| BDFT | 180224 | 134 | 96.15% |
| BDFT | 360448 | 16 | 99.59% |
| BDFT | 720896 | 1 | 99.98% |
| Naïve (1 in 20) | 524288 | N/A | 6.07% |
| Naïve (1 in 100) | 524288 | N/A | 1.58% |
| TDBF | 131072 | N/A | 58.79% |

TABLE VIII. PERFORMANCE COMPARISON FOR *N_12*

| Algorithm | Memory (bytes) | # of Overflowed Counters | Accuracy |
|---|---|---|---|
| BDFT | 1408 | 912 | 38.43% |
| BDFT | 2816 | 136 | 82.23% |
| BDFT | 5632 | 17 | 99.55% |
| BDFT | 11264 | 1 | 99.97% |
| BDFT | 22528 | 0 | 100.00% |
| Naïve (1 in 20) | 524288 | N/A | 8.34% |
| Naïve (1 in 100) | 524288 | N/A | 3.82% |
| TDBF | 131072 | N/A | 58.79% |

The significant difference in per-flow memory requirements for the two traces is caused by the differences in the flow duration distribution for the two flows. Referring back to Figures 5 and 6, we observe that the distribution for the *N_12* trace is tightly concentrated in the 0-4 second range, whereas *C_04* is relatively evenly distributed to higher durations. When the flow durations are concentrated such that most flows end within one or two seconds, the overall loading on the first and therefore subsequent bins is greatly reduced. In this situation flows are added to the first bin and then quickly removed so the number of flows actually stored in the bin at any time is quite low, resulting in lower memory requirements.

TABLES VII and VIII also illustrate two important additional points. BDFT is able to achieve 100% (or near 100%) accuracy when the available memory is increased to the point where almost all counter overflows are eliminated. Therefore, even though BDFT is an approximate method of tracking flow duration, it is applicable in situations where close to 100% accuracy is required if sufficient memory is available. The memory requirements for achieving 100% or close to 100% accuracy were 10.37 bits/flow for the *C_04* trace, and 0.657 bits/flow for the *N_12* trace.

### *C. BDFT Performance Analysis*

In general, there is a lack of thorough computational analysis for this area in the literature. Complexity analysis of algorithms is a commonly used method to analyze performance and scalability. But BDFT scales well with respect to the number of flows; this measure does not give much insight into its performance. Instead, this section presents an analysis for the computational cost of each BDFT operation. The analysis could be used as a baseline for comparing with other methods in the future.

Several assumptions about the environment and implementation can also be made. First, we assume that BDFT is implemented in software, though it can be implemented in hardware. Next, assume that they are executed in a single thread on a processor that performs sequential memory access only. Third, ignore the effects of caching.

For BDFT, it is assumed that 3 hash functions are used and counters are sized large enough that they do not overflow. Note that we deliberately choose a number of hash functions less than optimal to save on the number of memory accesses (the lower number of hash functions reduces the probability of a counter overflow, so smaller counters can be used). The following explores the operations of insert, remove, search, and age. Results of those operations are summarized in TABLE IX.

TABLE IX. EXPECTED COMPUTATIONAL PERFORMANCE

| Operation | Memory Reads | Memory Writes | Branches | Total |
|---|---|---|---|---|
| Insert | 3 | 3 | 3 | 9 |
| Removal | 6 | 3 | 6 | 15 |
| Search (rare) | 21 | 0 | 21 | 42 |
| Aging (periodic) | 2000 | 1000 | 1000 | 4000 |

***Insertion*** of elements into BDFT requires modification of the first bin only. The current counters corresponding to the items hashes must be read, incremented, and then updated with the new values, and a check performed for overflow. The step requires 3 memory reads and 3 writes, and 3 comparisons.

***Removal*** of items requires searching the bins starting with the shortest duration bin first. Given a nominal BDFT bin setup for fine-grained monitoring of nominal Internet traffic, about 75% of searches will end with the first bin, about 15% will end in the second, and some flows will require searching all bins [6][17]. We take the expected case to be two bins. Searching one bin requires three memory reads, and three comparisons. The removal operation is the reverse of insertion; decrement the counters and write the new values to the filter.

***Searching*** starts with the longest duration bin. Searching is the most expensive operation in BDFT, with the worst-case requiring reading the counters corresponding to the hashes in every bin except the shortest. This would require 3×(number of bins - 1) memory reads and the same number of comparisons, with about half the bins being the average case.

***Aging*** of flows must be performed during BDFT operation. Depending on the implementation, aging bins may require merging counting Bloom filters of different sizes, without overwriting the data in the recipient filter. In this case, the merging process requires reading all of the counters in both filters, adding them together, and writing the results to the longer-duration filter. In special cases, the aging process can be optimized. For instance, if both filters are the same size, and the recipient filter can be overwritten (because its contents were just aged to a longer duration bin), then aging the filter is a simple pointer update.

Aging occurs every 15 seconds, or as often as whatever the minimum duration state represents. In a nominal BDFT configuration, only the shortest duration filter needs to be processed every 15 sec, and the second shortest duration filter every 30 sec, and so on for all filters. The longer duration filters need to be processed only every 5-10 min. To calculate relative performance, it is assumed that the first Bloom filter in BDFT is of size 1000 and so is the second filter, and a merge operation must take place.

In brief, BDFT provides very good computational requirements for the most frequent operations, e.g., insert and remove operations. Search operation can be done efficiently. Aging requires more operations, depending on the filter size.

## X. Conclusion and Future Research

This paper presented a method of per-flow state tracking–BDFT. The method was analyzed to determine computational performance, and simulations were run with two real-world packet traces to determine memory usage and accuracy. Both computational performance and memory efficiency are critical to per-flow state tracking for high-speed routers. Based on the results, the "binning" concept appears to offer good performance for tracking per-flow state for a specific class of state machines.

Future research directions include a comparison of BDFT with other measurement methods, e.g., FCF ACSM [3], and a detailed investigation of the effect of false positives. The definition of a tracking success adopted in this paper is an estimated flow duration result that is within 50% of the actual flow duration, for flows that are greater than 30 seconds in duration. A more accurate or more restrictive definition of a tracking success along with its associated performance cost could also be investigated.

## ACKNOWLEDGEMENT

The authors thank Ontario Centres of Excellence (OCE), Canada, and Alcatel-Lucent, Ottawa, for support. Also, the experimental results were generated from traces made available to us by CAIDA and NLANR.

## References

[1] M. E. Attig and J. Lockwood, "SIFT: Snort intrusion filter for TCP," *Proc. of Symp. on High Performance Interconnects*, 2005, pp. 121–127.

[2] B. H. Bloom, "Space/time trade-offs in hash coding with allowable errors," *Communications of the ACM,* 13(7), 1970, pp. 422–426.

[3] F. Bonomi, M. Mitzenmacher, R. Panigrah, S. Singh, and G. Varghese, "Beyond Bloom filters: from approximate membership checks to approximate state machines," *SIGCOMM Computer Communication Review*, 36(4), 2006, pp. 315–326.

[4] A. Broder and M. Mitzenmacher, "Network applications of Bloom filters: a survey," *Internet Mathematics*, 1(4), 2004, pp. 485–509.

[5] A. Z. Broder and M. Mitzenmacher, "Using multiple hash functions to improve IP lookups," *Proc. of INFOCOM*, 2001, pp. 1454–1463.

[6] N. Brownlee and K. Claffy, "Understanding internet traffic streams: dragonflies and tortoises," *IEEE Communications Magazine*, 40(10), 2002, pp. 110–117,.

[7] K. Cheng, L. Xiang, M. Iwaihara, H. Xu, and M. Mohania, "Time-decaying bloom filters for data streams with skewed distributions," *Proc. of the 15th Int'l Workshop on Research Issues in Data Engineering: Stream Data Mining and Applications,* 2005, pp. 63–69.

[8] L. Fan, P. Cao, J. Almeida, and A. Z. Broder, "Summary cache: a scalable wide-area Web cache sharing protocol," *IEEE/ACM Transactions on Networking*, 8(3), 2000, pp. 281–293.

[9] National Lab. for Applied Network Research. NCAR-1 trace, collected in December, 2003, NSF ANI-0129677 (2002) and ANI-9807479 (1998), pma.nlanr.net/Special/ncar1.html (last accessed in Dec 2006).

[10] T. Karagiannis, A. Broido, M. Faloutsos, and K. Claffy, "Transport layer identification of P2P traffic," *Proc. of the 4th ACM SIGCOMM Conf. on Internet Measurement*, 2004, pp. 121–134.

[11] A. Kirsch and M. Mitzenmacher, "Using a queue to de-amortize cuckoo hashing in hardware," *Proc. of the 45th Annual Allerton Conf. on Communications, Control, and Computing*, 2007, pp. 751–758.

[12] A. Kirsch and M. Mitzenmacher, "The power of one move: hashing schemes for hardware," *Proc. of IEEE INFOCOM*, 2008, pp. 106–110.

[13] A. Kumar, J. Xu, L. Li, and J. Wang, "Space-code Bloom filter for efficient traffic flow measurement," *Proc. of the 3rd ACM SIGCOMM Conf. on Internet Measurement*, 2003, pp. 167–172.

[14] S. Sen, O. Spatscheck, and D. Wang, "Accurate, scalable in-network identification of P2P traffic using application signatures," *Proc. of the 13th Inte'l Conf. on World Wide Web*, 2004, pp. 512–521.

[15] K. Shah, S. Bohacek, and A. Broido, "Feasibility of detecting TCP_SYN scanning at a backbone router," *Proc. of the American Control Conf.*, 2004, pp. 988–995.

[16] C. Shannon, E. Aben, K. Claffy, D. Andersen, and Nevil Brownlee. The CAIDA OC-48 traces dataset, collected in

April, 2003,. http://www.caida.org/data/passive/ (last accessed on Dec 17, 2006).

[17] B. Whitehead, *Binned Duration Flow Tracking and Symmetric Connection Detection*, Master's Thesis, Carleton Univ., Canada, 2007.

[18] B. Whitehead, C.-H. Lung, and P. Rabinovitch, "A TCP connection establishment filter: Symmetric connection detection," *Proc. of IEEE ICC*, 2007, pp. 247–25.

**Brad Whitehead** is currently Principal Software Engineer at Yandex, Inc in Palo Alto, California. He received his B. Eng. degree in systems and computer engineering and M.A.Sc. in Electrical Engineering from Carleton University. He has held positions at Google Inc, and Alcatel-Lucent, where part of this research was conducted under a grant from Ontario Centers of Excellence. Current research interests include machine learning, data mining in real time streams, trend estimation, and optimization.

**Chung-Horng Lung** received the B.S. degree in Computer Science and Engineering from Chung-Yuan Christian University, Taiwan and the M.S. and Ph.D. degrees in Computer Science and Engineering from Arizona State University. He was with Nortel Networks from 1995 to 2001. In September 2001, he joined the Department of Systems and Computer Engineering, Carleton University, Ottawa, Canada, where he is now an associate professor. His research interests include: Communication Networks, Wireless Ad Hoc and Sensor Networks, and Software Engineering.

**Peter Rabinovitch** is currently pursuing a PhD in Probability and Statistics at Carleton University, Ottawa, Canada. He has an MSc, Carleton University, 2000 in Statistics applied to Telecommunications Networking, and a BA, Concordia University, Montréal, Canada 1992 in Mathematics. He currently works as a Systems Architect at Research in Motion (RIM) in Ottawa. Previously, he was a Senior Research Scientist in Alcatel-Lucent's Bell Labs/ Research & Innovation, and before that held various positions in IT.

# Impact of Retransmission Mechanism on SIP Overload: Stability Condition and Overload Control

Yang Hong, Changcheng Huang, James Yan
Dept. of Systems and Computer Engineering, Carleton University, Ottawa, Canada
E-mail: {yanghong, huang}@sce.carleton.ca, jim.yan@sympatico.ca

***Abstract*—SIP (Session Initiation Protocol) has been widely adopted as a signaling protocol to establish, modify and terminate media sessions between end-users in the Internet. SIP introduces a retransmission mechanism to ensure the reliability of its real-time message delivery. However, retransmission can make server overload worse, leading to server crashes in SIP-based carrier networks (e.g. Skype). In order to study the impact of retransmission mechanism on SIP overload, in this paper, we create a discrete time fluid model to describe the queuing dynamics of an overloaded SIP server. Then we derive a sufficient stability condition that a SIP server can handle the overload effectively under the retransmission mechanism. Fluid model allows us to run fluid-based Matlab simulation directly to evaluate the overload performance. Event-driven OPNET simulation was also conducted to validate our fluid model. Our simulation results demonstrate that: (1) the sufficient stability bound is quite tight. The bound indicates that effective CPU utilization as low as 20% can still lead to an unstable system after a short period of demand burst or a temporary server slowdown. Resource over-provisioning is not a viable solution to the server crash problem; (2) by satisfying the stability condition, the initial queue size introduced by a transient overload can avoid a system crash. Such stability condition can help the operator to determine whether and when to activate overload control mechanism in case of heavy load. A simple overload control solution is also proposed.**

***Index Terms*—-SIP, SIP Overload, SIP Overload Control, SIP Retransmission, Stability Condition, CPU Utilization**

## I. INTRODUCTION

SIP (Session Initiation Protocol) [1] has been widely deployed for significantly growing session-oriented applications in the Internet, such as Voice-over-IP, instant messaging and video conference. As a signaling protocol, SIP is responsible for creating, modifying and terminating sessions in a mutual real-time communication [2]. 3GPP (3rd Generation Partnership Project) has adopted SIP as the basis of the IMS (IP Multimedia Subsystem) architecture [3-5].

Fig. 1 illustrates a simplified configuration of a SIP network which consists of two basic elements: UA (User Agent) and P-Server (Proxy Server) [1]. A UA may perform two roles: in the UAC (User Agent Client) role, the originating UA sends requests; in the UAS (User Agent Server) role, the terminating UA receives requests and sends responses. The task of a P-server is to receive SIP requests and forward them to the terminating UA (or to another P-server that is closer to the terminating UA). Each P-server is assigned to serve multiple individual UAs. UA and P-server cooperate to establish, modify and terminate media sessions.

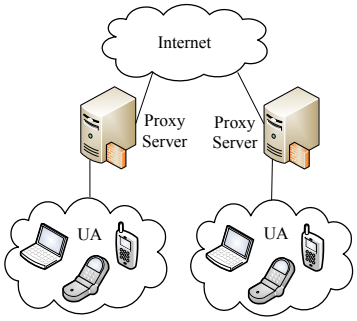



Figure 1. Simplified configuration of a SIP network.

SIP is designed to be an application layer protocol independent of the underlying transport mechanism which may be TCP (Transmission Control Protocol) or UDP (User Datagram Protocol). SIP introduces a retransmission mechanism (which will be reviewed briefly in Appendix A) to maintain its reliability by retransmitting lost SIP messages either end-to-end or hop-by-hop [5, 6]. A SIP source uses a delay to detect a message loss. It would produce one or more retransmissions if the corresponding reply message is not received in a predetermined time interval. If a retransmission triggered by a delay caused by the overload, it would introduce the overhead rather than reliability into the network. Such redundant retransmissions increase the memory and CPU load for a SIP server, which may cause a system overload and deteriorate the signaling performance [6-20]. In an overload situation, the throughput drops down to a small fraction of the original processing capacity, thus poses a serious problem for a SIP network [12]. This kind of behaviour has happened in real carrier networks where large scale collapses of SIP servers have been observed in present of sporadic SIP traffic burst (e.g., emergency-induced call volume) [13].

A SIP server can be overloaded due to various reasons such as poor capacity planning, dependency failures, component failures, avalanche restart, flash crowds,

Manuscript received December 15, 2010; revised March 31, 2011; accepted March 31, 2011.


doi:10.4304/jnw.7.1.52-62

denial of service attacks, etc., as indicated by RFC 5390 [21]. In general, a short period of demand burst or a server slowdown may bring a server overload and lead to server crash. The built-in SIP overload control mechanism has proven to be ineffective in practice, because it attempts to mitigate the overload by rejecting some calls, but the cost of rejecting a SIP session is comparable with the cost of serving a session.

SIP retransmission mechanism should be disabled for hop-by-hop transaction when running SIP over TCP to avoid redundant retransmissions at both SIP and TCP layer [1]. However, TCP layer lacks awareness of application context at SIP layer. TCP flow control mechanism cannot prevent SIP overload collapse, as indicated by recent experimental evaluation on SIP-over-TCP overload behaviour in [14]. Therefore, almost all real SIP networks run SIP over UDP mainly because the following reasons [9-23]: (1) TCP is optimized for accurate delivery by sacrificing its timeliness which is a critical requirement for real-time application such as SIP; (2) SIP works at application layer while TCP works at transport layer. Even TCP can provide reliability at transport layer, SIP messages can still be dropped or corrupted while being processed at application layer; (3) TCP keeps retransmitting outstanding packets until an ACK is received. Each retransmitted packet is pushed to the application layer to be a SIP message which costs extra CPU time and introduces more delay, therefore making CPU overload worse.

The contributions of this paper are: (1) Deriving a sufficient stability condition that a SIP server can handle the overload effectively under the retransmission mechanism; (2) Developing a discrete-time fluid model to reduce significantly simulation effort; (3) Comparing the results of both fluid-based Matlab simulation and event-driven OPNET simulation to demonstrate that the fluid-based simulation is relatively accurate and scalable for evaluating the performance of a SIP network; (4) Performing fluid-based simulation and event-driven simulation to verify that the stability bound is quite tight, or an initial queue size (created by a transient overload) which is 5% higher than the bound will bring a server crash. An effective CPU utilization as low as 20% cannot prevent a SIP server from overload during a short period of maintenance service and such overload continues to spread even after the normal service resumes; (5) Proposing a simple overload control solution to prevent a SIP server from overload collapse.

The paper is organized as follows. Section II reviews the related work on SIP overload and discusses simulation approaches. Section III analyzes the queuing dynamics of an overloaded SIP server under retransmission mechanism. Section IV derives a stability condition for SIP retransmission mechanism in the case of server overload. Section V evaluates the performance of an overloaded server. An overload control algorithm is proposed and validated in Section VI. Some conclusions are made in Section VII.

## II. Related Work

A large scale collapse of SIP servers would result in widespread outage similar in impact to that recently reported by Skype [24]. Such a major risk has motivated a surge of research attention to SIP overload control. Experiment evaluations demonstrate that the retransmission mechanism can deteriorate the overload performance [12, 19]. Thus it is necessary to investigate the impact of the retransmission mechanism on the SIP overload. A demand burst or routine server maintenance such as database synchronization may accumulate the signaling messages to create a long queue. Excessive queuing delay, introduced by a long initial queue size, may continue to trigger the redundant retransmissions to crash the server after an overloaded server resumes its normal service with a low effective CPU utilization. It would be interesting to find a sufficient stability condition for the initial queue size, which indicates whether the SIP server can handle overload effectively. Such stability condition can help the SIP operator to decide whether and when to activate the overload control algorithm. It also can help researchers propose more effective solutions to avoid SIP overload collapse caused by the SIP retransmissions.

Some attempts have been made to avoid overload collapse in SIP networks (e.g., [9-19]). For example, three window-based feedback algorithms were proposed to adjust the message sending rate of the upstream SIP servers based on the queue length [12]. Both centralized and distributed overload control mechanisms for SIP were investigated in [13]. Retry-after control, processor occupancy control, queue delay control and window based control were proposed to improve goodput and prevent overload collapse in [10]. However, these overload control proposals suggested that the overloaded receiving server advertises to its upstream sending servers to reduce their sending rates. Such pushback control solution would increase the queuing delays of newly arrival original messages at the upstream servers, which in turn cause overload at the upstream servers. Overload may thus propagate server-by-server to sources and block large amount of calls which means revenue loss for carriers. Similarly, a small buffer size has been proved to be a simple overload control mechanism at a cost of arbitrarily high call rejection rate and revenue loss in [25].

When retransmissions are caused by overload rather than message loss, they would bring extra overhead instead of reliability to the network and exacerbate the overload [19]. Different from all existing solutions discussed above, we propose to reduce the retransmission rate instead of reducing original message sending rate of the upstream servers. This will mitigate the overload while maintaining original message sending rate, which leads to less blocking calls and more revenue for carriers.

Event-driven simulation has been widely used for evaluating network performance. Its computation cost grows linearly with network sizes and message volumes [26]. When event-driven simulation is used to evaluate a SIP network, each outstanding SIP message requires a timer being maintained. When an overload happens,

outstanding messages are built up, and the simulator needs to increase the number of timers dramatically in order to track message retransmissions. Tracking and manipulating these timers consume large amount memory and CPU time which make the simulation process extremely slow, thus in some cases, cause the simulator to crash and terminate simulation unexpectedly. In order to simplify the CPU-consuming timer-tracking process, fluid-based simulation tracks time slot instead of individual messages. Messages arriving within the same time slot are aggregated and processed together. This will greatly simplify the complexity caused by large number of messages and allow smooth scalability by choosing different granularities as required.

## III. Dynamics Of SIP Retransmission Mechanism

A real SIP network consists of a series of geographically distributed P-servers and a large amount of UAs. Each P-server is responsible for setting up a session call between two UAs. It forwards the requests and also generates a provisional response to confirm the receipt of every request from the upstream sender (an originating UA or a P-server) [1]. It provides a retransmission mechanism to guarantee a reliable delivery of a SIP message [5]. However, the arrival of too many SIP messages may cause an unnecessary queuing delay, stimulate redundant retransmissions and accelerate the overload, thus eventually bring down the entire network [12]. Therefore, it is necessary to describe the queuing dynamics of an overloaded SIP server (e.g., [8, 25]), before we derive a stability condition for SIP retransmission mechanism.

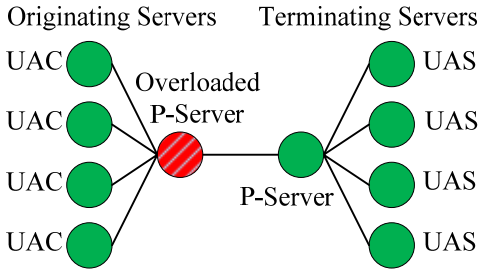



Fig. 2. SIP network topology with an overloaded P-server (which is marked in diagonal lines) and its multiple upstream originating servers.

When overloads happen in the network, at any time, one of the servers will be the most overloaded one among all the overloaded servers. Without loss of generality, we consider a typical SIP network which consists of an overloaded P-server and its multiple upstream originating servers [12], as shown in Fig. 2. For a clear presentation, we use difference equations to describe the queuing dynamics of an overloaded P-server. In our discrete time model, we make the following assumptions according to SIP RFC [1]:

(a) We investigate the retransmissions which are mainly caused by long queuing delay of the overloaded server. Therefore, for the round trip response time between the overloaded server and its neighbouring server, the queuing and processing delays are dominant, while transmission and propagation delay are negligible [13]. This assumption is valid because signaling messages are typically CPU capacity constrained rather than bandwidth constrained;

(b) Time is divided into discrete time slots. This makes it easy to describe how many retransmitted messages are triggered by a delay caused by the overload. The errors introduced by the discrete time slot can be made arbitrarily small by making the interval of a timeslot smaller and smaller. We use $t$ and $n$ to denote time and timeslot respectively;

(c) The SIP RFC [1] does not specify the queuing and scheduling discipline to be deployed by a SIP server. We assume that a SIP server maintains a First-In-First-Out (FIFO) queue for messages arriving at different time-slots. This FIFO queuing model reflects the common practice by most vendors today [11]. Within the same time slot, original request messages enter the tail of the queue prior to retransmitted request messages. Such enqueuing priority has negligible impact if the interval of the time slot is very small. There is no enqueuing difference for the messages arriving at different time slots;

(d) The time to process a response message or a timer timeout is typically much smaller than a request message [1]. We assumed that, within a time slot, the server has enough CPU capacity to process the incoming response messages, thus response messages will not be enqueued as long as they are treated with higher priority such as interrupt. They will not be dropped either when the queue for request messages are overflowed. The service capacity of the overloaded server includes the rate for processing response messages;

(e) In order to focus our analysis on the overloaded server, we assume multiple upstream originating servers and the downstream server of the overloaded P-server have sufficient capacity to process all requests, retransmissions, and response messages immediately without any delay;

(f) Practical buffer sizes vary with the actual service rates and system configuration plans. With the memory becoming cheaper and cheaper, typical buffer sizes are likely to become larger and larger. The buffer sizes for all servers are assumed to be large enough to hold all the incoming messages. Therefore there is no message loss at all servers.

(g) The hop-by-hop Invite-100Trying transaction is the major workload contributor due to its role for call setup and its hop-by-hop retransmission mechanism [1]. Given the proportionate nature and the general similarity of the retransmission mechanisms between the “Invite” and “non-Invite” messages in a typical session [1], we will focus on the hop-by-hop Invite-100Trying transaction and ignore other end-to-end transactions in this paper.

Fig. 3 depicts the queuing dynamics of an overloaded SIP server in a SIP network (as shown in Fig. 2). The overloaded server receives the original Invite requests with an aggregate rate $\lambda(n)$ at time slot $n$, where $\lambda(n)$ can be arbitrary. We can obtain the queue size $q(n+1)$ at next time slot $n+1$ based on the information at the current time slot $n$, i.e.,

$$q(n+1)=[q(n)+\lambda(n)+r(n)-\mu(n)]^{+}. \quad (1)$$

where $q(n)$ denotes the queue size; $r(n)$ denotes the aggregated retransmitted messages; $\mu(n)$ denotes an arbitrary service process for the request messages, which is equal to the server service capacity minus the service rate for the response messages. $\lambda(n)$ plus $r(n)$ give the total arrival messages at current time slot $n$. Adding $q(n)$ and deducting $\mu(n)$ would generate a new queue size $q(n+1)$ in the next time slot $n+1$, as described by Eq. (1). We use $[]^+$ to indicate that the queue size at each time slot should be nonnegative.

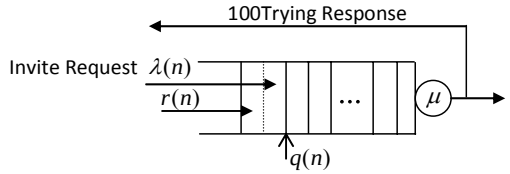



Fig. 3. Queuing dynamics of an overloaded SIP server ($\lambda(n)$ denotes aggregated original message arrivals, $r(n)$ denotes aggregated retransmitted message arrivals from multiple upstream servers, $q(n)$ denotes queue size, $\mu(n)$ denotes service rate).

If the server does not receive the corresponding response message for an original request message at a specific time-out, it would trigger retransmission. There are maximum 6 retransmissions for every original request message [1]. Thus we can obtain the total retransmitted messages $r(n)$ at current time slot $n$ as

$$r(n) = \sum_{j=1}^{6} r_j(n), \tag{2}$$

where $r_j(n)$ denotes the $j^{th}$–time retransmission for the original request messages arriving at time $n-T_j$, $T_j=(2^j-1)T_1$ and $1\le j\le 6$.

At time $(n-T_j)$, the original request message arrivals were $\lambda(n-T_j)$ and the queue size was $q(n-T_j)$. Since the overloaded server can process $\sum_{k=1}^{T_j}\mu(n-T_j+k)$ messages during the $T_j$ time slots, the remaining messages at current time slot $n$ become $[\lambda(n-T_j)+q(n-T_j)-\sum_{k=1}^{T_j}\mu(n-T_j+k)]^+$, which should be nonnegative. This may include both the original arrival messages at time $(n-T_j)$ and the queued messages right before the time slot $(n-T_j)$. However, only the remaining original arrival messages $\lambda(n-T_j)$ need to be retransmitted at current time $n$, we use $\min\{\}$ function to get the $j^{th}$–time retransmitted messages at current time slot $n$, i.e.,

$$r_j(n) = \min\{[\lambda(n-T_j)+q(n-T_j)-\sum_{k=1}^{T_j}\mu(n-T_j+k)]^+, \lambda(n-T_j)\} \tag{3}$$

Eqs. (1) to (3) shows the dynamic behaviour of an overloaded SIP server. Due to its nonlinear characteristic, it may show complex, sometimes chaotic, patterns that bring a potential server collapse.

## IV. Stability Condition For SIP Retransmission Mechanism

The retransmission can provide a reliable delivery of SIP messages. However, it also increases the queuing size and enhances the overload. It would be interesting to derive a stability condition that the server can handle overload effectively. The messages accumulated by a transient overload (e.g., a demand burst or a server slowdown) create an initial queue size when the server returns to its normal service state. Such initial queue size may bring a queuing delay long enough for the retransmissions of old remaining original messages in the queue as well as all the new incoming original messages. We would like to investigate whether the SIP server can serve the original messages in the initial queue size and their retransmissions under a low effective CPU utilization.

Without loss of generality, we consider the "Invite-Trying" request-response pair with a deterministic arrival rate $\lambda$ and a deterministic service rate $\mu$; there are $i$ retransmissions for the new arrival original messages; the initial queue size is $q(0)$.

*Theorem 1*: If the initial queue size $q(0)$ created by a demand burst can satisfy a sufficient stability condition described by Eq. (4), then the SIP server is stable.

$$q(0) < \min\left\{(2^{j+1}-1)\mu T_1, \frac{(2^{j+1}+3\times 2^i-i-4)\mu T_1-((i-1)2^i+1)\lambda T_1}{i+1}, 1\le i\le j\le 6\right\} \tag{4}$$

*Proof*:

To prevent messages from accumulating unlimitedly in SIP server, the total average incoming rate should be less than the service rate. Assume that there would be $i$ retransmissions for an arbitrary original Invite request message, a conservative condition to maintain stability is

$(i+1)\lambda/\mu\le 1$,

which is equivalent to

$\mu\ge(i+1)\lambda$, (5)

i.e.,

$i\le(\mu-\lambda)/\lambda$. (6)

To achieve the above sufficient stability condition, we need to guarantee that the original messages from both the initial queue size and the new arrivals are not retransmitted more than $j$ times, where we denote $j$ as $j=\lfloor(\mu-\lambda)/\lambda\rfloor$. Then we update the equivalent stability condition in Eq. (6) as

$i\le j=\lfloor(\mu-\lambda)/\lambda\rfloor$. (7)

To avoid $j+1$ retransmissions for the original messages in the initial queue size, we obtain a stability condition for the initial queue size as

$q(0)/\mu<T_{j+1}=(2^{j+1}-1)T_1$,

which is equivalent to

$q(0)<\mu T_{j+1}$. (8)

To avoid $(j+1)$ retransmissions for any newly arrival original messages, the queue size in any time should satisfy

$q(t)<\mu T_{j+1}$. (9)

Eq. (4) can certainly satisfy Eq. (8). To show that Eq. (4) can satisfy the requirement of Eq. (9), we consider five cases in the following discussion.

1. We first consider the queue sizes at each specified retransmission times $T_i=(2^i-1)T_1$ using Eqs. (1) to (3) as follows,

$q(T_1)=q(0)-(\mu-\lambda)T_1+[q(0)-\mu T_1]^+$,

$q(T_2)=q(T_1)-2(\mu-2\lambda)T_1+[q(0)-\mu T_2]^+$,

$$\vdots$$

$$q(T_i)=q(T_{i-1})-2^{i-1}(\mu-i\lambda)T_1+[q(0)-\mu T_i]^+, \quad (10)$$

$$\vdots$$

$$q(T_6)=q(T_5)-32(\mu-6\lambda)T_1+[q(0)-\mu T_6]^+.$$

2. We next consider the queue sizes between any two neighbouring retransmission times $T_{i-1}$ and $T_i$. Eqs. (1), (2) (3), (7) and (10) lead to

$$q(t)=q(T_{i-1})-(\mu-i\lambda)(t-T_{i-1})<q(T_{i-1}), \quad (11)$$

The inequality in Eq. (11) indicates that the queue size is decreasing continuously with a slope of $\mu-i\lambda$ during the time period. However, at time $t=T_i$, the *ith* retransmission for the remaining $[q(0)-\mu T_i]$ messages from the initial queue size $q(0)$ is triggered, resulting in a sudden increase in the queue size described by (10). Then when $0<t\le T_j$, the condition described by (9) becomes

$$q(T_i)<\mu T_{j+1} \qquad 1\le i\le j\le 6. \quad (12)$$

Given the condition of Eq. (8), we assume the worst case with $q(0)-\mu T_i\ge 0$. Using recursive substitution for Eq. (10), we can obtain

$$q(T_i)=(i+1)q(0)-\sum_{k=1}^{i}2^{k-1}(\mu-k\lambda)T_1-\sum_{k=1}^{i}(2^k-1)\mu T_1$$

which can be reorganized as

$$q(T_i)=(i+1)q(0)-\sum_{k=1}^{i}2^{k-1}\mu T_1 + \frac{d}{dx}\sum_{k=1}^{i}\lambda T_1 x^k\Big|_{x=2}-\sum_{k=1}^{i}2^k\mu T_1+i\mu T_1,$$

or

$$q(T_i)=(i+1)q(0)-\frac{1-2^i}{1-2}\mu T_1 + \frac{d}{dx}\left[\frac{x-x^{i+1}}{1-x}\lambda T_1\right]\Big|_{x=2}-\frac{2-2^{i+1}}{1-2}\mu T_1+i\mu T_1.$$

Then we can obtain

$$q(T_i)=(i+1)q(0)+((i-1)2^i+1)\lambda T_1-(3\times 2^i-i-3)\mu T_1. \quad (13)$$

Combining Eqs. (12) and (13), we can obtain the second condition in Eq.(4) as

$$q(0)<\frac{(2^{j+1}+3\times 2^i-i-4)\mu T_1-((i-1)2^i+1)\lambda T_1}{i+1} \quad 1\le i\le j\le 6 \quad (14)$$

3. We then consider the case that $T_j<t<T_{j+1}$. From Eqs. (1), (2), (3), (7), (9) and (12), we have

$$q(t)=q(T_j)-(\mu-(j+1)\lambda)(t-T_j)\le q(T_j)<\mu T_{j+1}. \quad (15)$$

This means the queue size is non-increasing during the time period $T_j<t<T_{j+1}$.

4. Next, we consider the case that $t=T_{j+1}$. Since Eq. (8) indicates $[q(0)-\mu T_{j+1}]^+=0$, from Eqs. (1), (2), (3), (7) and (12), we can obtain

$$q(T_{j+1})=q(T_j)-2^j(\mu-(j+1)\lambda)T_1+[q(0)-\mu T_{j+1}]^+\le q(T_j)<\mu T_{j+1}. \quad (16)$$

5. Finally, we consider the case that $t>T_{j+1}$. From Eqs. (1), (2), (3), (7) and (16), we have

$$q(t)=q(T_{j+1})-(\mu-(j+1)\lambda)(t-T_{j+1})\le q(T_{j+1})<\mu T_{j+1}. \quad (17)$$

Combining Eq. (8), (11), (12), (14), (15), (16) and (17), we can reach a sufficient stability condition for the initial queue size described by Eq. (4).□

## V. Performance Evaluation and Simulation

Since violating the sufficient stability condition does not always bring the instability to a SIP system, we would like to investigate how tight the sufficient stability bound for the retransmission mechanism is when a SIP overload happens. To achieve this goal, we will evaluate the performance of an overloaded SIP server by performing fluid-based Matlab simulation using the fluid model described by Eqs. (1) to (3), where the time slot is 50ms. In the mean time, in order to validate the accuracy and scalability of fluid-based simulation, we also performed event-driven OPNET simulation in a real SIP network as depicted by Fig. 2. In the OPNET simulator, messages were handled one by one instead of being aggregated over a time slot as in our Matlab simulation. Four originating servers generated original request messages with equal rate, and then sent them to four terminating servers via two P-servers. All the sending servers also maintained a list of all outstanding messages for tracking retransmissions. The mean service rate of P-Server is set to be the same as the mean service rate of the corresponding P-server in MATLAB simulation. For OPNET simulation, messages were enqueued based on first-come-first-in principle. That is, Assumption (c) was unnecessary for OPNET simulation. The default timer for the first retransmission was $T_1$=0.5s [1].

The retransmission messages triggered by the overload are redundant messages. Therefore, only the CPU consumed by the original messages can be regarded as effective use of resources. We define effective CPU utilization $\rho$ as the ratio between the total mean arrival rate for the original messages and the mean service rate, i.e., $\rho=\lambda/\mu$.

*Simulation Setting of Network Configuration Parameters*: The mean arrival rate and the service capacity of a SIP server may be different across different real carrier networks. To reduce the overload probability, resource over-provisioning has been employed to maintain a low effective CPU utilization in the real carrier networks [12]. Therefore, for both Matlab and OPNET simulation, we make the CPU utilization low by selecting appropriate mean original message arrival rate of regular demands and mean service rate. For the fluid-based Matlab simulation based on the fluid model, smaller time slot corresponds to more accurate simulation result, but a longer simulation time. In addition, the smallest time slot for arrival rate or service rate should guarantee at least one message per slot, e.g., the smallest time slot for 200 messages/sec is 5ms. Performance comparison between Matlab simulation and OPNET simulation in the following subsection demonstrates that a time slot of 50ms is relatively accurate for the fluid-based simulation.

To verify the sufficient stability condition for the initial queue size, we have considered two typical overload scenarios: (1) The overload was caused by a demand burst, while arrival rate and service rate were deterministic; (2) The overload was caused by a server slowdown, while arrival rate and service rate were Poisson distributed[1].

[1] Currently there is no measurement result for the workload in the real SIP networks. Poisson distributed message arrival rate and service rate are widely adopted by most existing research work (e.g., [13]).

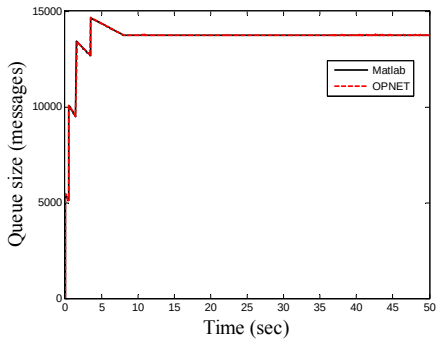
*A. Overload Caused by Demand Burst*

In this scenario, a demand burst overloaded the server and created an initial queue size at time $t$=0s, emulating a short surge of user demands; normal original request messages arrived at the overloaded server with a constant rate $\lambda$=200 messages/sec, emulating regular user demands. The overloaded server maintained a constant service rate $\mu$=1000 messages/sec. Thus the effective CPU utilization for regular user demands is $\rho=\lambda/\mu$=20%. The simulation period is 50s.

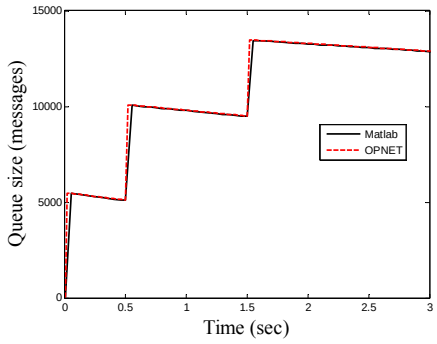
Eq. (7) gives $j=\lfloor(\mu-\lambda)/\lambda\rfloor$=4. Then using Eq. (4), we can obtain the stability condition for the overloaded server as $q(0)<\min\{15500, 8200, 6167, 5700, 6220\}$=5700 messages. We will consider two sub-scenarios with different initial queue sizes.

*1) Sub-scenario* (*a*): In this sub-scenario, a demand burst created an initial queue size as $q(0)$=5500 messages < 5700 messages, obeying the stability condition described by Eq. (4).

Figs. 4 and 5 show the dynamic behaviour of the overloaded SIP server using both Matlab simulation and OPNET simulation. One can observe that the curves obtained by Matlab simulation are very close to the curves obtained by OPNET simulation. The difference for instantaneous retransmission rate shown in Fig. 5 was caused by enqueuing priority within the same time slot (Assumption (c)). The similarity between Matlab simulation result and OPNET simulation result demonstrates that fluid-based simulation is a relatively accurate and cost-effective approach for performance evaluation of a SIP network, while it can simplify a CPU-consuming timer-tracking process by tracking single time slot instead of individual message timer.




(a) full view




(b) enlarged partly view

Fig. 4. Queue size $q$ (messages) versus time for the overloaded server when the initial queue size obeys the stability condition.

Fig. 4(b) shows that the queue size decreased linearly with 800 messages/sec at the beginning.

At time $t=T_1$=0.5s, the overloaded SIP server had processed 500 messages, the 1st-time retransmission for the residual 5000 original messages in the initial queue happened (as shown in Fig. 5). The new 100 original messages arriving between $t$=0s and $t=T_1$=0.5s joined the queue together with 5000 retransmitted SIP messages, so the queue size became 10,100 messages (as shown in Fig. 4(b)). The new arrival original messages at time $t$=0s started to trigger the first-time retransmissions (as shown in Fig. 5). Similarly, due to the 2nd-time and 3rd-time retransmissions, the queue size increased dramatically at time $t=T_2$=1.5s and $t=T_3$=3.5s respectively.

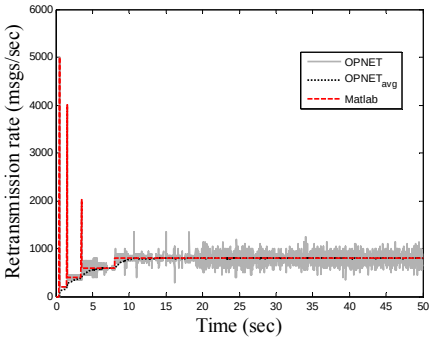



Fig. 5. Retransmission rate $r$ and moving average retransmission rate $r_{avg}$ (messages/sec) versus time for the overloaded server when the initial queue size obeys the stability condition.

At time $t$=8s, the retransmission rate of new arrival original messages increased from 600 messages/sec to 800 messages/sec (as shown in Fig. 5), thus the total incoming traffic rate of both original messages and retransmitted messages was equal to the service rate $\mu$=1000 messages/sec (or $\rho'=5\lambda/\mu$=1). Between the time $t$=3.5s and $t$=8s, 900 new incoming original messages and 2700 incoming retransmitted messages entered the overloaded SIP server, thus the queue size reached and stayed at a steady queue size as 14700+900+2700-4500=13800 messages, well matching our theoretical analysis on the queuing dynamics in Section III.

*2) Sub-scenario* (*b*): In this sub-scenario, a demand burst created an initial queue size as $q(0)$=6000 messages > 5700 messages, violating the stability condition described by Eq. (4).

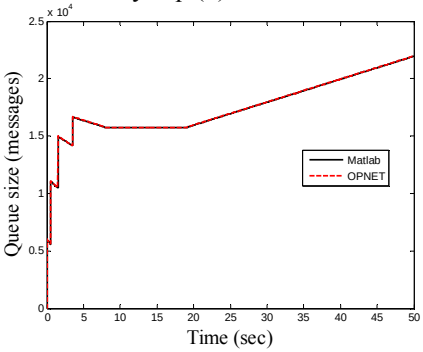



Fig. 6. Queue size $q$ (messages) versus time for the overloaded server when the initial queue size violates the stability condition.

Fig. 6 shows that the queue size decreased linearly except 4 spikes due to the dramatic retransmissions until the time $t$=19s. At time $t$=19s, the retransmission rate of

new arrival original messages increased from 800 messages/sec to 1000 messages/sec (as shown in Fig. 7). The total incoming traffic rate of both original messages and retransmitted messages was larger than the service rate $\mu$=1000 messages/sec (or $\rho'=6\lambda/\mu=1.2>1$). Therefore, after the time $t$=19s, the queue size increased linearly and continuously with 200 messages/sec (as shown in Fig. 6), which would bring a SIP server crash eventually.

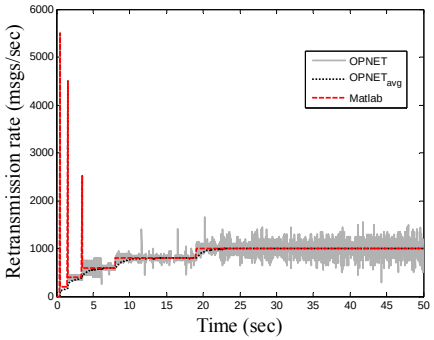



Fig. 7. Retransmission rate $r$ and moving average retransmission rate $r_{avg}$ (messages/sec) versus time for the overloaded server when the initial queue size violates the stability condition.

*3) Remarks*: In case of the deterministic traffic pattern, slightly different initial queue sizes due to the demand bursts (the difference is less than 10% in the two sub-scenarios) create totally different dynamic behaviour patterns. The slightly smaller initial queue size of 5500 messages allows the server to handle the initial temporary overload effectively, while the slightly larger initial queue size of 6000 messages will result in infinitely increasing queue size, thus bring a SIP server to crash. This indicates that the sufficient stability bound described by Eq. (4) is quite tight when both original message arrival rate of regular demands and service rate are deterministic.

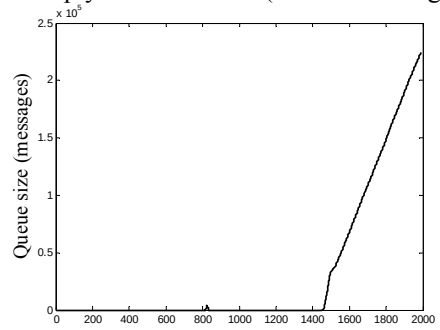
When the original message arrival rate and service rate, are arbitrarily distributed, the operators can apply Theorem 1 to determine the stability condition using the mean values of the original message arrival rate and service rate. A moving average filter can be used to measure the mean values $\lambda$ and $\mu$. Our simulation result in the next subsection indicates that the sufficient stability bound is also tight for Poisson distributed arrival rate and service rate which has been widely considered as one of the typical traffic patterns in the real carrier networks [13].

### *B. Overload Caused by Server Slowdown*

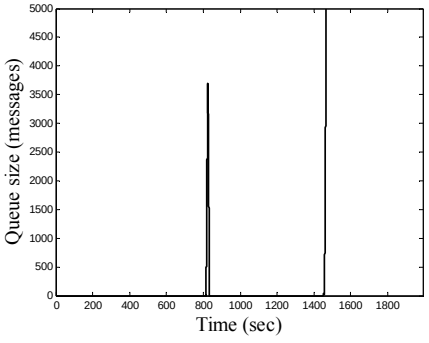
In this application scenario, the overloaded SIP server worked in one of the two states (i.e., normal service state and maintenance state) alternately. During the maintenance period, the overload may happen due to the server slow down. The mean service time at the normal service state was $m_1$=600sec; the mean service time at the maintenance state was $m_0$=30sec; all were exponential distributed. The mean service rate at the normal service state was $\mu_1$=1000 messages/sec; the mean service rate at maintenance state was $\mu_2$=180 messages/sec; the mean arrival rate of the SIP messages was $\lambda$=200 messages/sec; all were Poisson distributed. The simulation period is 2000s. The overall effective mean utilization was equal to $\rho=\lambda(m_1+m_0)/(m_1\mu_1+m_0\mu_0)\approx 0.21$. We will not show the OPNET simulation result because it was very close to the Matlab simulation result as the previous subsection.

Figs. 8 to 11 show the dynamic behaviour of the overloaded SIP server under two different service states.

Between the time $t$=812s and 823s, SIP server had short period of maintenance, service rate decreased (as shown in Fig. 11). The messages started to accumulate and the queue size increased to reach a peak around 3690 messages at time $t$=823s (as shown in Fig. 8(b)). When a mean service rate was 180 messages/sec, the queue size larger than 90 messages brought a queuing delay longer than 0.5s, and started to stimulate the retransmission (as shown in Fig. 10). After the server resumed normal service at time $t$=823s, the initial queue size was less than 5700 messages (the stability condition described by Eq. (4)). The server could process these accumulated messages in time, so the queue size decreased until the buffer was empty at time $t$≈833s (as shown in Fig. 8(b)).




(a) full view




(b) enlarged partly view

Fig. 8. Queue size $q$ (messages) versus time for the overloaded server which performed normal service and maintenance service alternately.

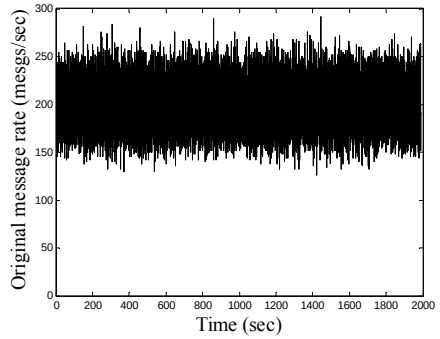



Fig. 9. Original message arrival rate $\lambda$ (messages/sec) versus time for the overloaded server which performed normal service and maintenance service alternately.

However, maintenance with a relatively long period (or a equivalent large demand burst)) happened at time $t$=1452s, the queue size increased continuously and triggered more than 5 retransmissions that made the total arrival message arrival rate exceed the normal service rate (as shown in Fig. 11). After the server entered the normal service state at time $t$=1495s, the initial queue size was larger than 5700 messages. Since the stability condition for the initial queue size was violated, the SIP server cannot handle the overload effectively. The queue size tended to infinity (as shown in Fig. 9(a)), thus eventually crashed the server.

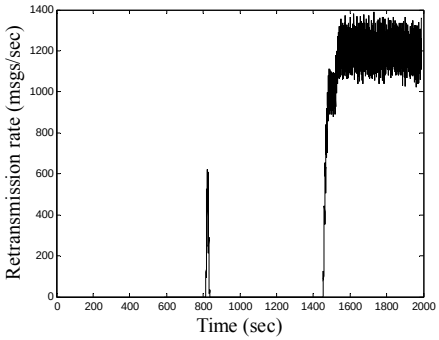



Fig. 10. Retransmission rate $r$ (messages/sec) versus time for the overloaded server which performed normal service and maintenance service alternately.

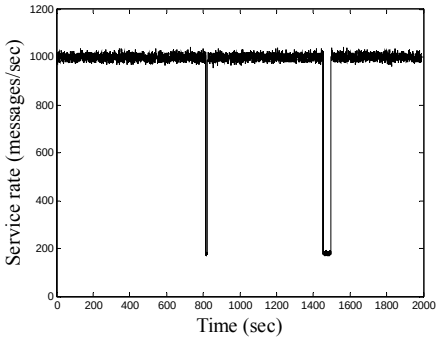



Fig. 11. Service rate $\mu$ (messages/sec) versus time for the overloaded server which performed normal service and maintenance service alternately.

In summary, although the effective mean utilization is as low as 20%, if the accumulated messages in the SIP server during the short maintenance period violate the stability condition for the initial queue size, the server cannot mitigate the overload effectively after it resumed its normal service. Goodput collapse persists and the server would crash eventually, well matching our theoretical analysis.

## VI. Overload Control Algorithm

The retransmitted messages triggered by the overload delay are redundant and may bring the overload collapse. Therefore, quite different from current existing overload control algorithms which adopt the push-back mechanism, our goal for mitigating the overload is to control the retransmission rate $r$, thus helping a server to cancel the overload eventually. To achieve our target, we use a retransmission probability $p$ to determine whether to retransmit an original message when its retransmission timer fires or expires. When the total message arrival rate of a downstream server exceeds its message processing capacity, its queue size would increase continuously, which indicates that an overload occurs. A long queue size would make a long queuing delay for the new arrival original messages sent by its upstream servers, thus triggering corresponding redundant retransmissions. Therefore, we propose to control the retransmission probability $p$ based on the average queue size $q_{avg}$ of the overloaded downstream server. Similar to existing push-back overload control solutions, we introduce a minor change to SIP protocol by defining one extra field in every response message to carry the retransmission probability $p$ calculated by the overloaded downstream server. Our simple overload control algorithm is given in Fig. 12.

*Setting $q_{min}$ and $q_{max}$*: Since the upstream servers only generate the retransmissions for the original messages whose retransmission timers fire or expire, and a queuing delay less than the 1st retransmission timer $T_1$ at the overloaded downstream server will not trigger any retransmissions, we suggest minimum and maximum thresholds as

$$q_{1\min}=0.2\mu^*T_1, \quad (18)$$

$$q_{1\max}=\mu^*T_1, \quad (19)$$

where $\mu$ is the mean service rate of the overloaded downstream server.

**Overload Control Algorithm**

Upstream Server Behaviour
  When each retransmission timer fires or expires
    Retransmit the message with probability $p$

Overloaded Downstream Server Behaviour
  Calculate retransmission probability $p$:
    if $q_{avg} < q_{min}$
      $p \leftarrow 1$
    else
      if $q_{min} \le q_{avg} \le q_{max}$
        $p \leftarrow (q_{max}-q_{avg})/(q_{max}-q_{min})$
      else
        $p \leftarrow 0$

**Server parameter:**
  $q_{avg}$: Average queue size of the overloaded downstream server
  $q_{min}$: Minimum threshold for $q_{avg}$
  $q_{max}$: Maximum threshold for $q_{avg}$

Fig. 12. Overload control algorithm for controlling retransmission rate.

### A. Modeling of Overload Control Algorithm

Since our overload control algorithm mitigates the overload by controlling retransmission probability, we need to create a respective fluid model for the fluid-based simulation.

Since stochastic process of both message arrival rate $\lambda$ and service rate $\mu$ may cause transient fluctuation in the instantaneous queue size $q$, we use an exponentially weighted moving average filter to obtain the average queue size $q_{avg}$, i.e.,

$$q_{avg}(n)=(1-w_q)q_{avg}(n-1)+w_q q(n), \quad (20)$$

where $w_q$ is the filter weight.

According to the overload control algorithm described by Fig. 12, the retransmission probability $p$ generated by the overloaded server can be computed as

$$p(n)=\min\{[(q_{max}-q_{avg})/(q_{max}-q_{min})]^+,1\} \quad (21)$$

By integrating the retransmission probability $p$ into the theoretical retransmission rate $r$, we can get the actual retransmission rate as $rp$. Therefore, we can reuse the fluid model (or Eqs. (1) to (3)) developed for the regular server in Section III by replacing the theoretical retransmission rate with the actual retransmission rate, i.e., we only need to update Eq. (3) as

$$r_j(n)=\min\{[\lambda(n-T_j)+q(n-T_j)-\sum_{k=1}^{T_j}\mu(n-T_j+k)]^+, \lambda(n-T_j)\}p(n) \quad (22)$$

### *B. Evaluation of Overload Control Algorithm*

In order to validate our overload control algorithm, we simulated a typical overload scenario caused by the server slowdown. We perform fluid-based MATLAB simulation using the fluid model we have derived. We performed our simulations with overload control algorithm and without overload control algorithm separately.

The simulation period is 90s. The mean server rate of the overloaded server was $\mu$=100 messages/sec from time $t$=0s to $t$=30s, and $\mu$=1000 messages/sec from time $t$=30s to $t$=90s; the mean original message arrival rate was $\lambda$=200 messages/sec; all were Poisson distributed. The maximum and minimum thresholds for average queue size were set as $q_{max}$=500 messages and $q_{min}$=100 messages respectively. The averaging filter weight $w_q$ was 0.1.

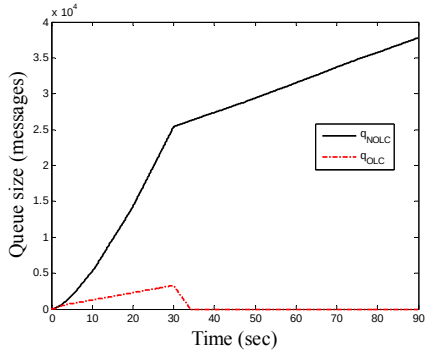



Fig. 13. Queue size $q$ (messages) versus time for the overloaded server when the overload control algorithm was/was not activated.

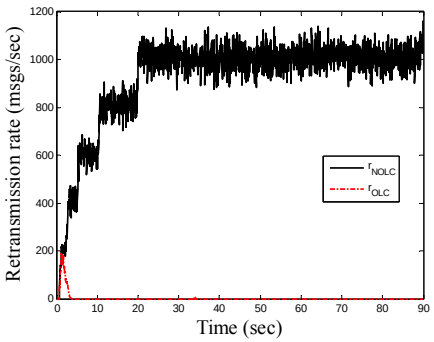



Fig. 14. Retransmission rate $r$ (messages/sec) versus time for the overloaded server when the overload control algorithm was/was not activated.

Figs. 13 and 14 show the dynamic behaviour of the overloaded server. We use "OLC"/"NOLC" to indicate that overload control algorithm "was"/"was not" activated at the upstream server of the overloaded server.

Without overload control algorithm applied, the redundant retransmissions were triggered to increase the queue size of the overloaded server sharply. After the overloaded server resumed its normal service at time $t$=30s, the mean new original message arrival rate was less than the mean service rate. However, the long initial queue size continued to stimulate redundant retransmissions (see Fig. 14). These retransmissions made the aggregated mean arrival rate of both original messages and retransmitted messages exceed the mean service rate, thus continuing to build up the queue (see Fig. 13). The persisted overload would eventually lead to an overload collapse.

With our overload control algorithm applied, the retransmission rate $r$ was restricted (see Fig. 14). The overload was mitigated and the queue size of the overloaded server increased relatively slowly. After the overloaded server resumed its normal service, it only spent 4s to cancel the overload and the buffer became empty at time $t$≈34s (see Fig. 13).

## VII. CONCLUSIONS

We have investigated the SIP retransmission mechanism in case of the overload. We have derived a sufficient stability condition that SIP server can handle the overload effectively under the retransmission mechanism. To prevent the system from crashing, the initial queue size caused by a transient overload should satisfy the stability condition. Such stability condition can help the SIP operator to trigger the overload control algorithm ahead of time to avoid the SIP server collapse.

We have performed simulation using both fluid-based simulation approach and event-driven simulation approach to evaluate the performance of an overloaded SIP server. Our study indicated that the behaviour of the SIP server is highly sensitive to the temporary overload due to the demand burst or the server slow down. The sufficient stability bound for the initial queue size caused by the overload is quite tight. Effective resource utilization as low as 20% cannot prevent an overloaded server from crash, if an initial queue created by a short-term overload (due to a demand burst or a temporary server slowdown) exceeds the sufficient stability bound slightly.

Event-driven simulation, adopted by most existing literature on SIP study, requires a series of retransmission timers to track outstanding messages, thus makes the experiment computationally expensive. As the network size increases to a large scale, the number of timers may build up to consume excessive memory and CPU time, thus crashes the simulator eventually. On the contrary, fluid-based simulation tracks time on a slot-by-slot basis. Events happening within the same time slot will be aggregated and processed together. Individual timers do not need to be tracked anymore. Thus fluid-based approach is much simpler than event-driven approach.

The similarity between fluid-based Matlab simulation result and event-driven OPNET simulation result demonstrate that fluid-based simulation can be a relatively accurate and cost-effective approach for evaluating the performance of a SIP network.

We also proposed a simple overload control algorithm to mitigate the overload by reducing the retransmission rate only, while keeping the original message rate unchanged to avoid blocking the calls unnecessarily. Our solution can help the carriers to maintain their revenue in case of the overload. The simulation has validated the effectiveness of our overload control algorithm.

## APPENDIX A. SIP RETRANSMISSION MECHANISM OVERVIEW

SIP works in the application-layer for media session establishment and tear-down. To briefly describe a basic SIP operation, we only consider originating UA, P-server and terminating UA, as shown in Fig. A1. To set up a call, an originating UA sends an "Invite" request to a terminating UA via two P-servers. The P-server returns a provisional "100 (Trying)" response to confirm the receipt of the "Invite" request. The terminating UA returns an "180 (Ringing)" response after confirming that the parameters are appropriate. It also evicts a "200 (OK)" message to answer the call. The originating UA sends an "ACK" response to the terminating UA after receiving the "200 (OK)" message. Finally the call session is established and the media communication is created between the originating UA and the terminating UA through the SIP session. The "Bye" request is generated to terminate the session thus cancel the communication.

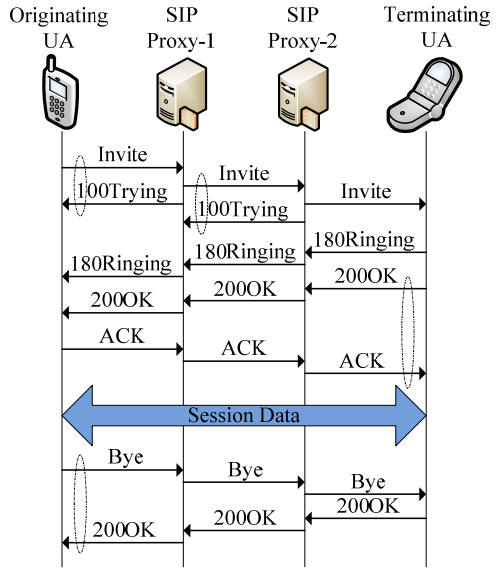



Fig. A1. A typical procedure of session establishment.

SIP has two types of message retransmission: (a) a sender starts the first retransmission of the original message at $T_1$ seconds, the time interval doubling after every retransmission (exponential backoff), if the corresponding reply message is not received. The last retransmission is sent out at the maximum time interval $64xT_1$ seconds. Default value of $T_1$ is 0.5s, thus there is a maximum of 6 retransmissions. The hop-by-hop "Invite"-"Trying" transaction shown in Fig. A1 follows this rule [1]; (b) a sender starts the first retransmission of the original message at $T_1$ seconds, the time interval doubling after every retransmission but capping off at $T_2$ seconds, if the corresponding reply message is not received. The last retransmission is sent out at the maximum time interval $64xT_1$ seconds. Default value of $T_2$ is 4s, thus there is a maximum of 10 retransmissions. The end-to-end "OK"-"ACK" and "Bye"-"OK" transactions shown in Fig. A1 follows this rule [1].

## ACKNOWLEDGMENT

This work was supported by the NSERC grant #CRDPJ 354729-07 and the OCE grant #CA-ST-150764-8.

## REFERENCES

[1] J. Rosenberg et al., "SIP: Session Initiation Protocol," *IETF RFC 3261*, June 2002.
[2] J. Rosenberg and H. Schulzrinne, "SIP: Locating SIP Servers," *IETF RFC 3263*, June 2002.
[3] 3GPP TS 24.228 v5.f.0 (2006-10), "Signaling flows for the IP Multimedia call control based on SIP and SDP; Stage 3 (Release 5)," October 2006.
[4] 3GPP TS 24.229 v8.5.1 (2008-09), "IP Multimedia call control protocol based on SIP and SDP; Stage 3 (Release 8)," September 2008.
[5] J. Rosenberg and H. chulzrinne, "Reliability of provisional responses in the Session Initiation Protocol (SIP)," *IETF RFC 3262*, June 2002.
[6] M. Govind, S. Sundaragopalan, K.S. Binu, and S. Saha, "Retransmission in SIP over UDP - Traffic Engineering Issues," in *Proceedings of International Conference on Communication and Broadband Networking*, Bangalore, May 2003.
[7] E.M. Nahum, J. Tracey, and C.P. Wright, "Evaluating SIP server performance," in *Proceedings ACM SIGMETRICS*, San Diego, CA, US, 2007, pp. 349–350.
[8] Y. Hong, C. Huang, and J. Yan, "Analysis of SIP Retransmission Probability Using a Markov-Modulated Poisson Process Model," in *Proceedings of IEEE/IFIP Network Operations and Management Symposium*, Osaka, Japan, April 2010, pp. 179–186.
[9] E. Noel and C.R. Johnson, "Initial simulation results that analyze SIP based VoIP networks under overload," in *Proceedings of 20th International Teletraffic Congress*, 2007, pp. 54-64.
[10] E. Noel and C.R. Johnson, "Novel Overload Controls for SIP Networks," in *Proceedings of 21st International Teletraffic Congress*, 2009.
[11] R.P. Ejzak, C.K. Florkey, and R.W. Hemmeter, "Network Overload and Congestion: A comparison of ISUP and SIP," *Bell Labs Technical Journal*, **9**(3), 2004, pp. 173–182.
[12] V. Hilt and I. Widjaja, "Controlling Overload in Networks of SIP Servers," in *Proceedings of IEEE ICNP*, Orlando, Florida, October 2008, pp. 83-93.
[13] C. Shen, H. Schulzrinne, and E. Nahum, "SIP Server Overload Control: Design and Evaluation," in *Proceedings of IPTComm*, Heidelberg, Germany, July 2008.
[14] C. Shen and H. Schulzrinne, "On TCP-based SIP Server Overload Control," in *Proceedings of IPTComm*, Munich, Germany, August 2010.
[15] M. Ohta, "Overload Control in a SIP Signaling Network," in *Proceeding of World Academy of Science, Engineering and Technology*, Vienna, Austria, March 2006, pp. 205—210.
[16] A. Abdelal and W. Matragi, "Signal-Based Overload Control for SIP Servers," in *Proceedings of IEEE CCNC*, Las Vegas, NV, January 2010.
[17] "SIP Express Router" http://www.iptel.org/ser/.
[18] T. Warabino, Y. Kishi, and H. Yokota, "Session Control Cooperating Core and Overlay Networks for "Minimum Core" Architecture," in *Proceedings of IEEE Globecom*, Honolulu, Hawaii, December 2009.

[19] J. Sun, R.X. Tian, J.F. Hu, and B. Yang, "Rate-based SIP Flow Management for SLA Satisfaction," in *Proceedings of 11th International Symposium on Integrated Network Management* (*IFIP/IEEE IM*), New York, USA, June 2009, pp. 125-128.
[20] V. Gurbani, V. Hilt, and H. Schulzrinne, "Session Initiation Protocol (SIP) Overload Control," *IETF Internet-Draft*, draft-ietf-soc-overload-control-02, February 2011.
[21] J. Rosenberg, "Requirements for Management of Overload in the Session Initiation Protocol," *IETF RFC 5390*, December 2008.
[22] W. R. Stevens, *TCP/IP Illustrated*, Volume 1, Addison-Wesley, Boston, 1994.
[23] Y. Hong, O. W. W. Yang, and C. Huang, "Self-Tuning PI TCP Flow Controller for AQM Routers With Interval Gain and Phase Margin Assignment," in *Proceedings of IEEE Globecom*, Dallas, TX, U.S.A, November 2004, pp. 1324-1328.
[24] R. Ando, "Internet phone and video service Skype went down in a global service outage," Reuters News, December 22nd, 2010. http://www.reuters.com/article/idUSTRE6BL47520101222
[25] Y. Hong, C. Huang, and J. Yan, "Modeling and Simulation of SIP Tandem Server with Finite Buffer," *ACM Transactions on Modeling and Computer Simulation*, **21**(2), February 2011.
[26] Y. Liu, F. L. Presti, V. Misra, D. F. Towsley, and Y. Gu, "Scalable fluid models and simulations for large-scale IP networks," ACM *Transactions on Modeling and Computer Simulation*, **14**(3), July 2004, pp. 305–324.

**Yang Hong** received his Ph.D. degree in electrical engineering from University of Ottawa, Ottawa, Canada. His research interests include SIP overload control, Internet congestion control, modeling and performance evaluation of computer networks, and industrial process control.

**Changcheng Huang** received his B.Eng. in 1985, and M.Eng. in 1988, both in Electronic Engineering from Tsinghua University, Beijing, China. He received a Ph.D. degree in Electrical Engineering from Carleton University, Ottawa, Canada in 1997. From 1996 to 1998, he worked for Nortel Networks, Ottawa, Canada where he was a systems engineering specialist. He was a systems engineer, and network architect in the Optical Networking Group of Tellabs, Illinois, USA during the period of 1998 to 2000. Since July 2000, he has been with the Department of Systems and Computer Engineering at Carleton University, Ottawa, Canada where he is currently an associate professor.

Dr. Huang won the Canada Foundation for Innovation (CFI) new opportunity award for building an optical network laboratory in 2001. He was an associate editor of IEEE COMMUNICATIONS LETTERS from 2004 to 2006. He is currently a senior member of IEEE.

**James Yan** is currently an adjunct research professor with the Department of Systems and Computer Engineering, Carleton University, Ottawa, Canada. Dr. Yan received his B.A.Sc., M.A.Sc., and Ph.D. degrees in electrical engineering from the University of British Columbia, Vancouver, Canada. From 1976 to 1996 with Bell-Northern Research (BNR) and from 1996 to 2004 with Nortel, he was a telecommunications systems engineering manager responsible for projects in performance analysis of networks and products, advanced technology research and assessment, planning new network services and architectures, development of network design methods and tools, and new product definition. From 1988 to 1990, he participated in an exchange program with the Canadian Federal Government, where he was project prime for the planning of the evolution of the nationwide federal government telecommunications network. Dr. Yan is a member of IEEE and Professional Engineers Ontario.

# Improving the Resilience of Transport Networks to Large-scale Failures

Juan Segovia, Pere Vilà, Eusebi Calle, and Jose L. Marzo
Institute of Informatics and Applications (IIiA), University of Girona, Spain
Email: {juan.segovia, pere.vila, eusebi.calle, joseluis.marzo}@udg.edu

*Abstract*— **Telecommunication networks have to deal with fiber cuts, hardware malfunctioning and other failures on a daily basis, events which are usually treated as isolated and unrelated. Efficient methods have been developed for coping with such common failures and hence users rarely notice them. Although less frequently, there also arise cases of multiple failures with catastrophic consequences. Multiple failures can occur for many reasons, for example, natural disasters, epidemic outbreaks affecting software components, or intentional attacks. This article investigates new methods for lessening the impact of such failures in terms of the number of connections affected. Two heuristic-based link prioritization strategies for improving network resilience are proposed. One strategy is built upon the concept of betweenness centrality, while the second is based on what we call the observed link criticality. Both strategies are evaluated through simulation on a large synthetic topology that represents a GMPLS-based transport network. The provisioning of connections in a dynamic traffic scenario as well as the occurrence of large-scale failures are simulated for the evaluation.**

*Index Terms*— **Network resilience, Large-scale failures, Multiple failures, GMPLS, Link criticality.**

## I. Introduction

Today's data transport networks are designed to withstand several types of failures, namely accidental or intentional fiber cuts, loss of switching capabilities caused by power outages, malfunctioning due to equipment aging, and even operator mistakes. This ability to maintain service continuity in the presence of failures owes to efficient recovery techniques incorporated in their design, as well as to diverse technologies that have been developed to that end. Several methods and techniques are reported in the literature for dealing with failures [1], [2]. The fundamental underlying idea is that of redundancy, whereby spare resources come into operation when the active one fails.

The majority of the approaches to network recovery assume that at any given time, only one failure is outstanding, which is known as the single-failure assumption [3]. However, networks are equally prone to multiple-failure events, that is, the concurrent failure of several communication elements. For instance, earthquakes, flooding and natural disasters have the potential to disrupt a large number of network elements simultaneously.

Although large-scale failure events may be relatively rare, that fact does not lessen the economic loss they cause, or the disruption the can bring onto thousands or even millions of users. Unfortunately, in such failure scenarios the redundancy-based recovery techniques that are effective under the single-failure assumption are not suitable anymore, simply because the cost of implementing massive redundancy for rarely occurring events is prohibitive [4].

Multiple failures has been studied in the context of complex networks, graph theory and epidemics for some time now, see for example [5], [6] and [7] and the references therein. However, in the context of the reliability of transport networks, which is our focus here, the number of reported research is fewer, much more recent and almost always concerned with multilayer networks, where several failures visible in an upper layer are all caused by exactly one failure at the physical layer.

In this article, we consider that a certain number of physical links which represent a large portion of the topology fail concurrently, and study how such large-scale failure affects the network's resilience. The elements that fail do it randomly, without any constraint on locality. Our network scenario corresponds to a GMPLS-based transport network with dynamic traffic, where end-to-end connection is the unit of service. We propose and evaluate through simulations two strategies that can be used for improving resilience from the perspective of the number of connections that survive the failure.

In the next section, we review some recent instances of large-scale failures that gained world-wide notoriety. Section III presents basic concepts of network resilience and a review of the relevant literature. Our two proposed strategies for improving resilience is in Section IV, and the simulations carried out are described in Section V. Section VI presents the results and, finally, Section VII gives concluding remarks.

## II. Network Vulnerability to Large-scale Failures

Most of the research in survivable optical networks assume that failures occur independently from one another, thus instances of failures such as fiber cuts and node malfunctioning are usually modeled as isolated and unrelated

This paper is based on "A Heuristic Analysis of Resilience to Multiple Failures in GMPLS Networks," by J. Segovia, E. Calle P. Vilà and J. L. Marzo which appeared in the Proceedings of the 2010 Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS), Ottawa, Canada, July 2010. © 2010 IEEE.


doi:10.4304/jnw.7.1.63-72

events. Furthermore, as multiple failures are considered possible but rare [8], the focus tends to be on single failures, and on single link failures in particular, with only a few studies tackling the design of networks capable of withstanding up to double link failures. Nonetheless, one specific form of multiple link failure that attracted much attention is that which results from damages to physical structures, such as ducts, that are shared by otherwise unrelated fiber links. The concept of Shared-Risk Link Groups [9] captures this situation and has been used extensively in network survivability.

However, several more disrupting failures can be found in the real world. These include the ones in which the malfunctioning cover a large geographical area, thus affecting several completely unrelated network elements simultaneously, where unrelated means different network operators, countries or users. Root causes of such large scale failures are typically natural disasters, but can also be virus or worms outbreaks as well as intentional attacks. One recent example is the 2006 earthquake in the Taiwan area. The damage it caused is detailed in [10]. Several submarine cables were broken, and the communication infrastructure of countries in the region suffered either complete interruption or serious disruption for several days. Although backup resources (multiple fiber cores installed together) were in place, and automatic restoring procedures were activated, the former proved useless as the earthquake affected them as well, and the latter caused even more trouble due to limitations in the management system that precluded it from fully handling the multilayer network, which ultimately forced human intervention to complete the repairing.

Hurricane Katrina is also a recent example of a natural disaster that had important consequences on the communication infrastructure of the affected area, which caused damages to telecommunication networks worth several billion dollars [11], [12]. Lastly, and out of the category of natural disasters, is the politically-motivated attack on web sites of Estonian government and businesses in 2007, which crippled the country's network for days.

What all these examples have in common is the inability of the network to cope with pervasive failures, although they are quite capable of handling isolated cases of failure of links, nodes and other communication equipment that happens continually. However, such failures go unnoticed by users thanks to the recovery mechanisms put in place.

## III. Basic Concepts of Network Resilience

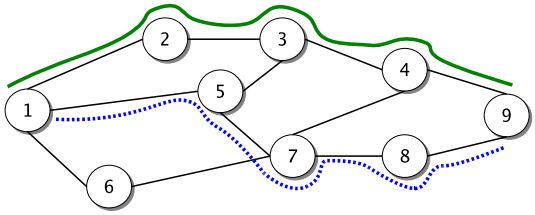
This section first gives an overview of basic protection schemes in transport networks, highlight their efficacy for dealing with single failures and their fundamental limitations to handle efficiently multiple failures. Then a review of network robustness metrics found in the literature is presented and their applicability to large-scale failures is discussed.

### *A. Resilience through Redundancy: Benefits and Limitations*

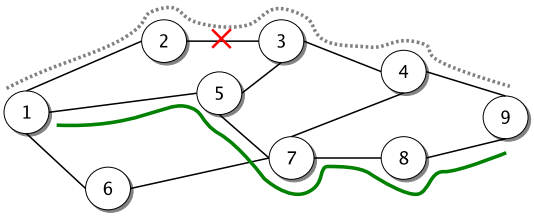
Redundancy is key for improving reliability. On one hand, physical components (e.g., optical cross-connects) can be deployed in redundant configurations, so that spare components come into operation when a failure occurs. Likewise, specific spans (links) or segments can be protected by provisioning additional (e.g., redundant) path(s) so that traffic can be switched to such alternate path(s) when the protected element fails. Many recovery techniques have been proposed and several of them have gained wide acceptance in the industry. A summary of the ones more relevant to next-generation optical networks can be found in [13] and [1]. Two basic recovery schemes based on path protection are Dedicated Path Protection (DPP) and Shared Path Protection (SPP), devised essentially to protect against single failures.

Fig. 1 illustrates the basic operation of DPP, which can be implemented through any technology that provides path protection, for example SONET/SDH 1+1 (or 1:1) Automatic Protection Switching or MPLS Fast Reroute [14]. In this example, one (bidirectional) connection exists between nodes 1 and 9. With DPP, two (link- or node-) disjoint paths are exclusively assigned to the connection, one acting as the working path and the other as the backup path. Thus, it can survive any single (link or intermediate node) failure. Now let's assume that links 2–3, 3–5 and 5–7 are geographically close to each other and that an event such as an earthquake affects them all so that their failure are concurrent in time. Although the network would continue being fully connected, DPP loses its efficacy because both paths are failed and, by design, dynamic recovery is not attempted. Note that the failed elements need not be geographically related; any failure that touches both paths will equally suffice. A possible solution is to assign more than two paths, say $k$ disjoint paths. However, finding $k$ such paths for arbitrary source-destination pairs is not always possible, as it depends on the topology and even on the routing strategy employed. Furthermore, even if there were, they might not comply with QoS constraints, for example on maximum hop count [15]. Note that in our example, we could not have three node-disjoint paths between nodes 1 and 9.

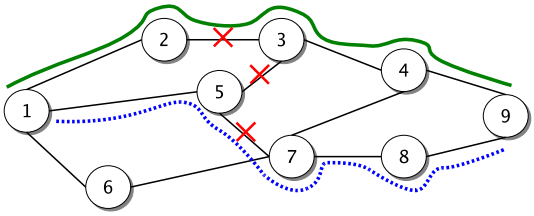
Nevertheless, DPP offers the fastest recovery and the best protection in single-failure scenarios. In fact, it works as expected even when there exist more than one concurrent failure network wide, for it operates at the connection level. The drawback is the total capacity required to support it, which is at least more than twice of what is necessary for unprotected connections. This situation can be alleviated by using SPP, which, in contrast to DPP, shares a single backup path among $n > 1$ separate connections, which can lead to substantial capacity savings. The sharing groups must be carefully set up, however, so that the working paths in each group are disjoint, a computationally complex problem especially in dynamic traffic scenarios [16]. If it is assumed that two or more working paths in the same set rarely fail at the same




(a) Failure-free state.




(b) Link 2–3 fails. Traffic is diverted to the backup path.




(c) Links 2–3, 3–5 and 5–7 fail. Both paths are affected (service fails).

Figure 1: Basic operation of DPP. A working path (solid line) and a backup path (dotted line) are provisioned for a connection between nodes 1 and 9.

time, the protection will be effective. However, if they do fail concurrently, one or more connections will not be recovered. Thus, once the setting changes from single failure to multiple failures, the sharing groups have to be smaller to reduce risks, and so the capacity savings diminishes as well.

DPP and SPP are just two of many existing protection schemes. However, they illustrate very well that it is possible to have effective redundancy-based protection at varying degrees of resource consumption for the dominant form of failure (i.e., single failures). At the same time, it is clear that it is not economically viable to implement massive redundancy for rarely occurring events (multiple or large-scale failures) however devastating their effects might be. Recent proposal of research in this area exploit the concept of overlay networks, e.g. [4], [17], though they target the commodity Internet instead of transport networks. Our approach in this article is to analyze the interaction of the topology with the routing to discover the network elements (e.g., links) whose failure would cause the largest service disruption and from there on devise strategies of extra protection.

### B. Network Robustness against Large-scale Failures

In networking, resilience is defined as the ability of the network to provide and maintain an acceptable level of service in the presence of faults [18]. The robustness of a network is a measure of its resilience, which can be appraised through a variety of approaches and metrics. One classical approach is through the graph-theoretical concept of *connectivity*, which determines the ability of a given topology to keep all its pair of nodes accessible via some path, as one or more nodes or links are removed. The level of disruption can also be studied by analyzing the flow variation in the network as a consequence of failures. Furthermore, the interest may be in evaluating the vulnerability to specific forms of failures, for example random failures versus attacks [19]. A more recent proposal is *elasticity*, which relates total throughput to node removal in complex networks [20].

Another dimension of the problem of assessing network robustness regards geographically correlated failures, which cause disruption to a large number of network elements (fiber ducts, regenerators, complete nodes, etc.) around a specific location. Correlated link failures caused by random line-cuts are studied in [21] and resilience metrics are proposed based on connectivity, more specifically, on all-terminal reliability and average two-terminal reliability. On the other hand, in [22] the focus is in identifying the most vulnerable areas of a given physical network, that is, the locations where large-scale failures would provoke a severe reduction of capacity and connectivity in the whole network.

### C. Limitations of these metrics

The metrics just discussed, while clearly useful in the general case of topology analysis, show some limitations, however, when applied to data transport networks. This stem from the fact that they largely depend on topological features, thus disregarding traffic dynamics and operational constrains such as network load, heterogeneity in link capacity, and routing strategy. Of the metrics mentioned, only elasticity takes into account some of these aspects. Moreover, when pre-planned protection is the only mechanism considered, as we do here, network connectivity is only marginally useful. As we have already pointed out, it is quite possible that the topology remains connected after a multiple failure event, but even so the number of lost connections reaches unacceptable levels. Thus, instead of observing the state of the abstract topology, it might be better to wonder about the fate of the units of service of the transport network, i.e., connections. Such approach is also taken in [23], but there the elements under consideration are exclusively nodes which become partially or totally disabled as a consequence of failures described by an epidemic model.

In this article we propose using an alternate measure of robustness, namely, the number of connections that survive a large-scale failure or attack. This approach is appropriate for connection-oriented networks precisely because each connection embodies in its path the influence of the structural properties of the topology on the traffic flow, as well as the network operator's policy on resource allocation, as implemented through routing.

## IV. Proposed Heuristics for Link Prioritization

Graph-theoretical studies on topology resilience usually focus on events of vertex removal, which translated to networking would mean complete node failure. However, more often than not, a network node fails only partially. In fact, the failure of one or more links attached to a node can be considered a partial node failure, altogether a much more frequent event. Therefore, from now on we will focus on link failure, which we assume as encompassing cable cuts as well as the malfunctioning of line cards, regenerators or any component necessary for a successful communication between two adjacent nodes, including software components running on the nodes. Naturally, the failure of all the links belonging to a node amounts to a full node failure.

Suppose that all the links in a network have equal probability of being hit by a certain failure, and that it is possible to make them invulnerable at a fixed cost per link. Suppose also that several links can be affected at once, and that a budget is available for shielding a limited number of them so as to reduce the total number of affected connections when such a large-scale failure event occurs. Which links should be part of this set of invulnerable links? What is the criterion for selecting them?

The combinatorial and non-deterministic nature of this problem make it difficult to offer a computationally simple and exact solution, and thus call for approximate solutions instead. In this section we propose two heuristic-based approaches to the problem. The first one takes advantage of the concept of betweenness centrality, which from now on we will identify as EdgeBC. The second is based on link usage statistics collected as part of the connection set-up phase, identified from now on as OLC for Observed Link Criticality.

In both cases, the idea is that they produce a prioritized list of links that we can choose from to satisfy the maximum number of links that are to become invulnerable. We proceed now to explain both approaches and highlight their strengths and limitations.

### A. EdgeBC: *The betweenness centrality approach*

Betweenness centrality determines how often a node or link on a given network topology lies along the shortest path between all possible pair of nodes [24]. More formally, the edge betweenness centrality $B_e$, i.e., the betweenness centrality of a link $e$, is defined as follows:

$$B_e = \sum_{s,t \in V, s \neq t} \frac{\sigma(s,t|e)}{\sigma(s,t)} \qquad (1)$$

where $V$ and $E$ are the set of nodes and the set of links, respectively, $\sigma(s,t)$ is the number of shortest paths that exist between nodes $s$ and $t$, and $\sigma(s,t|e)$ is the number of shortest paths between nodes $s$ and $t$ that use the link $e$, with $e \in E$. $B_e$ can be normalized so that a value close to 1 would mean that link $e$ is present in almost all the shortest paths in the network. Likewise, a value close to 0 would mean that its role as intermediary in the communication is marginal. Note that the above definition assumes the use of *shortest paths*, and thus it is sometimes called the Shortest-Path Betweenness Centrality. Although the cost function can vary, it is usually set to hop count.

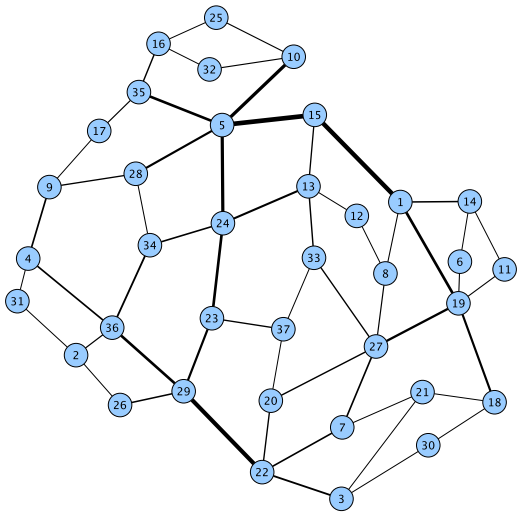



Figure 2: The COST266 topology [25]. Link thickness indicates link importance in terms of edge betweenness centrality.

Betweenness centrality has been used in network vulnerability studies to measure the impact of attacks that target those nodes which seem to play a bigger role as communication mediators in the topological sense, see for example [19] and the references therein. That approach is the converse of what we propose here, which is to explore proactive link protection measures against random failures.

Fig. 2 is a visual representation of the betweenness centrality of the links of the well-known COST266 reference topology[1], which has 37 nodes and 57 undirected links. The higher the value of $B_e$, the thicker the link line is. One can see that thick lines appear scattered on the topology, so that neither node position (periphery versus "core") nor degree of connectivity warrant a high $B_e$.

Given that $B_e$ depends only on the topology, it can be computed just once as long as the topology remains unchanged. However, this very fact is also the source of its weakness, for it cannot fully take into account some fundamental aspects of an operational network. For instance, from the point of view of routing, the network topology suffers recurrent virtual and transient changes. There is a virtual link removal when the corresponding residual capacity reaches zero. Conversely, the link is re-inserted later on when the connections that use it are torn down. Therefore, the shortest path at any instant depends on the network state: one particular connection request might be assigned the ideal shortest path, but the next one might not. In contrast, the definition in Eq. 1

[1] Available at `http://sndlib.zib.de/`

assumes unlimited capacity and thus immutable shortest paths. Consequently, for real-world data networks, $B_e$ can only give an approximate centrality. Further complications come from the variation in link capacity (not all links have the same capacity) and from the imbalances in the traffic matrix (the volume of communication between node pairs need not be the same). Though it would be feasible to recompute the betweenness centrality at each virtual topology change –its running time is in $O(|V||E|+|V|^2 log\ |V|)$ [26]– it would not fully resolve the divergence between the real centrality and the estimation based on static data.

### B. OLC: *The Observed Link Criticality approach*

This measure is based on the concept of criticality in minimum-interference routing [27]. The difference is that, instead of relying on an approximation based on static data, we can directly take advantage of dynamic information about resource usage that can be collected in the GMPLS control plane. Specifically, each link $e$ can have associated to it a counter $c_e$ for the number of connection paths that go through it, and that counter can be updated as connections are accepted and released. That way, the relative importance of $e$ is $M = \frac{c_e}{N}$, where $N$ is the number of active connections at a certain instant. From this, an estimation of the link importance can be obtained as a simple moving average of $M$:

$$I_e = \frac{M + M_{-1} + M_{-2} + \ldots + M_{-k-1}}{k} \qquad (2)$$

where $k$ is a constant for the number of consecutive samples to use.

The disadvantage of this approach compared to EDGEBC is that the network must already be in operation, and preferably in a steady state, before it can be applied.

## V. SIMULATION OF FAILURES FOR PERFORMANCE COMPARISON

In this section, we detail the steps carried out to compare through simulation the performance of EDGEBC and OLC in terms of their ability to minimize the number of affected connections when a large-scale failure occurs.

### A. Simulation parameters

The topology used in this work is shown in Fig. 3. This is a synthetic topology that represents a large transport network consisting of 222 nodes and 371 links, and share structural properties with reference topologies found in the Survivable Network Design Library (SNDlib) [25]. The size differs significantly, however, as this topology is more than three times larger than the ones found in that repository. The main properties of this topology are shown in Table I. As it represents the physical topology of a transport network, the nodal degree range is 2 to 5, the dominant degree being 3, as can be seen in Fig 4. For simplicity, we assume that links are capable of carrying an arbitrary number of Label Switched Paths (LSPs) [28] as long as free capacity is available, and that all nodes support full wavelength conversion. To introduce a minimum degree of heterogeneity in capacity, links have either $C$ or $2C$ units of total capacity. Half of them belong to the first group and the rest to the second, where the membership to either set was decided on a random basis.

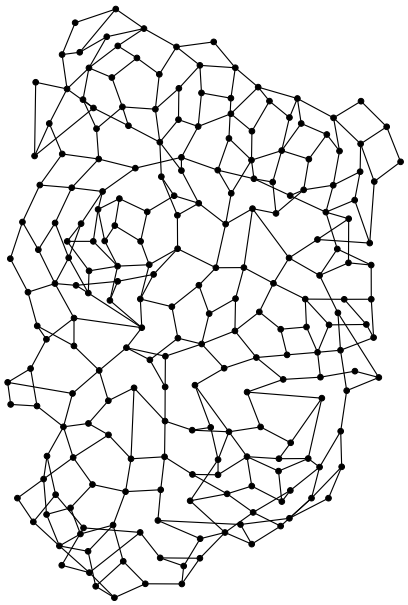


Figure 3: The synthetic topology used in the simulations

TABLE I.: Main properties of the topology used in the simulations

| Property | Value |
|---|---|
| Number of nodes | 222 |
| Number of links | 371 |
| Network diameter | 20 hops |
| Average shortest path | 9.06 hops |
| Average nodal degree | 3.3 |

In order to simulate the provision of service in the network and the occurrence of large-scale failures, an event-driven simulator that reproduces the process of route selection in a path-oriented transport network was developed. The simulator handles the reception of connection requests between node pairs, triggers and coordinates the proper routing and capacity allocation based on the demand, keeps track of the usage of resources (the residual capacity on each link), releases connections when their holding time has expired, and collects the required statistical data.

With respect to the traffic demand, the simulator accepts a series of connection requests between randomly-selected source and destination nodes, which arrive according to a Poisson process. As we are interested in a dynamic traffic scenario, the capacity allocated to an accepted connection, whose holding time is an exponentially distributed random variable, is released as soon as

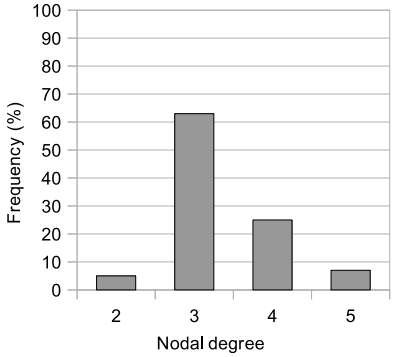



Figure 4: Frequency distribution of nodal degrees

the connection terminates. The aggregated traffic between any two pair of nodes is a fixed value or zero, meaning that either they do not communicate, or contribute the same amount of traffic as other pairs. This traffic matrix is randomly generated. Any demand whose source and destination are at four hops or less is disregarded (and not counted as rejected), i.e., we discard connections corresponding to "local traffic". The capacity requested by connections is a uniformly distributed random variable in the range 1–10 units.

In accordance with the assumptions made in Section IV, a minimum-hop routing algorithm is used. Links that do not have enough residual capacity to satisfy the arriving demand are filtered out before the exploration begins. That is, the routing employs a Constrained Shortest Path First algorithm, where the constraint is the minimum capacity required per link.

To avoid creating unrealistically long paths, any request whose feasible path exceeds 24 hops is also rejected. Note that the average minimum path length in this topology is about 9 hops, while the diameter is 20 (see Table I). The blocking ratio is approximately 0.01 in all the experiments reported. For each accepted connection, one path from source to destination is created. As no protection is provided at the connection level, no additional path is created in addition to this working path.

### *B. Simulating a Large-scale Failure*

Exactly one failure event is triggered during the simulation, whose time is chosen randomly once a stable state had been reached. In contrast, the extent of the failure and the size of the set of invulnerable links are user-provided input data, expressed as a percentage of the total number of links in the topology.

To emphasize that this work focuses on far-reaching failures, both in geographical coverage and size, from now on we refer to the extent of the failure as the *infection level* and, by analogy, to the percentage of invulnerable links as the *immunization level*.

The simulations were performed with infection levels 5, 10, and 20, and immunization levels 10, 20, and 30. Thus, one simulation run corresponds to a specific combination of: a) infection level, b) immunization level, and c) procedure for the selection of invulnerable links, which for simplicity is called *immunization strategy* from now on.

For comparison purposes, a procedure based on random selection of the links to be considered invulnerable was also added, so that the immunization strategies are three, as follows:

- RANDOM: failed links as well as invulnerable links are chosen randomly.
- EDGEBC: failed links are chosen randomly. Link immunization is based on their edge betweenness centrality.
- OLC: failed links are also chosen randomly. The immunization is based on average Observed Link Centrality.

This gives 27 cases in total. The values presented in Section VI are the average of 30 runs per case, each one processing a new demand set consisting of 95000 randomly-generated connection requests.

From each run, the following results are obtained, which constitute the figures of merit for comparing the performance of the immunization strategies:

1) The percentage of active connections affected by the failure.
2) The frequency distribution of path length of the affected connections.

A connection is considered affected if: a) its duration has not yet reached its declared lifetime, and b) at least one link included in its path was hit by the failure.

## VI. SIMULATION RESULTS AND DISCUSSION

The objective of these simulations is to compare the effectiveness of the immunization strategies proposed. However, there is the question whether these strategies provide any improvement in resilience at all. To answer this question and put in perspective the value of the proposed strategies, we also simulated the *zero immunization* case. As can be seen in Fig. 5, the result is striking, since even when the infection level is minimal (5%), almost half of the connections (42%) active at the time the failure was triggered were affected. The percentage of affected connections continues to grow as the infection level increases, but a moderation is observed in its progression. Nearly all connections (≈90%) are affected with 20% of infection. Thus, we consider that the infection scenarios worth exploring are between 5% and 20%.

Table II shows the distribution of the frequency of connection path length of one representative simulation run. For the reasons explained in Section V, the path length range is 5–24. Fig. 6 and Fig. 7 discriminate by path length the percentage of connections affected at infection levels 5 and 20, respectively. Each sub-figure presents the performance of one specific immunization strategy at immunization levels 10, 20, and 30.

It can be observed that RANDOM is essentially insensitive to the immunization level in both infection scenarios.

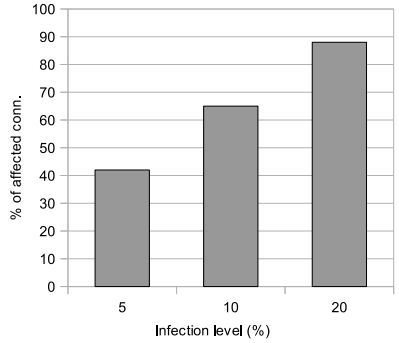



Figure 5: Affected connections at three infection levels when no immunization is in place

TABLE II.: Frequency distribution of connection path length of a representative simulation run

| Path length | Frequency (%) | Cumulative frequency |
|---|---|---|
| 5 | 8.5 | 8.5 |
| 6 | 9.4 | 17.9 |
| 7 | 10.2 | 28.1 |
| 8 | 10.3 | 38.4 |
| 9 | 10.2 | 48.6 |
| 10 | 9.4 | 58.0 |
| 11 | 8.4 | 66.4 |
| 12 | 7.6 | 74.0 |
| 13 | 6.4 | 80.4 |
| 14 | 5.2 | 85.6 |
| 15 | 4.3 | 89.9 |
| 16 | 3.5 | 93.4 |
| 17–24 | 6.6 | 100.0 |

It is interesting to note in Fig. 6c that there exists a case in which the immunization level is six times the infection level, but even then the effect is negligible. This behavior is similar for all path lengths. Only for the longest paths in Fig. 6c does performance vary with respect to the immunization level, which is due to the fact that the number of connections of such long lengths is very small compared to the rest (see Table II).

In the rest of the cases, the behavior clearly depends on the level of infection and immunization. For instance, with 20% of infection (Fig. 7) and 10% of immunization, EDGEBC and OLC offer similar results, outperforming RANDOM in that case as well as in almost all the others. However, when the immunization level is raised to 30, OLC is the one whose reaction is more visible, producing the lowest values for the number of affected connections. This effect is observed in the three infection scenarios.

Table III summarizes the performance of the three strategies. The column "% Aff. conn." gives the percentage of connections affected by the failure. The remaining columns put connections into three groups based on their path lengths, and show what proportion of each group was adversely affected. The groups are as follows: a) short (5–8 hops), b) medium (9–18 hops), and c) long (19–24 hops). Each value is an average of the individual results in the range. Every combination of infection level, strategy and immunization level considered in this work has an entry in the table.

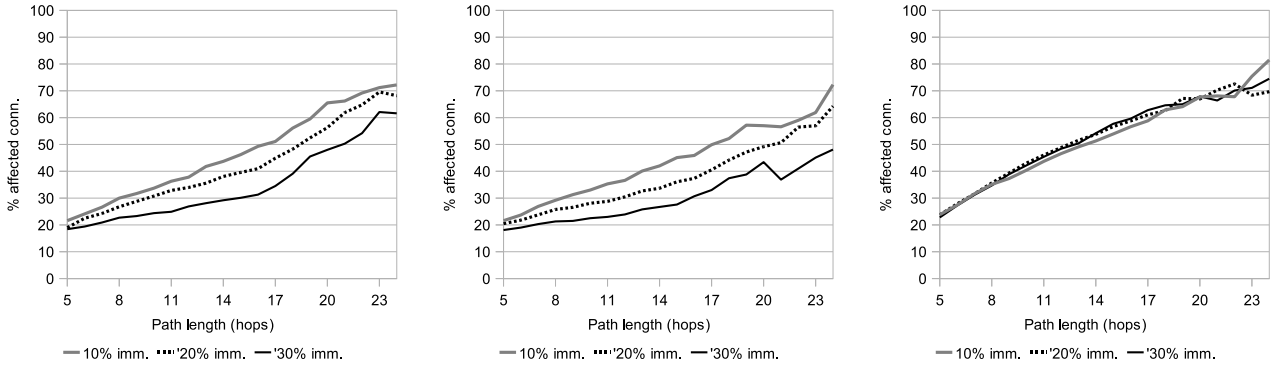
TABLE III.: Performance of the three strategies, discriminated by groups of path length, at selected combinations of infection and immunization level

| % Infect. | Strategy | % Immunization | % Aff. conn. | Path length | | |
|---|---|---|---|---|---|---|
| | | | | Short | Medium | Long |
| 5 | EDGEBC | 10 | 34.6 | 25.6 | 42.8 | 67.3 |
| | | 20 | 30.8 | 23.1 | 37.3 | 62.1 |
| | | 30 | 25.1 | 20.4 | 29.2 | 53.6 |
| | OLC | 10 | 33.7 | 25.4 | 41.1 | 60.7 |
| | | 20 | 28.6 | 22.9 | 33.8 | 54.1 |
| | | 30 | 23.4 | 19.7 | 27.2 | 42.2 |
| | RANDOM | 10 | 40.5 | 29.5 | 50.1 | 70.8 |
| | | 20 | 42.0 | 29.7 | 52.1 | 69.1 |
| | | 30 | 41.6 | 29.0 | 52.4 | 69.2 |
| 10 | EDGEBC | 10 | 54.5 | 43.1 | 64.5 | 86.2 |
| | | 20 | 49.2 | 40.0 | 57.3 | 82.5 |
| | | 30 | 42.7 | 35.1 | 50.0 | 75.8 |
| | OLC | 10 | 53.9 | 45.0 | 65.4 | 85.1 |
| | | 20 | 47.5 | 39.0 | 55.7 | 77.1 |
| | | 30 | 40.2 | 34.3 | 45.6 | 68.2 |
| | RANDOM | 10 | 64.1 | 49.1 | 76.2 | 90.4 |
| | | 20 | 64.5 | 50.2 | 75.8 | 90.1 |
| | | 30 | 65.2 | 50.9 | 76.6 | 90.9 |
| 20 | EDGEBC | 10 | 79.9 | 69.9 | 88.2 | 98.5 |
| | | 20 | 71.6 | 62.6 | 79.6 | 96.0 |
| | | 30 | 65.2 | 56.6 | 73.2 | 93.9 |
| | OLC | 10 | 78.8 | 70.4 | 87.9 | 96.4 |
| | | 20 | 70.7 | 61.8 | 78.6 | 94.7 |
| | | 30 | 62.2 | 55.4 | 68.9 | 90.0 |
| | RANDOM | 10 | 87.1 | 76.7 | 94.6 | 99.4 |
| | | 20 | 87.4 | 77.0 | 94.9 | 99.6 |
| | | 30 | 86.4 | 75.9 | 94.1 | 99.0 |

Fig. 8 shows graphically what happens from the perspective of connection path length when the infection level is 10 and the immunization level is 30. As can be seen, the difference between RANDOM and the other two is almost 20% for short paths. That difference jumps to around 30% for paths of medium length, and shrinks back to around 20% for long paths.

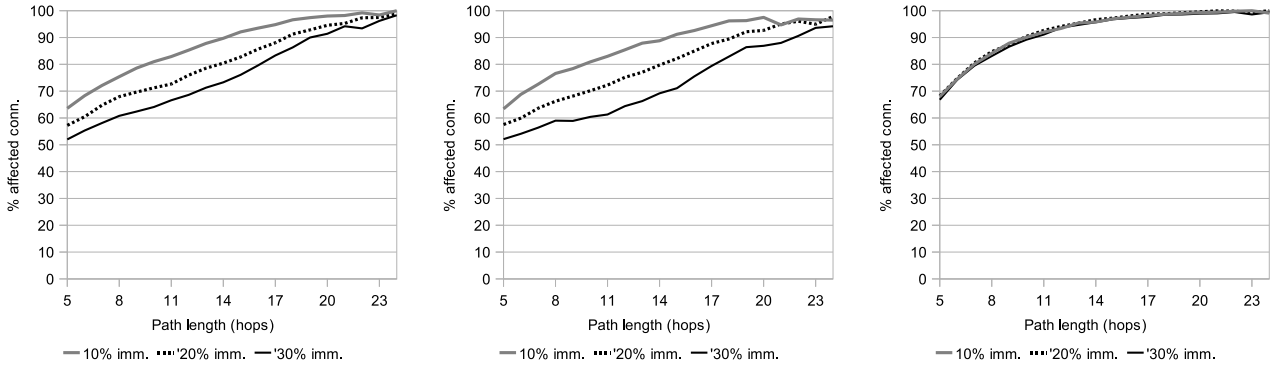
Additionally, the network's two-terminal reliability obtained after the failure event is presented in Table IV. The values correspond to selected levels of immunization and infection when the strategy is EDGEBC. We compute it as the ratio of the number of origin-destination pairs for which a path exists to the maximum possible number of paths. Thus, the closer this value to 1.0, the lower the network fragmentation is. As expected, our simulations confirm that this reliability measure is quite insensitive to the failure, for it remains very close to 1.0 in all cases, indicating almost full connectivity. That is, it does not reflect the degree of damage experienced by connections after the failure.

In summary, these results show the high sensitivity of a connection-oriented network to large-scale failures, because even a relatively small infection level (%5) causes disruption to almost half the connections in the studied scenario. Moreover, the RANDOM strategy, which implements a immunization without a specific criterion, shows




(a) EDGEBC (b) OLC (c) RANDOM

Figure 6: Performance comparison discriminated by path length at infection level=5%




(a) EDGEBC (b) OLC (c) RANDOM

Figure 7: Performance comparison discriminated by path length at infection level=20%

TABLE IV.: Average two-terminal reliability after the failure event. Strategy: EDGEBC

| Immunization (%) | Infection | |
|---|---|---|
| | 5% | 10% |
| 10 | 0.9994 | 0.9973 |
| 20 | 0.9985 | 0.9970 |

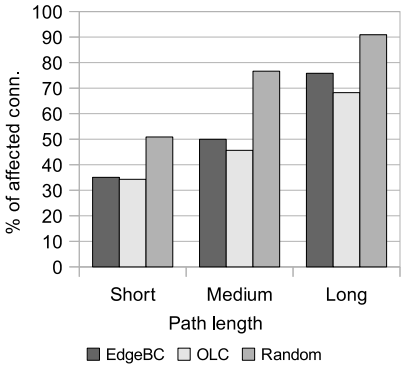



Figure 8: Affected connections under the three strategies when infection level=10% and immunization level=30%

that little or no benefit is obtained in terms of robustness by choosing links disregarding their role in the overall traffic flow.

With respect to the immunization strategies EDGEBC and OLC, results show that both are capable of minimizing the impact of these failures with the appropriate election of the immunization level. Of the two, we can see that OLC is the best performer, as with it the number of affected connections is lower and, at the same time, the number of surviving connections whose path lengths are medium and long is higher.

## VII. CONCLUSION

In this article, the problem of improving the resilience of large transport networks to pervasive failures has been addressed. The scenario is a GMPLS-based network subjected to a pervasive multiple failure event, where several links fail concurrently in random locations, provoking the loss of all the end-to-end connections passing through them. The ultimate aim is to identify elements (e.g., links) to which extra network protection can be applied so that the impact of such failure events, in terms of the number of connections affected, is minimized.

Two new strategies of link prioritization have been presented, whereby a subset of links are made invulnerable to the proposed type of failure. The first strategy uses the well-known measure of betweenness centrality, while the second is based on a new concept called the Observed Link Criticality. Both have been evaluated through extensive simulations. For comparison purposes, a third approach based on random selection of links was also included in the study.

The obtained results highlight how easily a large-scale failure event can seriously disrupt the operation of a connection-oriented network. At the same time, it shows that robustness metrics such as two-terminal reliability are not appropriate to assess the state of the network from the point of view of its ability to provide service (connections). For example, even when the network is suffering a major service degradation, the average two-terminal reliability reports a very high degree of connectivity. Results also show that the two proposed strategies succeed in decreasing the number of affected connections. The one based on observed link criticality offers better overall robustness and, at the same time, preserves a higher number of connections of medium and long path lengths.

## VIII. Acknowledgments

This work is partially supported by Spanish Ministry of Science and Innovation project TEC 2009-10724 and by Generalitat de Catalunya research support program (SGR-1202).

## References

[1] A. Haider and R. Harris, "Recovery techniques in next generation networks," *Communications Surveys and Tutorials, IEEE*, vol. 9, no. 3, pp. 2–17, 2007.

[2] P. Cholda, A. Mykkeltveit, B. Helvik, O. Wittner, and A. Jajszczyk, "A survey of resilience differentiation frameworks in communication networks," *Communications Surveys Tutorials, IEEE*, vol. 9, no. 4, pp. 32 –55, 2007.

[3] J.-P. Vasseur, M. Pickavet, and P. Demeester, *Network Recovery. Protection and Restoration of Optical, SONET-SDH, IP, and MPLS*. San Francisco, CA: Morgan Kaufmann Publishers, 2004.

[4] T. Horie, G. Hasegawa, S. Kamei, and M. Murata, "A new method of proactive recovery mechanism for large-scale network failures," in *Advanced Information Networking and Applications, International Conference on*. Los Alamitos, CA, USA: IEEE Computer Society, 2009, pp. 951–958.

[5] R. Pastor-Satorras and A. Vespignani, "Epidemic dynamics and endemic states in complex networks," *Phys. Rev. E*, vol. 63, no. 6, p. 066117, May 2001.

[6] M. Barthélemy, A. Barrat, R. Pastor-Satorras, and A. Vespignani, "Dynamical patterns of epidemic outbreaks in complex heterogeneous networks," *J. Th. Bio.*, vol. 235, p. 275, 2005.

[7] T. Tanizawa, G. Paul, R. Cohen, S. Havlin, and H. E. Stanley, "Optimization of network robustness to waves of targeted and random attacks," *Phys. Rev. E*, vol. 71, no. 4, p. 047101, Apr 2005.

[8] J. Zhang and B. Mukherjee, "A review of Fault Management in WDM Mesh Networks: Basic Concepts and Research Challenges," *IEEE Network*, vol. 18, no. 2, pp. 41–48, 2004.

[9] L. Shen, X. Yang, and B. Ramamurthy, "Shared risk link group (SRLG)-diverse path provisioning under hybrid service level agreements in wavelength-routed optical mesh networks," *IEEE/ACM Transactions on Networking (TON)*, vol. 13, no. 4, pp. 918–931, 2005.

[10] Y. Kitamura, Y. Lee, R. Sakiyama, and K. Okamura, "Experience with restoration of Asia Pacific network failures from Taiwan earthquake," *IEICE Transactions*, vol. 90-B, no. 11, pp. 3095–3103, 2007.

[11] S. Erjongmanee, C. Ji, J. Stokely, and N. Hightower, "Large-scale inference of network-service disruption upon natural disasters," in *Knowledge Discovery from Sensor Data*, ser. Lecture Notes in Computer Science, M. Gaber, R. Vatsavai, O. Omitaomu, J. Gama, N. Chawla, and A. Ganguly, Eds. Springer Berlin / Heidelberg, 2010, vol. 5840, pp. 134–153.

[12] G. O'Reilly, A. Jrad, R. Nagarajan, T. Brown, and S. Conrad, "Critical Infrastructure Analysis of Telecom for Natural Disasters," in *Telecommunications, Network Strategy and Planning Symposium, 2006. NETWORKS 2006. 12th International*. IEEE, 2006, pp. 1–6.

[13] W. Grover, J. Doucette, M. Clouqueur, D. Leung, and D. Stamatelakis, "New options and insights for survivable transport networks," *Communications Magazine, IEEE*, vol. 40, no. 1, pp. 34–41, Jan. 2002.

[14] P. Pan, G. Swallow, and A. Atlas, "Fast Reroute Extensions to RSVP-TE for LSP Tunnels," RFC 4090 (Proposed Standard), Internet Engineering Task Force, May 2005. [Online]. Available: http://www.ietf.org/rfc/rfc4090.txt

[15] P. Van Mieghem and F. Kuipers, "Concepts of exact QoS routing algorithms," *Networking, IEEE/ACM Transactions on*, vol. 12, no. 5, pp. 851–864, oct 2004.

[16] C. Ou, J. Zhang, H. Zang, L. Sahasrabuddhe, and B. Mukherjee, "New and improved approaches for shared-path protection in WDM mesh networks," *Journal of Lightwave Technology*, vol. 22, no. 5, p. 1223, 2004.

[17] K. Kim and N. Venkatasubramanian, "Assessing the impact of geographically correlated failures on overlay-based data dissemination," in *IEEE GLOBECOM 2010*, December 2010, pp. 1–5, (to appear).

[18] J. Sterbenz, D. Hutchison, E. Cetinkaya, A. Jabbar, J. Rohrer, M. Schöller, and P. Smith, "Resilience and survivability in communication networks: Strategies, principles, and survey of disciplines," *Computer Networks*, vol. 54, no. 8, pp. 1245–1265, 2010.

[19] T. Grubesic, T. Matisziw, A. Murray, and D. Snediker, "Comparative approaches for assessing network vulnerability," *International Regional Science Review*, vol. 31, no. 1, pp. 88–112, 2008.

[20] A. Sydney, C. Scoglio, M. Youssef, and P. Schumm, "Characterizing the Robustness of Complex Networks," *Int. J. Internet Technology and Secured Transactions*, vol. 2, no. 3/4, pp. 291–320, 2010.

[21] S. Neumayer and E. Modiano, "Network Reliability with Geographically Correlated Failures," in *INFOCOM, 2010 Proceedings IEEE*, Mar. 2010, pp. 1–9.

[22] S. Neumayer, G. Zussman, R. Cohen, and E. Modiano, "Assessing the vulnerability of the fiber infrastructure to disasters," in *INFOCOM*, 2009, pp. 1566–1574.

[23] E. Calle, J. Ripoll, J. Segovia, P. Vilà and, and M. Manzano, "A multiple failure propagation model in gmpls-based networks," *Network, IEEE*, vol. 24, no. 6, pp. 17–22, November-December 2010.

[24] L. C. Freeman, "A set of measures of centrality based upon betweenness," *Sociometry*, vol. 40, no. 1, pp. 35–41, 1977.

[25] S. Orlowski, R. Wessäly, M. Pióro, and A. Tomaszewski, "SNDlib 1.0–Survivable Network Design Library," *Networks*, vol. 55, no. 3, pp. 276–286, 2010.

[26] U. Brandes, "On variants of shortest-path betweenness

centrality and their generic computation," *Social Networks*, vol. 30, no. 2, pp. 136–145, 2008.

[27] E. Calle, A. Urra, J. Marzo, G.-S. Kuo, and H.-B. Guo, "Minimum interference routing with fast protection," *Communications Magazine, IEEE*, vol. 44, no. 10, pp. 104–111, Oct. 2006.

[28] E. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol Label Switching Architecture," RFC 3031 (Proposed Standard), Internet Engineering Task Force, Jan. 2001. [Online]. Available: http://www.ietf.org/rfc/rfc3031.txt

**Juan Segovia** is currently a PhD candidate at University of Girona, Spain. He received his bachelor's degree in Computer Science from the National University of Asunción in 1994, and a MPM degree in 2000 from the same university.

He was a lecturer at the National University of Asunción, Paraguay, 1997-2005. He is currently a member of the Broadband Communications and Distributed Systems group, and his research interests include protection and restoration of GMPLS-based optical networks, and reliability against large-scale failures in complex networks.

**Pere Vilà** received his PhD in Computer Science from the University of Girona in 2004.

He is a lecturer in the Department of Computer Architecture and Technology at the University of Girona, and a member of the Broadband Communications and Distributed Systems research group. His current research interests are in the fields of network management, routing and protection, and interdomain mechanisms.

Dr. Vilà has co-authored several papers in journals and international conferences and worked on several funded research projects.

**Eusebi Calle** received his doctorate degree in Computer Science from the University of Girona (UdG) in 2004.

He is currently an associated professor at the UdG. Since 1998, he is a member of the research and teaching staff in the Broadband Communications and Distributed System Group, where he develops his research in GMPLS fault management, routing and network science.

Dr. Calle has co-authored several papers in international journals and international conferences. He is also member of different TPC, and part of the Institute of Informatics and Applications at the UdG.

**Jose L. Marzo** received his Ph.D. degree in Industrial Engineering in 1997 from the University of Girona, Spain. Currently, he is a Full Professor at the Department of Electronics, Informatics and Automatics at the same university.

He leads the research group Broadband Communications and Distributed Systems. His research interests are in the fields of management and performance evaluation of communication networks, network management based on intelligent agents, MPLS and GMPLS, and distributed simulation. From 1978 to 1991, he was with Telefónica de España.

Dr. Marzo is a member of the IEEE Communications Society. He has participated in technical program committees and chairing sessions of several international conferences. He serves in the editorial board of the International Journal of Communications Systems. He has co-authored several papers published in international journals and presented in leading international conferences.

# Performance Evaluation of Efficient Solutions for the QoS Unicast Routing

A. Bellabas, S. Lahoud
Institut de Recherche en Informatique et Systèmes Aléatoires IRISA, Rennes, FRANCE
M. Molnár
Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier LIRMM, Montpellier, FRANCE
Email: {alia.bellabas, samer.lahoud}@irisa.fr, miklos.molnar@lirmm.fr

***Abstract*— Quality of Service (QoS) routing known as multi-constrained routing is of crucial importance for the emerging network applications and has been attracting many research works. This NP-hard problem aims to compute paths that satisfy the QoS requirements based on multiple constraints such as the delay, the bandwidth or the jitter. In this paper, we propose two fast heuristics that quickly compute feasible paths if they exist. These heuristics are compared to the exact QoS routing algorithm: Self Adaptive Multiple Constraints Routing Algorithm (SAMCRA). For that, two main axes are explored. In the first axis, we limited the execution time of our heuristics. The simulation results show that the length of the computed paths is very close to the optimal ones that are computed by SAMCRA. Moreover, these heuristics satisfy more than $80\%$ of the feasible requests. In the second axis, to enforce our hypothesis about the relevancy of the proposed heuristics, we force our algorithms to compute paths until a feasible path is found if such a path exists. The success rate becomes then $100\%$. Moreover, the qualities of found solutions as well as the combinatorial complexity of our heuristics are still attractive.**

***Index Terms*— Routing, multi-constrained, quality of Service, heuristic**

## I. Introduction

Quality of Service (QoS) routing known as multi-constrained routing, consists in computing paths that meet a set of requirements such as delay, bandwidth, and cost. Most of the emerging multimedia applications become more stringent with QoS and require more guarantees. Wang and Crowcroft have proved that the multi-constrained routing problem is NP-hard [1]. For solving exactly or approximatively this problem, many solutions are proposed in the literature. Among the exact solutions, we mention the Depth First Search (DFS) approach which returns a feasible solution to the problem if such a solution exists [2]. Since the worst-case time complexity approach is exponential, Shane et al. proposed in 2001 a heuristic based on the DFS approach [3]. Another approach uses the Constrained Bellman-Ford algorithm for the delay-cost-constrained routing problem as done in [4]. The main idea of this algorithm is at first find minimal cost paths between the source and the other nodes. Then, the algorithm maintains a list of paths that increase the cost and decrease the delay. When the path with the smallest cost and acceptable delay is found the algorithm returns this path. However, in [5], it has been shown that a non-linear length is necessary to solve the QoS routing problem, and the authors replace the cost function by a non-linear length, and proposed the Self Adaptive Multiple Constraints Routing Algorithm SAMCRA. SAMCRA is an exact algorithm based on two main concepts: the use of a non-linear length function and the non dominance of paths. SAMCRA explores like Dijkstra's algorithm all nodes beginning by the source node and maintains a set of non-dominated feasible paths at each node. The algorithm stops when it computes the non dominated feasible path with the smallest non-linear length between the source and the destination nodes.

Since the optimal QoS routing with multiple constraints is NP-hard, heuristics are required for real network applications. A heuristic version of SAMCRA is given by TAMCRA [6], which was proposed earlier as a heuristic to solve the unicast QoS routing. Unlike SAMCRA, TAMCRA bounds the number of non dominated paths that can be stored at each node by a predefined integer $k$. Therefore, the found solution may not be the optimal one. In 2001, Yuan and Liu proposed an extended version of the Bellman-Ford algorithm that finds all optimal paths then chooses a feasible one if such a path exists [7]. In [8], Jaffe defined an approximation algorithm, which computes shortest paths based on a linear combination of the weight values of each link in one new weight. This algorithm was illustrated in the case of two metrics, a generalization for multiple metrics was proposed in [9]. H_MCOP is one of the well-known multi-constrained unicast algorithms that was introduced in [10]. This algorithm is based on the execution of two modified versions of Dijkstra's algorithm in forward and backward directions to compute the shortest paths between two nodes. Other works aim to solve the multi-constrained routing problem using Lagrange relaxation to mix the metrics. In this category, we can cite the algorithms proposed by Feng et al. in 2001 [11], 2002 [12], by Juttner et al. in 2001 [13] and by Guo and Matta in 1999 [14].

Recently, the research community is exploring the use of the metaheuristics to solve the multi-constrained routing problem, such as genetic algorithms [15], tabu search [7], and ant colonies [16].

Instead its effectiveness, the major drawback of SAMCRA resides in its complexity [17]. In this paper, we investigate the possibility to replace SAMCRA by two


doi:10.4304/jnw.7.1.73-80

simple heuristics that efficiently reduce the execution time and return satisfying solutions. These heuristics are based on the computation of the $k$ shortest paths. For this, we adapt the Yen's algorithm [18], which has the smallest combinatorial complexity [19]. The difference between the two heuristics lies in the metric used for shortest paths computation. Hop Count Approach (HCA) considers the hop count metric of a path, while the second heuristic Metric Linearization Approach (MLA) combines the QoS metrics into one weighted metric.

In the following, we first give a formal definition of the multi-constrained routing problem. In Section III, we outline SAMCRA algorithm. In Section IV, we give an overview of the proposed algorithms for computing the $k$ shortest paths. Our heuristics are presented in Section V. In Section VI, the performance of our heuristics is investigated through a large number of simulations.

## II. Problem Formulation

A communication network is modeled as an undirected weighted graph $G(N, E)$, where $N$ is the set of nodes and $E$ the set of links. Each link $e \in E$ of the network is associated with $m$ QoS parameters denoted by a weight vector: $\overrightarrow{w}(e) = [w_1(e), w_2(e), ..., w_m(e)]^T$. The QoS metrics can be classified into additive metrics such as delay, multiplicative metrics such as loss rate or bottleneck metrics such as available bandwidth. The end-to-end constraints of a given QoS request, from a source node $s$ to a destination node $d$, are given by an $m$-dimensional vector: $\overrightarrow{L} = [L_1, ..., L_m]^T$. In general, bottleneck metrics can be easily dealt with by pruning from the graph all the links that do not satisfy the QoS constraints, while the multiplicative metrics can be transformed into additive metrics by using a logarithm function. The additive metrics cause more difficulties. Therefore, and without loss of generality, we only consider additive metrics. The length of a path $p(s,d)$ corresponding to the metric $i$ is given by: $l_i(p(s,d)) = \sum_{e \in p(s,d)} w_i(e)$. Thus, we define a feasible path $p(s,d)$ as follows:

$$l_i(p(s,d_j)) \leq L_i, \ \forall i = 1, ..., m \tag{1}$$

Using the Pareto dominance, a path $p(s,d)$ dominates another path $p'(s,d)$ if:

$$\begin{cases} l_i(p(s,d)) \leq l_i(p'(s,d)), \ \forall i = 1, ..., m \\ l_j(p(s,d)) < l_j(p'(s,d)), \text{for at least one } j \end{cases} \tag{2}$$

In [5], the authors formulated the multi-constrained routing problem in two different ways.

**MCP problem** The Multi-Constraint Path (MCP) problem consists in finding a path $p(s,d)$ that satisfies a given constraint vector $\overrightarrow{L}$:

$$l_i(p(s,d)) \leq L_i, \ i \in \{1, ..., m\} \tag{3}$$

**MCOP problem** Considering a length function $l$ (e.g. $l(p(s,d)) = \frac{1}{m}\sum_1^m l_i(p(s,d))$), the Multi-Constraint Optimal Path (MCOP) problem consists in finding among the feasible paths, the path $p^*(s,d)$ with the smallest length $l(p^*(s,d))$.

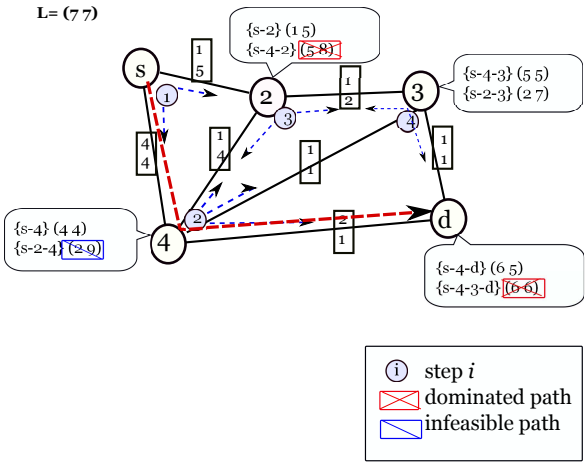



Figure 1. An example of SAMCRA

To evaluate the quality of a path $p(s,d)$, an interesting non-linear length function was defined in [5]:

$$l(p(s,d)) = max_{i=1,...,m}\left(\frac{\sum w_i(e)_{e \in p(s,d)}}{L_i}\right) \tag{4}$$

This length function considers the value of the most critical constraint of a path regarding the end-to-end requirements.

## III. SAMCRA

SAMCRA [5] is an exact multi-constrained routing algorithm that solves the MCOP problem using the non-linear length function and the dominance of paths.

For a node pair $(s,d)$, SAMCRA returns the shortest path that satisfies the constraint vector if such a path exists. Thus, SAMCRA begins by the source node $s$. At each iteration, the algorithm explores the neighbors of the current node and chooses the closest node using the non-linear length function defined in Equation 4. The dominated paths are regularly dropped, while all non dominated ones are memorized. SAMCRA ends when the destinations $d$ is reached, and there is no possibility to find a better path for this destination.

For instance, let us consider the example in Figure 1, where SAMCRA computes the path with the smallest non-linear length between $s$ and $d$. SAMCRA begins by exploring the neighbors of $s$ $\{2, 4\}$ , and chooses the node $4$ with the smallest non-linear length. Then, it explores its neighbors $\{2, 3, d\}$. The algorithm stops at the $4^{th}$ iteration, when the node $d$ is selected to having the smallest non-linear length. In the same figure, we notice that dominated paths like $(s-4-3-d)$ are dropped as well as the infeasible one $(s-2-4)$.

## IV. The $k$ shortest paths algorithms

Routing is usually associated with the computation of shortest paths (in number of hops for example). However,

when the shortest path does not satisfy the constraints of QoS, it becomes necessary to compute a set of $k$ shortest paths between a source node and destination node to find a feasible path. The $k$ shortest paths problem is a natural and long-studied generalization of the shortest path problem in which not one but several paths in an increasing order of length are required. The $k$ shortest paths problem in which paths can contain loops turns out to be significantly easier. An algorithm with a complexity in $O(|E| + k.|N|.log|N|)$ has been known since 1975 [20]; a recent improvement by Eppstein achieves the optimal complexity in: $O(|E| + |N|.log|N| + k)$ [21]. However, the problem of determining the $k$ shortest paths without loops has proved to be more challenging. The problem was first examined by Hoffman and Pavley [22]. For undirected graphs, the most efficient algorithm was proposed by Katoh et al. [23], which has a complexity in $O(k.(|E|+|N|.log|N|))$. Since undirected graphs can be transformed on directed graphs, by replacing the undirected link with two directed links with the same weights, on the most general case, the best known algorithm is that proposed by Yen in [18]. The Yen's algorithm was generalized by Lawler in [24] and has a complexity in $O(k.|N|(|E| + |N|.log|N|))$.

## V. Proposed Heuristics

### A. Motivation

It has been proved that the multi-constrained routing problem is NP-hard [1]. SAMCRA is an efficient algorithm that exactly solves this problem. Although its effectiveness, SAMCRA can be expensive with a combinatorial complexity in: $O(k|N|\log(k|N|)+k^2m|E|)$ [17], with $k$ the number of non dominated paths that can be stored at each node queue. Therefore, it may be more interesting to use an approximate algorithm with less combinatorial complexity expecting a feasible path between the source and the destination nodes. Reducing the execution time while giving satisfying solutions is our main motivation to propose our fast heuristics. These heuristics are based on the computation of the $k$ shortest paths. The idea of applying such an algorithm is that the shortest paths may be feasible. For that, we propose the modification of the well-known algorithm of Yen [18] to compute the paths between two nodes in increasing order of their length until (i) a feasible path that satisfies all constraints is found, (ii) or a given limitation of computed paths is reached.

To solve the multi-constrained unicast routing, we propose two heuristics based on the computation of shortest paths. Indeed, we argue that one of these computed paths will be feasible. Moreover, these heuristics have a combinatorial complexity that is bounded by the number of allowed computed shortest paths. The proposed heuristics are using one additive metric. The first heuristic computes the paths with the smallest number of links. The second heuristic uses a combination of the QoS metrics in a single one, using an efficient technique that will be explained in detail in Section V-C.

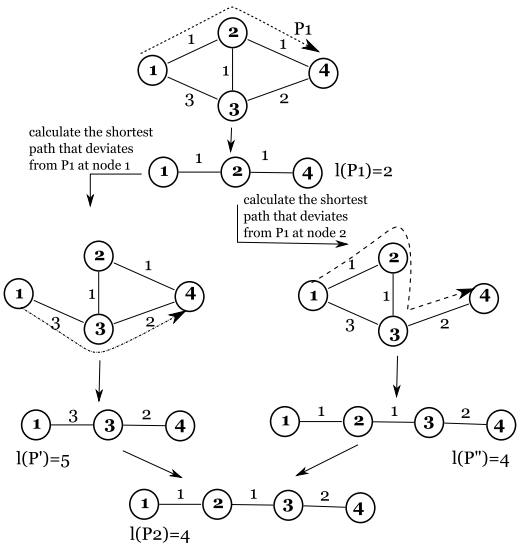



Figure 2. An example of Yen's algorithm processing

Both the proposed heuristics are based on Yen's algorithm, which we outline in the following.

### B. Yen's Algorithm

As shown in many studies [19], Yen's algorithm is the most pertinent and fast $k$ shortest paths algorithm that was introduced in [18].

For a given node pair $(s, d)$ and a given integer $k$, this iterative algorithm computes, using one additive metric, the $k$ shortest paths between these nodes. For that, it begins by computing the first shortest path using the Dijkstra algorithm. At the $i^{th}$ iteration, the algorithm computes the $i^{th}$ shortest path by considering all possible paths that deviate from the $(i-1)^{th}$ shortest path that are not already computed.

For instance, let us consider the example in Figure 2, where the first two shortest paths between the nodes 1 and 4 are required. The algorithm begins by computing the shortest path $P_1$. Then, it computes all shortest paths that deviate from $P_1$ at nodes 1 and 2. Two paths are computed $P^{'}$, $P^{''}$. The shortest one, here $P''$ with length 4, is the second shortest path $P_2$.

### C. Algorithmic Description of the Proposed Approaches

In this paper, we propose two fast heuristics that compute shortest paths in an increasing order, given an additive length function. These heuristics stop when (i) a path satisfying all the QoS constraints is found or (ii) an upper bound of the number of computed paths $k_{max}$ is reached. $k_{max}$ is an important parameter, since the combinatorial complexity and so the execution time of our heuristics depend on its value. Indeed, when $k_{max}$ is small, these heuristics are fast, and the number of satisfied request can be small. In parallel, when $k_{max}$ increases, the number of satisfied requests increases too. Furthermore, the proposed heuristics use a single additive metric obtained by two ways:

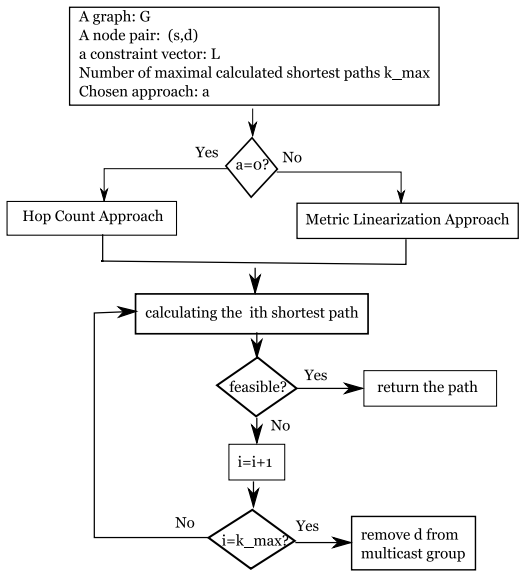



Figure 3. Proposed heuristics diagram

**Hop Count Approach (HCA):** in this approach, the algorithm searches for the shortest paths considering the number of hops as the only metric to optimize.

**Metric Linearization Approach (MLA)**: at first, it substitutes the weight vector $\overrightarrow{w}\ (e) = [w_1(e), .., w_m(e)]^T$, by a scalar weight $w^{'}(e) = \sum_{i=1,...,m} \alpha_i w_i(e)$. To calculate the parameters $\alpha_i$, the MLA approach computes $p_i^*(s, d_j),\ i = 1, ..., m$, the shortest path that optimizes the metric $i$. Then, the parameter $\alpha_i$ is calculated as follows:

$$\alpha_i = \frac{l_i(p_i^*(s, d_j))}{L_i} \tag{5}$$

$\alpha_i$ can be named *the criticality degree* of the constraint $L_i$. When $\alpha_i$ is close to 1, that means $p_i^*(s, d_j)$ is very close to $L_i$. Consequently, it is necessary at first to satisfy the constraint $L_i$.

Figure 3 summarizes the process of our heuristics HCA and MLA. For a given graph $G$, a given node pair $(s, d)$, with a constraint vector $\overrightarrow{L}$ and a chosen approach, the algorithm returns the fist feasible path between $s$ and $d$, if such a path is one of the $k_{max}$ shortest paths that are allowed to be computed. For that, the algorithm chooses the approach to use and compute the combined link weights in case of MLA. Then, it computes shortest paths in an increasing order. The process stops when a feasible path is found, or $k_{max}$ iterations are already done. A more detailed meta-code is presented in Algorithm 1 for MLA approach.

**Algorithm 1** MLA meta-code

**for** $(i = 1, ..., m)$ **do**
  Compute $p_i^*(s, d)$
  $\alpha_i = \frac{l_i(p_i^*(s,d))}{L_i}$
**end for**
**for all** $e \in G$ **do**
  $w^{'}{}_i(e) = \sum_{i=1,...,m} \alpha_i w_i(e)$
**end for**
$j \leftarrow 1$, Find←false, $k \leftarrow 1$,
$p_1(s, d) = Dijkstra(s, d), P, P'$//paths
$D = p_1(s, d)$ // candidate set
$X = \phi$ //shortest paths set
**while** ((Find≠true) and ($k \leq k_{max}$)) **do**
  $P \leftarrow$shortest path in $D$

  **if** ($p_j(s, d)$ is feasible) **then**
    Find←true
  **else**
    $j \leftarrow j + 1$
    $v \leftarrow$ deviation[1]$(p_j(s, d), X)$
    **while** ($v \neq d$) **do**
      $P \leftarrow Dijkstra(v, d)$
      $P' \leftarrow$ prefix[2]$(p_j(s, d), v) + P$
      $D \leftarrow D + P'$
      $v \leftarrow$successor$(v, p_j(s, d))$
    **end while**
  **end if**
**end while**

### *D. Limitations of the Yen's algorithm*

The proposed heuristics MLA and HCA are based on the computation of the $k$ shortest paths using the Yen's algorithm. However, Yen's algorithm computes the $k$ shortest paths without considering all the existing paths as we will demonstrate it in Figure 4.

On the left side (a), at first HCA computes the three shortest paths $P_1$,$P_2$ and $P_3$, that have the same hop count. When computing the fourth shortest path using the deviation concept at node 2, the algorithm will choose one of the two paths $P'$ and $P$". As the algorithm chooses one of the two latter paths, it can not choose the other one at the next iteration. This can lead to some cases where not all paths are explored and the feasible path can be skipped. For instance, if the constraints are given by the vector $\overrightarrow{L}\ (3, 3)$, only the path $P'$ will be feasible, and this path can be skipped by the original Yen's algorithm.

On the right side (b), we suppose that $\overrightarrow{L} = (4, 4)$. MLA starts by computing the shortest paths considering successively the two metrics. Here the path $(s - 2 - d)$ minimizes the first metric with the value 2, and the path $(s - 1 - d)$ minimizes the second metric with the value 2. Then, MLA computes the parameters $\alpha_1 = \frac{2}{4} = 0.5$ and $\alpha_2 = \frac{2}{4} = 0.5$. After adding the new weights to the links, MLA computes the first shortest path $P_1$. At the second iteration, when computing the shortest path at the deviation node $s$, MLA can chooses $P^*$ or $P^{**}$ since they have the same length 4. However, only $P^*$ is feasible.

This limitation prevents to compute feasible paths even if this path is one of the shortest paths. To cure this, we propose a modification of the Yen's algorithm. We replace the computation of the shortest path using Di-

[2]deviation is a function that returns the furthest node at which a path P deviates from a set of paths

[2]prefix retuns the sub-path of a path P, between the source node and a defined node v

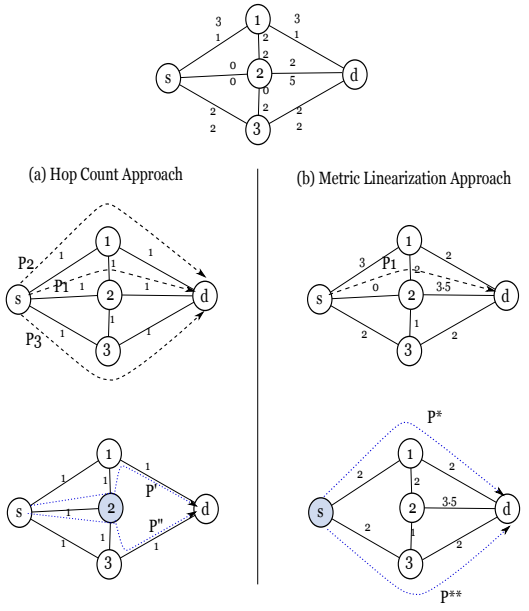



Figure 4. The relevance of enumerating all paths having the same length

jkstra's algorithm by a modified version denoted Mod-Dijkstra that computes all shortest paths that have the same length at the same time, if more than one exist. For this, instead of saving only one of the predecessors with the smallest length at each node, the algorithm saves all the predecessors that have the same smallest length.

### *E. Exact HCA and Exact MLA algorithms*

Exact HCA (E-HCA) and exact MLA (E-MLA) are two modified version of HCA and MLA respectively. These algorithms are based ob three main modifications.

- at each deviation node, the algorithm of computation of shortest paths computes all paths with the same additive length (hop count for E-HCA and metric linearization for E-MLA),
- the upper bound $k_{max}$ of computed paths become infinite ($k_{max} \rightarrow \infty$),
- the algorithm stops only when a solution is found for feasible requests.

Considering the example of Figure 4, two iterations will be sufficient for HCA to compute the feasible path $P$" without skipping this path. In the first iteration, Mod-Dijkstra returns the set $P_1, P_2, P_3$. Then, at the second iteration it returns both $P'$ and $P$". For MLA, the problem is less recurrent because of the new weights computation. Indeed, the paths have generally different lengths, and MLA cannot skip paths that have the same length.

## VI. Performance Evaluation and Simulations

The performance of the two proposed heuristics and SAMCRA are investigated through extensive simulations. For that, we use a realistic network with $50$ nodes and $82$ links denoted by Real-Topology [25]. Each link is associated with two additive weights. These weights are randomly generated using a uniform distribution in the interval $[1, 1024]$.

Different classes of constraints are also considered. For each pair of nodes $(s, d)$, the constraint vectors are generated in a way that they browse a defined space generation by areas from the strictest constraints to the loosest ones. In Figure 5, where only two metrics are considered, $P_1$ and $P_2$ denote the shortest paths between $s$ and $d$ that minimizes the first and second metric respectively. The shaded rectangle (B) delimited by $l_1(P_1),\ l_1(P_2)$ and $l_2(P_2),\ l_2(P_1)$ circumscribes the region where the constraints are selected. This region is divided in $10$ areas: area 1, area 2,..., area 10 (also denoted in the simulation figures by 1,2,...,10). The constraints are randomly selected within these areas. Outside the specified region, the QoS constraints are less interesting to be examined. Indeed, all constraints that are generated within space (A) are infeasible, while all constraints generated in space (C) are trivial and any polynomial algorithm will be sufficient to find solutions. We note that strict constraints are close to $l_1(P_1)$ and $l_2(P_2)$ (area 1), while loose constraints are close to $l_1(P_2)$ and $l_2(P_1)$ (area 10).

In the followings, two series of simulation are performed.

### *A. HCA and MLA performance evaluations*

In this part, several series of simulations have been performed. We randomly generate $100$ instances of link weights. For each instance of link weights, $100$ pair of nodes are randomly selected. Thereafter, $10$ routing requests are generated within each area, from the strictest constraints (area 1) to loosest ones (area 10). After that, the three algorithms: SAMCRA, HCA and MLA are executed independently to find a solution. Four performance measures are computed.

- *Success rate:* it is the number of satisfied routing requests from $100$ generated requests,
- *Quality of computed paths:*it corresponds to two lengths. The non-linear length and the average length of computed paths. The non-linear length is used by SAMCRA (Equation 4) and corresponds to the satisfactory degree of the most critical constraint. The average length ($l_{\text{avg}}(p(s,d)) = \frac{1}{m}\sum_1^m l_i(p(s,d))$) equivalently considers all the metrics, and computes the average quality of the computed paths.
- *Relative complexity:* it is the number of operations[3] that are performed to find a feasible solution,
- *Absolute complexity:* it is the number of operations that are performed before answering a given routing request, instead the request is infeasible,

We note that all these performance measures have been computed with $95\%$ confidence intervals according to the ten constraint generation areas. The upper bound $k_{max}$ of the computed shortest paths in our heuristics is fixed to three.

- **Success rate**

[3]an elementary operation corresponds to the visit of one node

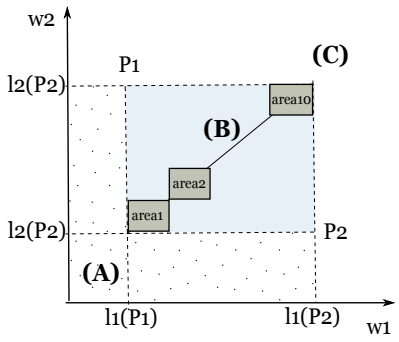



Figure 5. The constraints generation

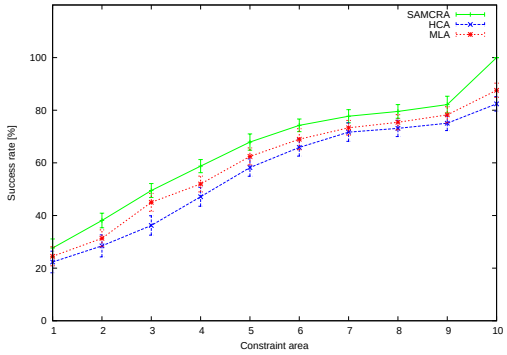



Figure 6. The success rate of the three algorithms SAMCRA,HCA and MLA

Figure 6 shows the success rate of the three algorithms: SAMCRA, HCA and MLA. Foremost, we notice that the success rate of the three algorithms is increasing. In fact, for strict constraints there is few feasible requests, and this number is increasing when constraints become loose. Since SAMCRA is an exact algorithm, it gives the highest success rate, while the upper bound of computed paths in our heuristics is fixed to three. The success rate of SAMCRA varies from $29\%$ for strict constraints to $100\%$ for the loose ones. For strict constraints, the gap between SAMCRA and our heuristics does not exceed $4\%$ with MLA and $7\%$ with HCA. For loose constraints (area $10$), the difference becomes $12\%$ with MLA and $16\%$ with HCA. Indeed, the area $10$ presents the trivial constraints for which SAMCRA always finds a solution, and the number of feasible requests is $100\%$.

The execution time is an important parameter when evaluating any routing algorithm. To evaluate more deeply the three algorithms, two kinds of complexity metrics are proposed. The relative complexity for feasible requests and the absolute complexity for the total generated requests.

- **Relative complexity**

Relative complexity is calculated only if a solution is found by the three algorithms. In Figure 7, we notice that the relative complexity of SAMCRA is significantly larger than both of HCA and MLA. Indeed, when the constraints are not strict, SAMCRA has more paths to explore before returning the optimal solution according to the non-linear length, while both approaches HCA and MLA stop at the first feasible path they find. This reduces their execution time.

- **Absolute complexity**

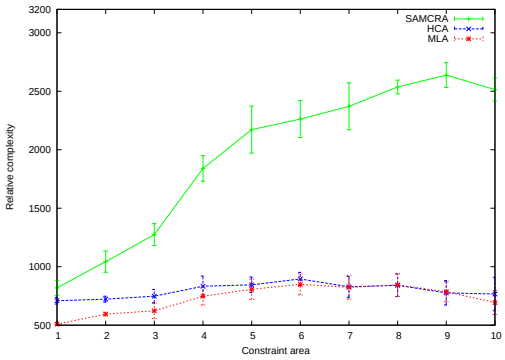



Figure 7. Relative complexity of the algorithms SAMCRA, HCA and MLA

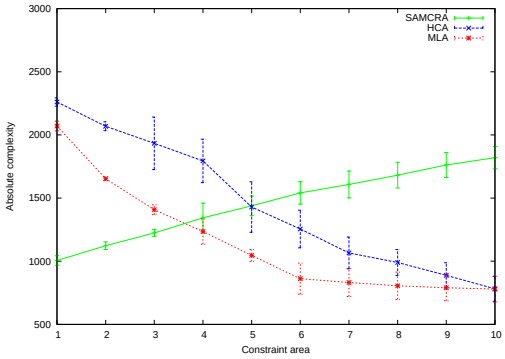



Figure 8. The absolute complexity of the algorithms SAMCRA, HCA and MLA

In Figure 8, we can state that the absolute complexity of both proposed heuristics HCA and MLA is greater than SAMCRA complexity for strict constraints. Indeed, when the constraints are strict, SAMCRA rapidly rejects non feasible requests, while both approaches explore the maximum number of paths (here fixed to three) without finding solutions. When constraints become less strict; the two proposed approaches find feasible solutions before computing the three paths, which significantly reduces their execution time.

- **Quality of computed paths**

In Figure 9, the solutions found by our approaches HCA and MLA are a little bit worse than those found by SAMCRA with $6.67\%$ and $2.39\%$ respectively. Obviously, SAMCRA finds the path with the smallest non-linear length, while both heuristics HCA and MLA stops at the first feasible path, which can be worse than the optimal one. Moreover, for strict constraints, the solutions computed by the three algorithms are significantly equal. In fact, for strict constraints, the number of feasible paths is very small, and if HCA or MLA computes a feasible path, this path has big chances to be the optimal one.

In Figure 10, the average length of MLA is better than that of SAMCRA with $1.59\%$, and the average length of HCA is very close to that of SAMCRA. Indeed, the linearization of the metrics involves the computation of paths based on both metrics, unlike SAMCRA which does not consider the variation between the metrics and

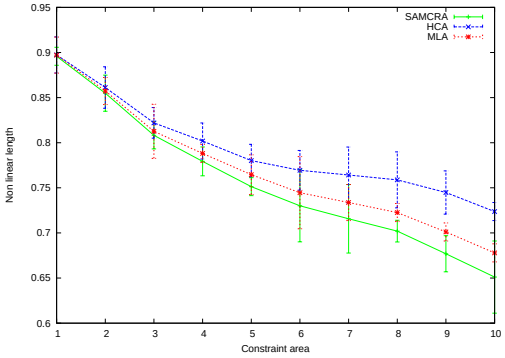



Figure 9. The non-linear length of the paths computed by SAMCRA, HCA and MLA

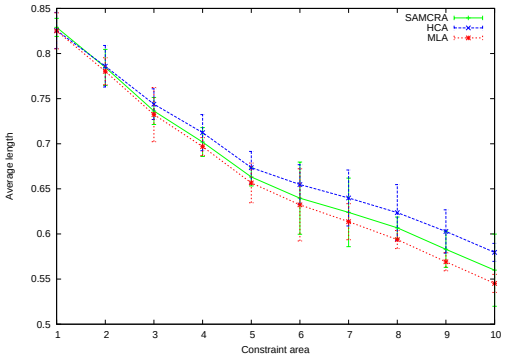



Figure 10. The average length of the paths computed by SAMCRA, HCA and MLA

focuses on the most critical one. The following example highlights the difference between non-linear length and average length evaluation. For that, let us consider two paths: the path computed by SAMCRA $p_1(s,d)$ : with the lengths $l_1(p_1(s,d)) = 0.6$, $l_2(p_1(s,d)) = 0.7$ corresponding to the metrics 1 and 2 respectively, and the path computed by HCA $p_2(s,d)$ : with the lengths $l_1(p_2(s,d)) = 0.1$, $l_2(p_2(s,d)) = 0.9$. It is clear that the non-linear length $l(p_1(s,d)) = 0.7 < l(p_2(s,d)) = 0.9$, while the average length $l_{avg}(p_1(s,d)) = 0.65 > l_{avg}(s,d) = 0.5$.

### *B. Exact-HCA and Exact-MLA Performance Evaluation*

In the second series of our study the exact version of the algorithms E-HCA and E-MLA has been analyzed. In fact, after the modification of the Yen's algorithm in order not to skip paths with the same length, E-HCA and E-MLA will explore all existing paths ($k_{max} \rightarrow \infty$) if necessary, between the source and the destination nodes. For this, they compute the paths in an increasing order according to the previously explained metrics. E-HCA uses the hop count metric, while E-MLA uses the linearization metric. In this series of simulations, the requests as well as the constraints are generated similarly to the first series of simulations. Two measure parameters are evaluated:

- *Number of computed paths:* is the average number of computed paths to find a feasible solution for feasible requests,
- *Exact complexity :* is the number of operations that are performed to find a solution for all feasible requests.

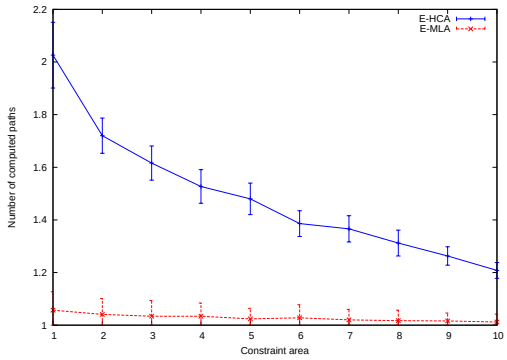



Figure 11. Number of computed paths before finding a solution by E-HCA and E-MLA

- **Number of computed paths**

Figure 11 shows the number of computed paths by E-HCA and E-MLA before finding a feasible path. For strict constraints, E-HCA needs 2.1 paths to find a solution while E-MLA does not need more than 1.13 paths. The number of computed paths is decreasing when the constraints are less strict. Instead numbers of computed paths by E-HCA and E-MLA are lower than the upper bound $k_{max}$ of HCA and MLA, E-MLA as well as E-MLA computes paths until a feasible one is found. In the $10\%$ of cases where HCA and MLA do not find feasible paths they need to compute more than 3 paths. In the other cases, the computed paths is almost the first or the second one.

A simple demonstration can justify these small values. Let us suppose that HCA needs $1.5$ paths to find solution for $23\%$ of the generated requests as shown in Figure 6. In addition, E-HCA needs to compute $5$ paths for the $5\%$ to reach the success rate of $28\%$ as SAMCRA. Since $23\%$ presents $82\%$ of $28\%$, the average number of computed paths $N_p$ will be computed as follows: $N_p = 0.82 * 1.5 + 0.18 * 5 = 2.13$. A similar demonstration can be done for MLA.

- **Exact complexity**

In Figure 12, the exact complexity of E-HCA is bigger than SAMCRA for strict constraints. This is due to the modified version of Yen's algorithm and the computation of additional paths. This high complexity can also be explained by the increasing number of explored paths and the small number of feasible ones. Let us suppose that the only feasible path is the second shortest one and there are $10$ first shortest paths, and $10$ second shortest paths. In this case, at least $10$ paths and at most $19$ paths will be explored before finding the feasible one. Since the number of computed paths in E-MLA is less than $1.13$ paths, its exact complexity still the smallest one for both strict and loose constraints. Indeed, E-MLA computes paths using the linearization metric that simultaneously takes into account the two considered metrics.

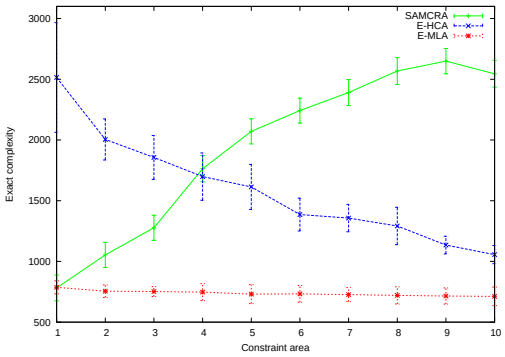



Figure 12. The exact complexity of SAMCRA, E-HCA and E-MLA

The exact complexity of E-MLA is very interesting since it is still smaller than SAMCRA, while having 100% of success rate. However, the compromise done is regarding the quality of found solutions by E-MLA which may not be optimal but still feasible and this can be sufficient for most requests.

## VII. Conclusion

To solve the QoS routing problem, we proposed two fast heuristics HCA and MLA. These heuristics are simple to implement and give attractive results regarding the success rate, the quality of found solutions and they considerably reduce the execution time. The execution time is one of the most challenging parameters in the treated NP-hard problem. For this, extensive simulations are performed to compare our heuristics to the well-known algorithm SAMCRA. Two main ideas are developed then confirmed in this paper. The first idea is that HCA as well as MLA have a bounded combinatorial complexity that can be readjusted to reach a minimum success rate. The second idea focuses on the combinatorial complexity of HCA and MLA if they are transformed into exact algorithms, and the obtained results are very satisfying. Finally, since the current network applications become more exigent, it is interesting to explore the heuristic solutions to solve the NP-hard QoS routing problem.

## References

[1] Z. Wang and J. Crowcroft, "Quality-of-service routing for supporting multimedia applications," *IEEE Journal on Selected Areas in Communications*, vol. 14, pp. 1228–1234, 1996.

[2] R. E. Tarjan, "Depth-First Search and Linear Graph Algorithms," *SIAM Journal of Computers*, vol. 1, no. 2, pp. 146–160, 1972.

[3] D. W. Shin, E. K. P. Chong, and H. J. Siegel, "A multi-constraint QoS routing scheme using the depth-first search method with limited crankbacks," in *IEEE workshop on high performance switching and routing*, 2001, pp. 385–383.

[4] S. Chen and K. Nahrstedt, "On finding multi-constrained paths," in *ICC*, 1998.

[5] P. Van Mieghem, H. De Neve, and F. A. Kuipers, "Hop-by-hop quality of service routing," *Computer Networks*, vol. 37, no. 3/4, pp. 407–423, 2001.

[6] H. D. Neve and P. V. Mieghem, "Tamcra: a tunable accuracy multiple constraints routing algorithm," *Computer Communications*, vol. 23, no. 7, pp. 667–679, 2000.

[7] Y. Xin, "Heuristic algorithms for multiconstrained quality-of-service routing," *IEEE/ACM Transaction in Networks*, vol. 10, no. 2, pp. 244–256, 2002.

[8] J. M. Jaffe, "Algorithms for finding paths with multiple constraints," *Networks*, vol. 14, pp. 95–116, 1984.

[9] H. A. L. Lachlan and A. Kusumat, "Generalised analysis of a qos-aware routing algorithm," in *in IEEE GLOBECOM'98*, 1998, pp. 118–123.

[10] T. Korkmaz and M. Krunz, "Multi-constrained optimal path selection," vol. 2, pp. 834–843, 2001.

[11] G. Feng, K. Makki, N. Pissinou, and C. Douligeris, "An efficient approximation algotihm for delay-cost-constrained QoS routing," in *Tenth International conference on computer communications and networks*, 2001, pp. 395–400.

[12] G. Feng, C. Douligeris, K. Makki, and N. Pissinou, "Performance evaluation of delay-constrained least-cost QoS routing algorithms based on linear Lagrange relaxation," in *IEEE international conference on communications*, vol. 4, 2002, pp. 2273–2281.

[13] A. Jttner, B. Szviatovszki, I. Mcs, and Z. Rajk, "Lagrange relaxation based method for the qos routing problem," 2001.

[14] L. Guo and I. Matta, "Search space reduction in QoS routing," *Comput. Networks*, vol. 41, no. 1, pp. 73–88, 2003.

[15] Y. H. S. Wan and Y. Yang, "Approach for multiple constraints based qos routing problem of network," *Hybrid Intelligent Systems, International Conference on*, vol. 2, pp. 66–69, 2009.

[16] W. L.-j. S. Li-juan and W. Ru-chuan, "Ant colony algorithm for solving qos routing problem," *Wuhan University Journal of Natural Sciences*, vol. 7, pp. 449–453, 2004.

[17] P. Van Mieghem and F. A. Kuipers, "On the complexity of QoS routing," *Computer Communications*, vol. 26, no. 4, pp. 376–387, 2003.

[18] J. Y. Yen, "Finding the k shortest loopless paths in a network," *Management Science*, vol. 17, pp. 712–716, 1971.

[19] M. M. J. Hershberger and S. Suri, "Finding the k shortest simple paths: A new algorithm and its implementation," *ACM Transactions on Algorithms*, vol. 3, 2007.

[20] B. L. Fox, "More on kth shortest paths," *Commun. ACM*, vol. 18, no. 5, p. 279, 1975.

[21] D. Eppstein, "Finding the k shortest paths," in *FOCS*, 1994, pp. 154–165.

[22] W. Hoffman and R. Pavley, "A method for the solution of the nth best path problem," *J. ACM*, vol. 6, no. 4, pp. 506–514, 1959.

[23] N. Katoh, T. Ibaraki, and H. Mine, "An efficient algorithm for $K$ shortest simple paths," *Networks*, vol. 12, no. 4, pp. 411–427, 1982.

[24] E. L. Lawler, "A procedure for computing the $K$ best solutions to discrete optimization problems and its application to the shortest path problem," *Management Science*, vol. 18, pp. 401–405, 1972.

[25] A. Zimolka, "Design of survivable optical networks by mathematical optimization," Ph.D. dissertation, Technische University Berlin, 2007.

# Bounded Length Least-Cost Path Estimation in Wireless Sensor Networks Using Petri Nets

Lingxi Li and Dongsoo Stephen Kim
Department of Electrical and Computer Engineering, Indiana University-Purdue University Indianapolis, Indiana, USA
Email: {LL7, dskim}@iupui.edu

*Abstract*— **This paper proposes an iterative algorithm for estimating the least-cost paths in a wireless sensor network (WSN) that is modeled by a Petri net. We assume that each link between two sensor nodes (each transition in the net) is associated with a positive cost to capture its likelihood (e.g., in terms of the transmission time, distance between nodes, or the link capacity). Given the structure of the Petri net and its initial/final state, we aim at finding the paths (with length less than or equal to a given bound $L$) that could lead us from the initial state to the final state while having the least total cost. We develop an iterative algorithm that obtains the least-cost paths with complexity that is polynomial in the length $L$.**

*Index Terms*— **Wireless sensor networks, Petri nets, least-cost, path estimation**

## I. Introduction

In recent years, major advances in wireless communication and digital electronics have led to the development of sensors that can broadcast data over short communication ranges. A wireless sensor network (WSN) is composed of a number of sensor nodes that have sensing, processing, and communication capabilities that enable all sensor nodes to collaborate in order to achieve a particular task [1]. In a WSN, sensor nodes are densely deployed, have limited computational and power capacities, and use basic communication protocols to exchange information with their neighbors. In practice, WSNs have been widely employed in a variety of applications including transportation systems, military systems, and health-care systems [2]– [6].

As the size and complexity of WSNs increase, significant attention is paid to problems of modeling and analysis of WSNs. Since information transmission between sensor nodes can be treated as the occurrences of discrete events after some levels of abstraction, several discrete event dynamic system (DEDS) approaches have been proposed including finite state machines [7]– [9], Markov chains [10], [11], and Petri nets [12]– [17]. In literature, several aspects of problems related to WSNs (e.g., sensor data processing, energy consumption, and intelligent monitoring) have been studied using Petri net models due to their powerful modeling and mathematical analysis framework. In particular, the authors of [12] proposed a fuzzy Petri net model to process sensor data and handle flexible and continuous queries. In [13], the authors developed an attack-resistant and efficient localization scheme in WSNs based on Petri nets. The power consumption of processors in WSNs was studied using Petri nets in [14] and the energy consumption of wireless sensor nodes based on Petri net models was analyzed in [15]. A decentralized Petri net based wireless sensor node architecture (PN-WSNA) was developed to construct a flexible and reconfigurable WSN for intelligent monitoring systems in [16] and [17].

Other than the works mentioned above, in this paper we study a different problem in WSNs based on Petri net models, namely, the least-cost path estimation problem. In literature, path estimation and path planning in WSNs have been investigated using a variety of approaches. In [18], the authors considered the dynamic shortest path routing problem in mobile ad hoc networks. An genetic algorithm was developed to tackle this problem and was demonstrated its effectiveness through simulations. The authors of [19] studied the path planning problem in WSNs to deal with transmission delay using queueing networks. They proposed an approximate iterative algorithm to calculate node packet arrival rate and designed the pre-selection algorithm based on a path search tree. The optimal paths can then be obtained based on the analysis of the path delay. In [20], the authors proposed an shortest path routing protocol for mobile WSNs. This protocol was based on neighbor discovery and shortest path construction. The authors of [21] developed an optimal multipath planning algorithm for intrusion detection in wireless sensor and actor networks. They associated each link with a cost that is related to packet transmission and employed graphical algorithms to determine the optimal paths. In [22], the authors considered the cluster routing problem in WSNs and proposed a fuzzy reasoning algorithm based on fuzzy Petri nets to improve the routing reliability and enhance the energy efficiency.

In this paper we study the least-cost path estimation problem in WSNs based on Petri net models. In particular, we consider a setting where we have knowledge of the structure of a Petri net and its initial state, and we are

This paper is based on "Least-cost path estimation in wireless ad hoc sensor networks using Petri nets," by L. Li and D. S. Kim, which appeared in the Proceedings of the 5th International Conference on Ubiquitous Information Management and Communication (ICUIMC), Seoul, Korea, February 2011. © 2011 ACM.

Manuscript received May 28, 2011; revised July 10, 2011; accepted August 31, 2011.

This work was supported in part by the Indiana University-Purdue University Indianapolis (IUPUI) Research Support Funds Grant. Corresponding author: Lingxi Li, 723 W. Michigan St., SL 160, Indianapolis, IN 46202, USA. Email: LL7@iupui.edu.


doi:10.4304/jnw.7.1.81-87

given a final state (i.e., a target state to be reached) together with a positive integer number $L$ which serves as the maximum allowable length for sequences of transitions whose firings can lead us from the initial state to the final state. We assume that each link in the WSN (each transition in the net) is associated with a positive cost that captures its likelihood (e.g., in terms of the transmission time, distance between nodes, or the link capacity). Given the above constraints (i.e., a Petri net structure, the initial/final state, and a length $L$), we aim at finding the paths in the WSN (transition firing sequences in the Petri net) with length less than or equal to $L$ which: (i) are consistent with the structure of the Petri net, (ii) lead us from the initial state to the final state, and (iii) have the least total cost (the total cost of a transition firing sequence is taken to be the sum of the costs of all transitions in the sequence). We develop an iterative algorithm that obtains the least-cost transition firing sequences with complexity that is polynomial in $L$.

This work is motivated by the problem of least-cost path estimation in WSNs that are modeled by Petri nets. For example, in a WSN, information transmission involves the determination of detailed sequences of paths from an initial state to a final state. In our setup, the given final state captures the completion of the information transmission, whereas the structure of the given Petri net represents the interactions among different sensor nodes (as imposed by the given WSN). We also associate each transition in the given net with a positive cost which could represent its likelihood of occurrence. Note that in this paper, we focus on finding the least-cost transition firing sequences of length upper bounded by the given value $L$, which follows from the reasonable assumption of completing the information transmission in a WSN within a bounded number of steps due to energy constraints of sensor nodes.

In literature, estimation using Petri net models has been studied extensively. For instance, in [23], [24] the authors present an algorithm for obtaining an estimate (and a corresponding error bound) for the state of a given Petri net based on full knowledge of the observed firing sequence but without knowledge of the initial state; these works also discuss how this state estimate may be used to design a controller. In this paper we consider path estimation other than state estimation and our approach is different from those mentioned above. The authors of [25] studied the least-cost firing sequence estimation problem in labeled Petri nets with unobservable transitions and proposed an estimation algorithm with complexity that is polynomial in the length of the observed sequence of labels. In this paper we consider ordinary Petri net models rather than labeled Petri net models.

Note that due to the structure of the given net, the particular final state might be reached from the initial state through paths of different lengths. Therefore, in general, we need to consider all states reachable from the initial state in $L$ or less steps in order to capture the actual least-cost paths that could lead us from the initial state to the final state. Crucial to our algorithmic complexity analysis is the fact that the number of states that are consistent with a path of a certain length is upper bounded by a function that is polynomial in this length (this follows from a translation of the result in [26]). Using this observation, we are able to show that our algorithm can obtain the least-cost paths with complexity that is polynomial in length $L$.

## II. Petri Net Notation

In this section, we provide basic definitions and terminology that will be used throughout the paper. More details about Petri nets can be found in [27], [28].

A Petri net structure is a weighted bipartite graph $N = (P,T,A,W)$ where $P = \{p_1, p_2, \ldots, p_n\}$ is a finite set of *places* (drawn as circles), $T = \{t_1, t_2, \ldots, t_m\}$ is a finite set of *transitions* (drawn as bars), $A \subseteq (P \times T) \cup (T \times P)$ is a set of arcs (from places to transitions and from transitions to places), and $W : A \to \{1,2,3,\ldots\}$ is the *weight function* on the arcs.

Let $b_{ij}^-$ denote the integer weight of the arc from place $p_i$ to transition $t_j$, and $b_{ij}^+$ denote the integer weight of the arc from transition $t_j$ to place $p_i$ ($1 \le i \le n$, $1 \le j \le m$). Note that $b_{ij}^-$ ($b_{ij}^+$) is taken to be zero if there is no arc from place $p_i$ to transition $t_j$ (or vice versa). We define the *input incident matrix* $B^- = [b_{ij}^-]$ (respectively the *output incident matrix* $B^+ = [b_{ij}^+]$) to be the $n \times m$ matrix with $b_{ij}^-$ (respectively $b_{ij}^+$) at its $i^{th}$ row, $j^{th}$ column position. The *incident matrix* of the Petri net is defined to be $B \equiv B^+ - B^-$.

A *marking* (state) is a vector $M : P \to Z_0^+$ that assigns to each place in the Petri net a nonnegative integer number of tokens (drawn as black dots). We use $M_0$ to denote the initial marking of the Petri net. A transition $t$ is said to be *enabled* if each of its input places $p_{in}$ has at least $B^-(p_{in},t)$ tokens. We use $M[t\rangle$ to denote that transition $t$ is enabled at marking $M$. An enabled transition $t$ may *fire* and if it fires, it removes $B^-(p_{in},t)$ tokens from each input place $p_{in}$ of $t$ and deposits $B^+(p_{out},t)$ tokens to each output place $p_{out}$ of $t$ to yield a new marking $M' = M + B(\cdot,t)$, where $B(\cdot,t)$ denotes the column of $B$ that corresponds to transition $t$. This is also denoted by $M[t\rangle M'$ and we say marking $M'$ is *reachable* from marking $M$ via the firing of transition $t$.

Let $\sigma = t_{i1}t_{i2}\ldots t_{ik}$ ($t_{ij} \in T$) be a transition firing sequence. We say $\sigma$ is enabled with respect to $M$ if $M[t_{i1}\rangle M_1[t_{i2}\rangle \ldots M_{k-1}[t_{ik}\rangle$; this is denoted by $M[\sigma\rangle$. Let $M[\sigma\rangle M'$ denote that $M'$ is reachable via the firing of transition sequence $\sigma$ from $M$, and let $\overline{\sigma}(t)$ be the total number of occurrences of transition $t$ in $\sigma$. More specifically, $\overline{\sigma} = [\overline{\sigma}(t_1)\ldots\overline{\sigma}(t_m)]^T$ is the *firing vector* that corresponds to $\sigma$. Furthermore, we use $|\sigma|$ to denote the number of transitions in sequence $\sigma$. A *cost function* $C : T \to Z^+$ assigns to each transition in the net a positive integer cost. Given a transition firing sequence $\sigma = t_{i1}t_{i2}\ldots t_{ik}$, its total cost is given by $C(\sigma) = \sum_{j=1}^{k} C(t_{ij}) = \sum_{j=1}^{m} C(t_j)\cdot\overline{\sigma}(t_j)$.

**Definition 1** Given an initial marking $M_0$, the set of reachable markings with respect to transition firing sequences

of length $L$ is given by $Z(L) = \{M' \mid \exists \sigma : M_0[\sigma\rangle M'$ and $|\sigma| = L\}$.

## III. Problem Formulation

The problem we deal with in this paper is the following. Consider a WSN that is modeled as a Petri net $N$ with an initial state $M_0$ and costs associated with each transition (via a cost function $C$). Given a final state $M$ (i.e., a target state to be reached) and a positive integer number $L$ (i.e., an upper bound on the length of sequences of transitions whose firings can lead us from the initial state to the final state), we aim at finding the transition firing sequence(s) with length less than or equal to $L$ which: (i) is (are) consistent with the structure of the Petri net, (ii) leads (lead) us from the initial state to the final state, and (iii) has (have) the least total cost.

Clearly, the set of least-cost firing sequence(s) $\{\sigma_{min}\}$ is the solution to the following problem:

$$arg\min_{\sigma} C(\sigma) \quad s.t. \quad M_0[\sigma\rangle M \quad \& \quad |\sigma| \leq L. \qquad (1)$$

**Example 1** Consider the Petri net shown in Fig. 1 with places $P = \{p_1, p_2, p_3, p_4\}$; transitions $T = \{t_1, t_2, t_3, t_4, t_5\}$; and transition costs given by $C(t_1) = 1$, $C(t_2) = 4$, $C(t_3) = 2$, $C(t_4) = 3$, and $C(t_5) = 5$. The initial marking is given by $[2\ 0\ 0\ 0]^T$ and the final marking is taken to be $[1\ 0\ 1\ 0]^T$, whereas the upper bound on the length of the transition firing sequences is set to $L = 2$. In other words, we need to find, among all valid transition firing sequences of length equal to or less than 2 that lead us from the initial marking to the final marking, the ones that have the least total cost. It is not hard to show that there are two possible transition firing sequences, $t_2$ and $t_1t_3$, with total costs given by 4 and 3 respectively. Therefore, we conclude that the least-cost transition firing sequence (within length 2) that leads us from the initial marking to the final marking is $t_1t_3$. □

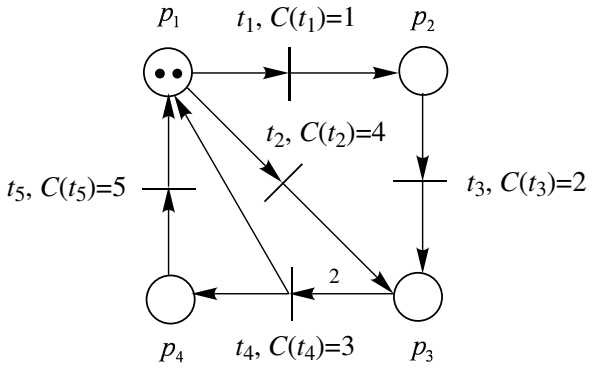



Figure 1. A Petri net with transition costs.

The problem in (1) could be solved by starting from the initial marking, enumerating all transition firing sequences $\sigma$ with length equal to or less than $L$, evaluating whether they are enabled or not, if enabled then computing the markings $M'$ resulting from their firings, determining whether they satisfy $M' = M$, and obtaining the valid one(s) with the least cost. The problem with this approach is that, in the worst case, the number of transition sequences considered is exponential in the length of $L$ (e.g., we start with $\sum_{i=1}^{L} m^i$ possible transition firing sequences). However, by looking at this problem in a different way, i.e., in terms of a trellis diagram [29], we will show that one can use a dynamic programming approach to obtain the solution more efficiently in an iterative manner. This approach will take advantage of the fact that several of these sequences visit identical markings at identical points in time (and therefore need not to be explored separately).

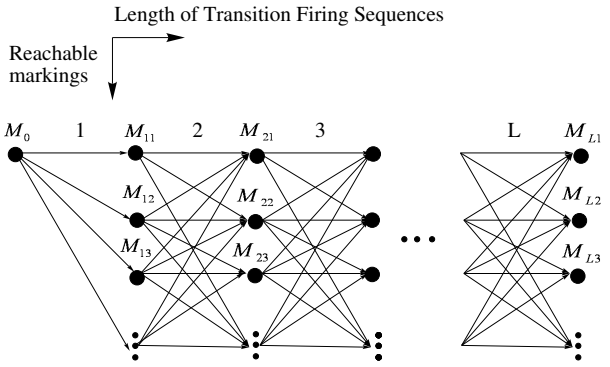



Figure 2. Trellis diagram of the evolution of reachable markings.

In Fig. 2, We depict a trellis diagram that captures the evolution of reachable markings as the length of transition firing sequences increases. In particular, $\{1, 2, \ldots, L\}$ denotes the length of transition firing sequences with time epochs (stages) corresponding to each time a transition fires. Each node in the trellis diagram (drawn as a big black dot) denotes a marking that is reachable from the initial marking through transition firing sequences of length up to the current time epoch, i.e., $M_{ji} \in Z(j)$ where $j \in \{1, 2, \ldots, L\}$ and $i$ is the index of a given marking within the set $Z$. Arcs between nodes represent transitions whose firings will lead from one marking to another. Given the initial marking $M_0$, the final marking $M$, and the upper bound $L$ on the length of transition firing sequences, we need to find the set of transition firing sequences that appear in the trellis diagram and have the least total cost from the initial marking $M_0$ to the final marking $M$.

**Definition 2** The set of least-cost firing sequences (of length $j$) that lead to the $i^{th}$ marking $M_{ji}$ at the $j^{th}$ stage of the trellis diagram is given by $LC_i^{(j)} = \{\sigma' \mid \sigma' = arg\min_{\sigma} C(\sigma)$ for $\sigma$ such that $M_0[\sigma\rangle M_{ji}$ and $|\sigma| = j\}$.

By formulating the problem in terms of a trellis diagram, it is clear that each reachable marking in the set $Z(j+1)$ has to be reached via at least one reachable marking in the set $Z(j)$. Moreover, a dynamic programming approach can be used to compute the least-cost firing sequence(s) iteratively. The basic observation is that the transition firing sequences which have the least cost at time epoch $j+1$ only depend on the least-cost transition firing sequences that reach a marking at time epoch $j$, and

the transitions that can fire at time epoch $j+1$ (from the markings reached at time epoch $j$). By taking advantage of this observation, one can search for the sequence that has the least cost, one stage in the trellis diagram at a time.

In our setup, at time epoch $j$, given each reachable marking $M_{ji} \in Z(j)$ and its associated set of least-cost firing sequences $LC_i^{(j)}$, we can compute the set of least-cost firing sequences $LC_i^{(j+1)}$ associated with each reachable marking $M_{(j+1)i} \in Z(j+1)$ at time epoch $j+1$ as follows:

$$LC_i^{(j+1)} = \{\sigma_{j'}t_p \mid [\sigma_{j'}, t_p] = arg \min_{\sigma_{j'},\ t_p} (C(\sigma_{j'}) + C(t_p))\}$$

where $\sigma_{j'}$ and $t_p$ are such that,

$$\exists i' \ s.t. \ \sigma_{j'} \in LC_{i'}^{(j)} \text{ and } M_{ji'}[t_p\rangle M_{(j+1)i}. \qquad (2)$$

By calculating (2) iteratively in the length of the transition firing sequences, we can efficiently find the firing sequence(s) that has (have) the least cost to all reachable markings in the trellis diagram. Note that we want to find a particular final marking $M$ that can be reached from the given initial marking $M_0$. Therefore, once we compute all reachable markings at each stage in the trellis diagram, we need to check whether marking $M$ has appeared or not. If $M$ first appears at a particular stage, the least-cost firing sequence(s) that may lead us from $M_0$ to $M$ is (are) not necessarily these one(s), because $M$ can potentially be reached from $M_0$ via a different transition firing sequence that has longer length but smaller cost ($M$ may be reached from the initial marking via transition firing sequences of different lengths). Therefore, we must keep track of all reachable markings within $L$ steps and select, among all sequences whose firings can lead us from $M_0$ to $M$, the one(s) that has (have) the least total cost.

Note that each time the final marking $M$ appears in the trellis diagram, we do not need to consider the sequences that emanate from it in the later stage. Since each transition in the net is associated with a positive cost, these sequences emanating from $M$ are guaranteed to have higher cost even if they reach $M$ again.

## IV. Obtaining Least-Cost Paths

### A. Algorithm

In this section, we propose an iterative algorithm to find the least-cost firing sequence(s), of length less than or equal to the given bound $L$, that lead from the initial marking $M_0$ to the final marking $M$. We use a data structure $\mathscr{C} = (M_{current}, leastcost, (t_{in}, M_{previous}))$ to capture the information we need to store for each node in the trellis diagram. More specifically, at time epoch $j$: $M_{current}$ denotes the marking that is associated with the node (and can be reached from $M_0$ through a transition firing sequence of length $j$); $leastcost$ is the least cost among all valid firing sequences from $M_0$ to $M_{current}$; ($t_{in}$, $M_{previous}$) denotes that input transition $t_{in}$ that is enabled at $M_{previous}$ such that the least cost firing sequence goes through $M_{previous}$ at time epoch $j-1$ and leads to $M_{current}$ by firing transition $t_{in}$.

Note that if $M$ can be reached from $M_0$ via one or more transition firing sequences, the algorithm outputs the one(s) that has (have) the least total cost; if no such firing sequence exists, the algorithm does not provide any output. We describe the algorithm in details below.

**Algorithm 1**

**Input**: A Petri net $N$ with positive costs associated with each transition, a given initial marking $M_0$, a final marking $M$, and a maximum length $L$ for transition firing sequences.

1. $\mathscr{C}(0) = \{(M_0,\ 0,\ (\emptyset,\ \emptyset))\}$.
2. Let $j = 1$.
3. Set $\mathscr{C}(j) = \emptyset$.
4. For all $R \in \mathscr{C}(j-1)$ and $R.M_{current} \neq M$ do
   For all $t$ such that $R.M_{current}[t\rangle$
   compute $M' = R.M_{current} + B(\cdot, t)$
   If $M'$ is a new marking that has not appeared in $\mathscr{C}(j)$
   $\mathscr{C}(j) = \mathscr{C}(j) \cup \{(M', R.leastcost + C(t), (t, R.M_{current}))\}$
   Else
   $M'$ has appeared in $R' \in \mathscr{C}(j)$
   If $R.leastcost + C(t) < R'.leastcost$
   $R' = (M', R.leastcost + C(t), (t, R.M_{current}))$
   Else If $R.leastcost + C(t) = R'.leastcost$
   $R' = (M', R'.leastcost, R'.(t_{in}, M_{current}) \cup (t, R.M_{previous}))$
   End IF
   End IF
   End For
   End For
5. $j = j + 1$.
6. If $j = L + 1$, Goto 7; else Goto 3.
7. Recover all firing sequences from $M_0$ to $M$ using the information stored and output the one(s) that has (have) the least total cost. □

Given the Petri net structure its initial marking $M_0$, Algorithm 1 iteratively computes markings that are reachable from the initial marking along with the least cost of transition sequences that lead from the initial marking to each of these markings. At each time epoch $j$, the algorithm looks at the set of all enabled transitions and associates each new marking with the transition whose firing (from a reachable marking at stage $j-1$) leads to an overall transition firing sequence that reaches that marking with least total cost. The algorithm then stores all reachable markings and their corresponding least-cost transition(s), and goes to the next step (where the length of the transition firing sequence is increased by one). After considering all valid transition firing sequences with length equal to or less than $L$ (i.e., after we complete $L$ stages of the trellis diagram), the algorithm recovers all transition firing sequences from the initial marking $M_0$ to the final marking $M$, and outputs the one(s) that has (have) the least total cost.

**Example 2** Recall the Petri net with transition costs shown in Fig. 1. Given the final marking as $M =$

$[1\ 0\ 1\ 0]^T$, we want to find the least-cost transition firing sequences within length 2 (i.e., $L=2$) that could lead us from the initial marking to this marking. After running Algorithm 1 on this example, the corresponding trellis diagram is shown in Fig. 3. The set of firing sequences that can lead us from $M_0$ to $M$ is given by $\sigma=\{t_2,t_1t_3\}$. The least-cost firing sequence is $\sigma_{min}=t_1t_3$ and has cost 3. Note that although $M_{12}$ and $M_{21}$ have the same marking, they correspond to different data structures in our algorithm (because they are reached via sequences of different length): one is given by $([1\ 0\ 1\ 0]^T,4,(t_2,[2\ 0\ 0\ 0]^T))$ and the other by $([1\ 0\ 1\ 0]^T,3,(t_3,[1\ 1\ 0\ 0]^T))$. Also, after the final marking is reached at $M_{12}$, we do not consider transitions emanating from it because the sequence will have higher cost when it reaches the final marking again. □

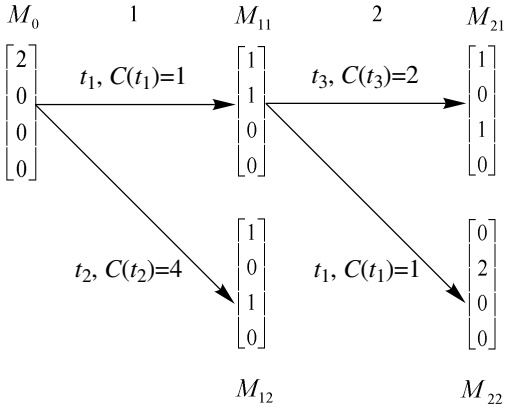



Figure 3. Trellis diagram after running Algorithm 1 for Example 1.

### B. Complexity Analysis

Before discussing the complexity of our algorithm, we would like to first translate our problem into the context of labeled Petri nets; then we will use results established in [26] for labeled Petri nets to analyze the complexity of Algorithm 1.

Given the (unlabeled) Petri net structure and its initial marking, we can think of it as a labeled Petri net with all transitions in the net associated with a same label $l$ through a degenerate labeling function $E$ defined below.

A *labeling function* $E: T \rightarrow \Sigma$ assigns to each transition in the net a label from a given alphabet $\Sigma$. In our setup, we assume that there is only one label $l \in \Sigma$. Therefore, given a transition firing sequence $\sigma = t_{i1}t_{i2}\ldots t_{iL}$ of length $L$, the corresponding label sequence is given by $\omega = E(\sigma) = E(t_{i1})\cdots E(t_{iL}) = \underbrace{l\cdots l}_{L} = l^L$, i.e., a string of length $L$.

By thinking of the problem in this way, the upper bound $L$ on the number of transitions in a particular transition sequence translates to an upper bound on the number of labels in its corresponding label sequence. Therefore, the problem can be represented equivalently as follows: Consider a labeled Petri net $N$ with initial marking $M_0$, positive costs associated with each transition (via a cost function $C$), and a single label $l$ associated with all transitions; given an observed label sequence $\omega = l\cdots l$ of length $L$ that has been generated by an underlying (unknown) firing sequence $t_{i1}t_{i2}\ldots t_{iL}$ and a final marking $M$, find the transition firing sequence(s) that: (i) is (are) consistent with the structure of the net, (ii) leads (lead) us from the initial marking $M_0$ to the marking $M$, (iii) has (have) length less than or equal to $L$, and (iv) has (have) the least total cost.

It was shown in [26] that given an observed sequence of labels of length $L$, the total number of markings at the $L^{th}$ stage in the trellis diagram is upper bounded by a polynomial function in $L$, namely $L^b$ where $b$ is a constant associated with structural parameters of the given Petri net[1]. We use this fact in our algorithmic complexity analysis below.

The complexity of Algorithm 1 can be obtained as follows. First, regarding space complexity, the storage needed is proportional to the number of reachable markings (nodes). For each reachable marking (apart from the marking information itself and the least cost to get to it), we need to store the valid pairs of transitions and reachable markings in the previous time epoch that could lead to this marking and have the least cost. Since there are total $m$ transitions in the net, the number of transitions that could lead to the current (reachable) marking from distinct (reachable) markings in the previous stage is bounded by $m$. Thus, for each reachable marking, the information to be stored is proportional to $m$. Clearly, since the number of reachable markings at the $L^{th}$ stage in the trellis diagram is upper bounded by $L^b$, the total space needed to store all reachable markings at every stage in the trellis diagram is $\sum_{j=1}^{L} O(m\cdot j^b)$ which can be simplified as $O(mL\cdot L^b) = O(mL^{b+1})$, i.e., the storage required is polynomial in the length $L$ of the transition firing sequences.

We now proceed to analyze computational complexity. We use $n_{j-1}$ to denote the number of reachable markings at the $(j-1)^{st}$ stage of the trellis diagram ($n_{j-1} = O((j-1)^b)$ and $n_j$ to denote the number of reachable markings at the $j^{th}$ stage of the trellis diagram ($n_j = O(j^b)$). Clearly, at time epoch $j$, the number of possible transitions enabled from the $(j-1)^{st}$ stage is upper bounded by $n_{j-1}\cdot m$ where $m$ is the number of transitions in the net. For each marking (in the $j^{th}$ stage) yielding from these transitions, we need at most $n_j$ comparisons to decide if it has appeared or not (by searching through the set of existing markings) and at most $n_j$ comparisons to decide if it is equal to the final marking $M$. Thus, the computational complexity for finding the sequence that has least-cost for all markings at the $j^{th}$ stage is bounded by $n_{j-1}\cdot m\cdot 2n_j$, which has complexity $O(m\cdot j^{2b})$. Thus, for reachable markings over all stages, the total

[1]More specifically, in [26] it is argued that $b = c(d-1)$ where $c$ is the number of nondeterministic labels in the net and $d$ is the maximum number of transitions corresponding to a label in the net. We say a label is nondeterministic if there is more than one transition corresponding to the label. In our setup, $c=1$ and $d=m$, thus $b=m-1$.

computational complexity is given by $\sum_{j=1}^{L} O(m \cdot j^{2b})$, which can be simplified as $O(mL^{2b+1})$. Therefore, the algorithm has complexity that is polynomial in the length of the transition firing sequences.

## V. An Illustrative Example

In this section, we illustrate the algorithm via an example of a WSN. Consider a WSN that is modeled by the Petri net shown in Fig. 4. Initially, six pieces of information are available (modeled by places $p_1$ to $p_6$ where each place has a token). Following a number of different ways of transmission, the information transmission is completed (modeled by $p_{12}$). Therefore, the final state is given by $M = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1]^T$). The Petri net has 12 places $P = \{p_1, p_2, \ldots, p_{12}\}$, 8 transitions $T = \{t_1, t_2, \ldots, t_8\}$ and the initial state $M_0 = [1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0]^T$. We assume that the cost of each transition is given by the cost vector $C = [5\ 5\ 30\ 20\ 20\ 20\ 30\ 10]^T$. Our goal is to estimate the least-cost transition firing sequence(s) that leads (lead) us from the initial state to the final state under different length constraints.

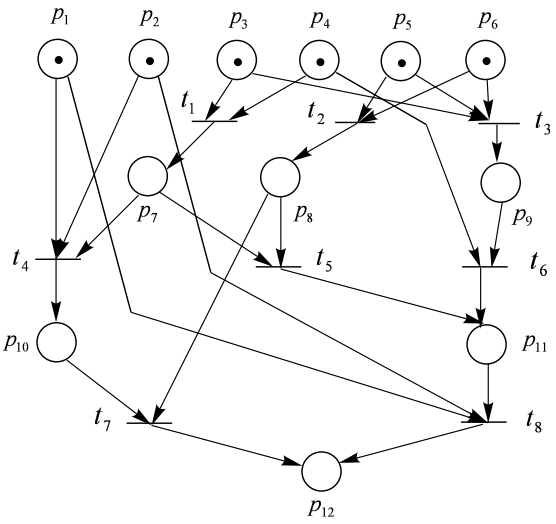



Figure 4. A Petri net model of a WSN.

Assume that we want to find the least-cost transition firing sequences from $M_0$ to $M$ within length 4. In this case, Algorithm 1 finds the set of least-cost firing sequences to be $\sigma_{min} = \{t_1t_2t_5t_8, t_2t_1t_5t_8\}$ with total cost 40. We provide the following table where *Length* denotes the allowable maximum length of the transition firing sequences considered so far, *Leastcost* captures the least total cost of sequence(s) that satisfies (satisfy) the maximum length constraint and leads (lead) to $M$ from $M_0$ (we use $\infty$ to denote that $M$ cannot be reached via transition sequences of the given maximum length from $M_0$), and $\{\sigma_{min}\}$ is the set of least-cost transition sequence(s) that has (have) least total cost and leads (lead) from $M_0$ to $M$.

| *Length* | *Leastcost* | $\{\sigma_{min}\}$ |
|---|---|---|
| 1 | $\infty$ | $\emptyset$ |
| 2 | $\infty$ | $\emptyset$ |
| 3 | 60 | $t_3t_6t_8$ |
| 4 | 40 | $t_1t_2t_5t_8$<br>$t_2t_1t_5t_8$ |

**Remark 1** Note that the final marking is reached via transition firing sequence $t_3t_6t_8$ of length 3 with least cost 60; however, the final marking can also be reached via transition firing sequences of longer length with smaller cost (both $t_1t_2t_5t_8$ and $t_2t_1t_5t_8$ have length 4 with least cost 40). Therefore, in order to capture the least-cost transition firing sequences of length less than or equal to $L$, we need to consider all markings reached from $M_0$ within $L$ stages of the trellis diagram.

## VI. Conclusions and Future Work

In this paper, we consider the problem of finding the least-cost paths in a WSN that is modeled by a Petri net. In particular, we consider a setting where we are given an initial state and a final state along with a positive integer $L$ that serves as an upper bound on the length of paths that lead us from the initial state to the final state. We assume that each transition in the given net is associated with a positive cost (which could represent its likelihood of occurrence by considering different constraints) and we aim at finding the paths (of length less than or equal to $L$) which: (i) are consistent with the structure of the Petri net, (ii) lead us from the initial state to the final state, and (iii) have the least total cost. We develop an iterative algorithm that obtains the least-cost paths with complexity that is polynomial in $L$. This algorithm can be used for least-cost path estimation in WSNs that are modeled as Petri nets.

One direction for future work is to study the timed Petri net model to incorporate temporal information of sensor transmissions. Another interesting extension is to incorporate the constraint of energy consumption of sensor nodes in our Petri net model to optimize the performance of WSNs.

## References

[1] C.-Y. Chong and S. P. Kumar, "Sensor networks: evolution, opportunities, and challenges," *Proc. of the IEEE*, vol. 91, no. 8, pp. 1247–1256, August 2003.

[2] F. Batool and S. Khan, "Traffic estimation and real time prediction using adhoc networks," in *Proc. IEEE Symp. Emerging Technologies,* pp. 264–269, September 2005.

[3] S. Biswas, R. Tatchikou, and F. Dion, "Vehicle-to-vehicle wireless communication protocols for enhancing highway traffic safety," *IEEE Communications Magazine,* vol. 44, no. 1, pp. 74–82, January 2006.

[4] M. Bhardwaj, T. Garnett, and A. Chandrakasan, "Upper bounds on the lifetime of sensor networks," in *Proc. IEEE Intl. Conf. Communications,* vol. 3, pp. 785–790, June 2001.

[5] C. Jaikaeo, C. Srisathapornphat, and C.-C. Shen, "Diagnosis of sensor networks," in *Proc. IEEE Intl. Conf. Communications,* vol. 5, pp. 1627–1632, June 2001.
[6] B. Warneke, M. Last, B. Liebowitz, and K. Pister, "Smart dust: communicating with a cubic-millimeter," *Computer,* vol. 34, no. 1, pp. 44–51, January 2001.
[7] L. Yin, T. Hong, C. Liu, Y. Xia, and H. Zhou, "A reprogramming protocol based on state machine for wireless sensor network," in *Proc. Intl. Conf. Electrical and Control Engineering,* pp. 232–235, December 2010.
[8] O. Kasten and K. Romer, "Beyond event handlers: programming wireless sensors with attributed state machines," in *Proc. 4th Intl. Symp. Information Processing in Sensor Networks,* pp. 45–52, April 2005.
[9] X. Ye and J. Li, "An FSM-based automatic detection in AODV for ad hoc network," in *Proc. Intl. Symp. Computer Network and Multimedia Technology,* pp. 1–5, December 2009.
[10] S. Kokalj-Filipovic, P. Spasojevic, and R. Yates, "Geographic data propagation in location-unaware wireless sensor networks: a two-dimensional random walk analysis," *IEEE Journal on Selected Areas in Communications,* vol. 27, no. 7, pp. 1158–1168, September 2009.
[11] C. Chi-Kin, Q. Fei, S. Sayed, M. H. Wahab, and Y. Yang, "Harnessing battery recovery effect in wireless sensor networks: Experiments and analysis," *IEEE Journal on Selected Areas in Communications,* vol. 28, no. 7, pp. 1222–1232, September 2010.
[12] B. Bostan-Korpeoglu, A. Yazici, I. Korpeoglu, and R. George, "A new approach for information processing in wireless sensor network database applications," in *Proc. 22nd Intl. Conf. Data Engineering Workshops*, April 2006.
[13] D. He, L. Cui, H. Huang, and M. Ma, "Design and verification of enhanced secure localization scheme in wireless sensor networks," *IEEE Trans. on Parallel and Distributed Systems*, vol. 20, no. 7, pp. 1050–1058, July 2009.
[14] A. Shareef and Y. Zhu, "Energy modeling of processors in wireless sensor networks based on Petri nets," in *Proc. 37th Intl. Conf. Parallel Processing*, pp. 129–134, September 2008.
[15] A. Shareef and Y. Zhu, "Energy modeling of wireless sensor nodes based on Petri nets," in *Proc. 39th Intl. Conf. Parallel Processing*, pp. 101–110, September 2010.
[16] C.-H. Kuo and J.-W. Siao, "Petri net based reconfigurable wireless sensor networks for intelligent monitoring systems," in *Proc. Intl. Conf. Computational Science and Engineering*, pp. 897–902, August 2009.
[17] C.-H. Kuo and T.-S. Chen, "PN-WSNA: An approach for reconfigurable cognitive sensor network implementations," *IEEE Sensors Journal*, vol. 11, no. 2, pp. 319–334, February 2011.
[18] S. Yang, H. Cheng, and F. Wang, "Genetic algorithms with immigrants and memory schemes for dynamic shortest path routing problems in mobile ad hoc networks," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 40, no. 1, pp. 52–63, January 2010.
[19] T. Qiu, F. Xia, L. Feng, G. Wu, and B. Jin, "Queueing theory-based path delay analysis of wireless sensor networks," *Advances in Electrical and Computer Engineering*, vol. 11, no. 2, pp. 3–8, March 2011.
[20] Z. Duan, F. Guo, M. Deng, and M. Yu, "Shortest path routing protocol for multi-layer mobile wireless sensor networks," in *Proc. IEEE Intl. Conf. Networks Security, Wireless Communications and Trusted Computing*, pp. 106–110, April 2009.
[21] Y.-L. Lai and J.-R. Jiang, "Optimal multipath planning for intrusion detection in smart homes using wireless sensor and actor networks," in *Proc. 39th IEEE Intl. Conf. Parallel Processing Woshshops,* pp. 562–570, September 2010.
[22] Z. Yu, X. Fu, Y. Cai, and M. C. Vuran, "A reliable energy-efficient multi-level routing algorithm for wireless sensor networks using fuzzy Petri nets," *Journal of Sensors*, vol. 11, no. 3, pp. 3381–3400, March 2011.
[23] A. Giua, "Petri net state estimators based on event observation," in *Proc. 36th IEEE Intl. Conf. Decision and Control*, pp. 4086–4091, December 1997.
[24] A. Giua and C. Seatzu, "Observability of place/transition nets," *IEEE Trans. on Automatic Control*, vol. 47, no. 9, pp. 1424–1437, September 2002.
[25] L. Li and C. N. Hadjicostis, "Least-cost transition firing sequence estimation in labeled Petri nets with unobservable transitions," *IEEE Trans. on Automation Science and Engineering,* vol. 8, no. 2, pp. 394–403, April 2011.
[26] Y. Ru and C. N. Hadjicostis, "Bounds on the number of markings consistent with label observations in Petri nets," *IEEE Trans. on Automation Science and Engineering*, vol. 6, no. 2, pp. 334–344, April 2009.
[27] T. Murata, "Petri nets: properties, analysis and applications," *Proceedings of the IEEE,* vol. 77, no. 4, pp. 541–580, April 1989.
[28] C. G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems,* Springer, 1999.
[29] S. Lin, T. Kasami, T. Fujiwara, and M. Fossorier, *Trellises and Trellis-Based Decoding Algorithms for Linear Block Codes,* Kluwer Academic Publishers, 1998.

**Lingxi Li** received the B.E. degree in Automation from Tsinghua University, China, in 2000, the M.S. degree in Control Theory and Control Engineering from the Institute of Automation, Chinese Academy of Sciences, China, in 2003, and the Ph.D. degree in Electrical and Computer Engineering from University of Illinois at Urbana-Champaign, USA, in 2008.

In 2008 he joined the Faculty at Indiana University-Purdue University Indianapolis (IUPUI) where he is currently an Assistant Professor with the Department of Electrical and Computer Engineering. His research interests include modeling, diagnosis, and control of complex systems; fault-tolerant systems; discrete event systems; graph theory; with applications to intelligent transportation systems, power systems, communication networks, and biological systems.

**Dongsoo Stephen Kim** received the M.S. degree in Computer Science from the University of Texas at Dallas, TX, USA, in 1994, and the Ph.D. degree in Computer Science and Engineering from the University of Minnesota, Minneapolis, MN, USA, in 1998. Dr. Kim worked as a research scientist for Electronics and Telecommunications Research Institute from 1986 to 1992, and as a project manager for Megaxess Inc. from 1998 to 2000. In 2000, he joined the Department of Electrical and Computer Engineering, Purdue School of Engineering and Technology, Indiana Univ. Purdue Univ. Indianapolis, Indiana, USA. He is currently an Associate Professor of the Department, Indiana University Purdue University Indianapolis.

His research includes switch networks, optical switches, network survivability, protection switching, network planning, quality-of-service provisioning in the Internet, mobile ad-hoc networks, mobility modeling, sensor networks, and power-aware routing.

# Analytic Hierarchy Process aided Key Management Schemes Evaluation in Wireless Sensor Network

Ruan Na †, Yizhi Ren ‡, Yoshiaki Hori †, Kouichi Sakurai †
† Department of Informatics, Kyushu University, Fukuoka, Japan
‡ School of Software Engineering, Hangzhou Dianzi University, China
Email: {ruannana, renyizhi}@gmail.com, {hori, sakurai}@inf.kyushu-u.ac.jp

***Abstract*— Wireless sensor networks (WSNs) have been widely used in various applications. Since their sensor nodes are resource-constrained and their security primitives need to store a set of security credentials to share a secure channel, key management is one of the most challenging issues in the design of WSN. Currently, various efficient lightweight key management schemes (KMs) have been proposed to enable encryption and authentication in WSN for different application scenarios. According to different requirements, it is important to select the trustworthy key management schemes in a WSN for setting up a fully trusted WSN mechanism. In this context, adaptive methods are required to evaluate those schemes.**

**In this paper, we exploit Analytic Hierarchy Process (AHP) to help with the complex decision. Specifically, we consider the following performance criteria: *scalability, key connectivity, resilience, storage overhead, processing overhead and communication overhead*. Two case studies are added for verifying our proposal. Via the two case studies, it is verified that our method is able to help selecting a suitable scheme for given requirements.**

***Index Terms*— Analytic Hierarchy Process, Key management scheme, Trustworthy decision, Wireless sensor network**

## I. Introduction

### A. Background

The advance in miniaturization techniques and wireless communications has made possible the creation and subsequent development of the wireless sensor network (WSN) paradigm [1]. The application area of WSN includes military sensing and tracking, environmental monitoring, patient monitoring and smart environment. When a sensor node is installed in a dangerous and untrusted area, its security becomes very important. Thus, WSN security is a prerequisite for wider use [2]. The communication channels between any pair of nodes inside WSN must be protected to avoid attacks from external parties. Such protection, in terms of confidentiality, integrity and authentication, is provided by some security primitives. A key management scheme is an important security primitive for WSN. The task of generating and distributing those keys has to be done by a global key management system [3]. For the above reasons, designing a trustworthy key management scheme is a necessary work. Meanwhile, to select a appropriate key management scheme is a necessary work.

In this paper, we design an evaluation method which supports the decision-making processes of selecting a trustworthy key management scheme in a WSN. We focus on the calculation of how much the existing key management schemes can be appropriate to perform a particular application. The trust of the trustworthy decision is based on the firm belief in the reliability under the assumed wireless sensor network scenario. The key management schemes must satisfy traditional needs of security, such as availability, integrity, confidentiality, authentication and non-reputation [6] in a typical wireless network. Compared with the typical wireless network, the key management has other special challenges such as resilience, expansibility and efficiency [7] in WSN because of its specificity.

### B. Related Work

Recent research works focus on producing an efficient system to evaluate these key management schemes. In recent years, there has been a significant progress on key management schemes in WSN. Researchers have proposed a large number of key management schemes in WSN which focus on different security requirements. Each scheme has its own advantages and disadvantages. Even though quite a number of key management schemes in wireless sensor network exist now, they can be divided into six categories. The six categories are stated as follows: Dedicated pair-wise key management solution in distributed wireless sensor network (DWSN), reusable pair-wise key management solutions in DWSN, group-wise key management solutions in DWSN, pair-wise key management solutions in hierarchical wireless sensor network (HWSN), group-wise key management solutions in HWSN and network-wise key management solutions in HWSN [3]. If changing into another perspective, because WSN is energy limited network and pre-distributed key

This paper is based on "A Generic Evaluation Method for Key Management Schemes in Wireless Sensor Network," by R. Na, Y. Ren, Y. Hori and K. Sakurai, which appeared in the Proceedings of the 5th International Conference on Ubiquitous Information Management and Communication (ICUIMC), Seoul, Korea, February 2011. © 2011 ACM.

This work was supported by the governmental scholarship from China Scholarship Council.


doi:10.4304/jnw.7.1.88-100

management scheme is energy-efficient scheme, most of the key management schemes in WSN are based on pre-distribution key management schemes. Commonly used key management schemes in WSN are listed as follows: random pre-distribution key management scheme based on key-pool [8]; pre-distribution key management scheme based on polynomial [9]; pre-distribution key management scheme based on block design [10]; pre-distribution key management scheme based on position [11]; pre-distribution key management scheme based on matrix [12] and so on [13].

Some researchers proposed certain evaluation indexes for qualitative evaluation of these key management schemes in WSN [3]. However, such proposals have limited utility unless they take node replication attacks and robustness into consideration. Their proposals fail to address all of the criteria that a key management scheme in WSN should satisfy to.

In this paper, we propose a generic method to evaluate key management schemes, which can help researchers to select the scheme quantitatively according to different network requirements. The most related work to our research on security evaluation is Hwang *et al.* [5]. It employs the Analytical Hierarchy Process (AHP) method in guiding information security policy decision making. It uses the application of AHP as a method to develop information security decision model for information security policy. Meanwhile, after comprehensively surveying all of the criteria for KMs evaluation in WSN, we propose an AHP-aided method to select the optimum key management scheme for an assumed WSN.

### C. Challenging Issues

The following reasons motivate us to propose the AHP-aided method for evaluating key management schemes in wireless sensor network.

1) The security of a WSN depends on the existence of efficient key management solutions [3]. Many key establishment techniques have been designed to address the trade off between limited computational resources and security requirements, but it is not easy to determine which scheme is the best one in an assumed scenario.
2) All these key management schemes have their own advantages and disadvantages. All of them can be suitable for different needs. Comprehensive consideration of the parameters selection is not a simple problem.
3) To select the most proper key management scheme quantitatively from a large amount of existing schemes is not an easy issue [15].
4) Despite the utmost importance of a generic evaluation method for these key management schemes, it is surprising that we find almost nothing in literature on this subject.

### D. Our Contribution

In this paper, we propose an evaluation method to evaluate the key management schemes, which can help us to select the scheme quantitatively according to different network requirements. The contributions of our paper can be summarized as follows:

1) We use an analytical hierarchy process (AHP) model to construct a framework to do the decision making. AHP can help with a quantitatively decision. Thus, we can overcome the difficulty in selecting a proper key management scheme for wireless sensor network having multiple criteria decision making.
2) Based on our proposal, we provide evaluation and analysis of the existing key management schemes. We show that our method can build an intuitive method to select a proper scheme and to present key management schemes in the order of suitability, based on the previously given network requirements. In a word, we provide a feasible quantitative evaluation system to select the best key management scheme from various schemes.
3) Finally, we classify several typical key management schemes and make a comparison among the trade off in those schemes. At the same time, we can obtain quantitative analysis results via two kinds of case study. In other words, our method can be helpful in a complicated network environment.

This work is organized as follows: Section II describes basic definitions and notions used in wireless sensor network for key management schemes evaluating. At the same time, corresponding case study is proposed. Section III provides our quantitative system which based on linear algebra and focused on matrix. Section IV discusses the system in details via two case studies. Finally, we draw conclusions in Section V.

## II. Preliminaries

### A. Brief reviews of AHP

In a set number of application domains, the Analytical Hierarchy Process (AHP) is a decision approach designed to aid in the solution of complex multiple criteria problems. It was developed by Thomas L.Saaty in the 1980s [4]. This method has been found to be an effective and practical approach that can make complex and unstructured decisions. The AHP has been used in a large number of applications to provide certain structures on a decision making process. When used in the systems engineering process, AHP can be a powerful tool for comparing alternative design concepts. The decision-maker judges the importance of each criterion in pairwise comparisons. The outcome of AHP is a prioritized ranking or weighted of each decision alternative. There are three steps for considering decision problems by AHP: constructing hierarchies, comparative judgments, and synthesis of priorities.

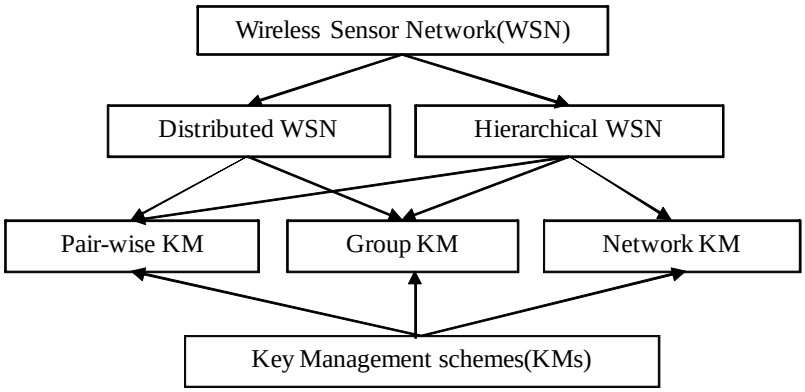



Figure 1. Classification of Key Management schemes

1) **Construction hierarchies:** User of the AHP first decomposes his decision problem into some hierarchy of more easily comprehended sub problems, each of them can be analyzed independently.
2) **Comparative judgments:** After the hierarchy is built, the decision makers systematically evaluate various elements of the hierarchy by comparing each one of them to another one of them at a time. In making the comparisons, the decision makers can either use concrete data about the elements or use their judgments about the elements' relative meaning and importance. The AHP converts these evaluations to numerical values that can be processed and compared over the entire range of the problem.
3) **Synthesis of Priorities:** Numerical priorities are calculated for each of the decision alternatives. These numbers represent the alternatives' relative ability to achieve the decision goal. Something are presumable missing in allowed range.

The above three steps show a brief review of AHP hierarchy for the decision making process.

Futhermore, details on both the synthesis of priorities and the measurement of consistency are claimed as follows [4]:

- $n$: the order of the matrix. The AHP authors use $n$ for explaining the size of these matrixes in AHP method. In section III, the matrix of hierarchies and the matrix of judgements will be used in our AHP-aided method.
- $\lambda$: the eigenvalue of the matrix. Maximum value of $\lambda$ is expressed by $\lambda_{max}$. If we want to calculate the consistency ratio, we should calculate the eigenvector of the relative weights $\lambda_{max}$ for each matrix with order $n$.
- $RI$: the average Random Index for consistency checking. $RI$ is a known random consistency index obtained from a large number of simulations which run and vary depending upon the order of matrix. Tables I shows the value of the $RI$ for matrix with the size from order 1 to 10 [16].
- $CI$: the Consistency Index. $CI$ for each matrix of order $n$ can be calculated by using the formula: $CI = (\lambda_{max} - n)/(n-1)$.
- $CR$: the Consistency Ratio. $CR$ is calculated by using the formula: $CR = CI/RI$.

TABLE I.
AVERAGE RANDOM INDEX (RI) BASED ON MATRIX SIZE

| Size of matrix(n) | Random consistency Index(RI) |
|---|---|
| 1 | 0 |
| 2 | 0 |
| 3 | 0.52 |
| 4 | 0.89 |
| 5 | 1.11 |
| 6 | 1.25 |
| 7 | 1.35 |
| 8 | 1.40 |
| 9 | 1.45 |
| 10 | 1.49 |

As constructing hierarchy is the first step of AHP, the pair-wise comparisons generate a matrix of relative rankings for each level of the hierarchy. The number of criteria depends on the number elements at each level. The order of the criteria at each level depends on its lower level number of elements. After all criteria are developed and all pair-wise comparisons are obtained, eigenvectors of the relative weights (the degree of relative importance among the elements), global weights and the maximum eigenvalue $\lambda_{max}$ for each matrix are calculated by using Expert Choice software (Expert Choice, 2000). The software is easy to use and understand. It provides visual representations of overall ranking on a computer screen.

The value of $\lambda_{max}$ is an important validating parameter in AHP. It is used as a reference index to screen information via calculating the consistency ratio $CR$ of the estimated vector. This step is in order to validate whether the pair-wise comparison matrix provides a completely consistent evaluation or not.

$n$-order matrix means the order of matrix $n$ equals $n$. In section III, our proposal which is based on AHP will use 5-order matrix and 6-order matrix. Thus, we present the consistency check of them in this section in advance.

When $n = 5$, we can calculate the eigenvalue of the matrix $\lambda$ for consistency check. The processes are as

follows:

1) Selecting $n = 5$ from Table I as an example, the average random index $RI$ is 1.11.
2) Because the matrix should be validated to pass the consistency check, the consistency ratio $CR$ need to be smaller than 0.1. Meanwhile, $CR$ equals $CI/RI$.
3) Thus, the consistency index $CI$ needs to satisfy: $CI < 0.1 \times 0.11 = 0.011$
4) Furthermore, as $CI = (\lambda - n)/(n-1) = (\lambda-5)/4$ and $CI < 0.011$, the maximum eigenvalue $\lambda$ is smaller than 5.444.
5) The 5-order matrix will pass the consistency check when $\lambda < 5.444$.

Similar to the process where $n = 5$, we do consistency check while $n = 6$. The result is as follows:

1) When there is $n = 6$, the average random index is $RI = 1.25$. Accordingly, the maximum eigenvalue $\lambda$ is smaller than 6.625.
2) The 6-order matrix will pass the consistency check when $\lambda < 6.625$.

### *B. Classification of key management schemes in WSN*

Key management schemes (KMs) in wireless sensor network (WSN) can be categorized into several types. Figure 1 explains the classification of KMs in WSN. WSN are organized in distributed or hierarchical structures in generally. WSN communication usually occurs in ad hoc manner, and shows similarities to wireless ad hoc network. When nodes in hierarchical WSN communicate, data flow may be classified into three parts: pair-wise (unicast) among pairs of sensor nodes and from sensor nodes to base station, group-wise (multicast) within a cluster of sensor nodes and network-wise (broadcast) from base stations to sensor nodes. Likewise, data flow in distributed WSN is similar to data flow in hierarchical WSN with a difference that network-wise (broadcast) messages can be sent by every sensor nodes.

As Table II shows, S. A. Camtepe *et al.* 2008 [3] classified the currently existing key management schemes based on the network structure. The network structure is classified into two types: Distributed WSN (DWSN) and Hierarchical WSN (HWSN). In DWSN, key management schemes (KMs) in DWSN are categorized into three types: dedicated pair-wise KMs, reusable pair-wise KMs and group-wise KMs. Meanwhile, KMs in HWSN are categorized into three types: pair-wise KMs, group-wise KMs and network-wise KMs. Our evaluation work follows this classification.

## III. Our Proposal based on AHP

In order to determine which key management scheme is the best for an assumed WSN scenario, we propose a method based on AHP.

In different proposed key management schemes, there have different parameters assumption even distinct assumption. It is not possible to give strict quantitative comparison criteria due to distinct assumptions made by these key management solutions. However, the following criteria can be used to evaluate and compare these key management schemes in WSN [3]. Our target is to give quantitative comparisons among various KMs in WSN based on these five criteria.

- **Scalability**: Ability of a key management solution to handle an increase in the WSN size.
- **Key connectivity**: Probability that a pair or a group of sensor nodes can generate or find a common secret key to secure their communication.
- **Resilience**: Resistance of the WSN against node capture and node replicate. The adversary often captures or replicates a sensor node, such as in some well-known network attacks in the WSN (e.g., sybil attack and wormhole routing attack). Keys which are stored on a sensor node or exchanged over radio links should not reveal any information about the security of any other links.
- **Storage overhead**: Amount of memory units required to store security credentials.
- **Processing overhead**: Amount of processing cycles required by each sensor node to generate or find a common secret key.
- **Communication overhead**: Amount and size of messages exchanged between a pair or a group of sensor nodes to generate or find a common secret key.

We can see that processing overhead is based on the hardware selecting. Considering the power consumption, especially comparing with communication overhead [35], processing overhead is not the main power consumption for WSN. Thus, it is appropriate if we omit the processing overhead of KMs in our AHP-aided evaluation proposal.

Numerical priorities, derived from the decision makers' input, are shown for each item in the hierarchy of AHP method. To make comparisons, the scale of numbers indicates that how much one element is more important than another element. The indication is based on the criterion or property with respect to which they are compared.

Then, based on the five criteria which are used to evaluate and the key management scheme comparison in an assumed network scenario by quantitative calculation, we present the framework of AHP based method for selecting the most suitable key management scheme among these schemes.

Figure 2 is the framework of our AHP-aided method. In the top of this figure, there is an assumed network scenario. Under the scenario, six criteria are listed. Under the criteria, six key management schemes which are called alternatives are listed. Each of the alternatives belongs to one category of key management schemes. In this figure, the criteria are used to select the optimum alternative for the assumed network scenario.

Our proposal consists of three steps. Figure 3 shows the procedure of our proposal. First step is establishment of a structural hierarchy. The center of this step is to construct pair-wise comparison matrix A for assumed network

TABLE II.
CLASSIFICATION OF KMS [S. A. CAMTEPE. 2008]

| | Notions | Steps |
|---|---|---|
| DWSN | Dedicated pair-wise KMs | **H.Chan *et al.* 2003** [8],D.liu *et al.* 2003 [23], B. Dutertre *et al.* 2004 [24], D. Huang *et al.* 2004 [25]. |
| | Reusable pair-wise KMs | **L.Eschenauer *et al.* 2002** [18], D. Hwang *et al.* 2004 [26], R. D. Pietro *et al.* 2003 [27], S. A. Camtepe *et al.* 2004 [28]. |
| | Group-wise KMs | **C.Blundo *et al.* 1992** [9], M. Ramkumar *et al.* 2004 [29]. |
| HWSN | Pair-wise KMs | **S. zhu *et al.* 2003** [19], G. Jolly *et al.* 2003 [30] |
| | Group-wise KMs | **M. Shehab *et al.* 2005** [20], A. Chadha *et al.* 2005 [31] . |
| | Network-wise KMs | **S. Slijepcevic *et al.* 2002** [21], A. Perrig *et al.* 2002 [32], D. Liu *et al.* 2003 [33], M.Bohge *et al.* 2003 [34] |

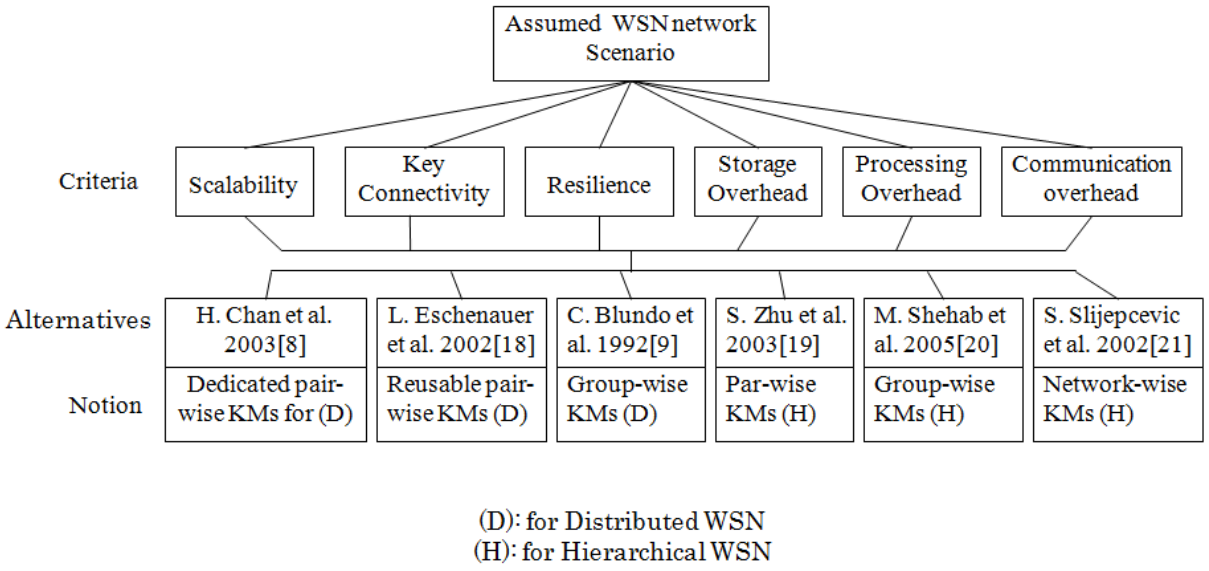



Figure 2. Framework of AHP based method for selecting a key management scheme

scenario. The importance preference of each criterion is the input. Output is the weighted vector of criteria. Second step is establishment of comparative judgments. Likewise, the center of this steps is to construct series of pair-wise comparison matrix B for each criterion. The importance value of each key management scheme is the input. Output is the weighted vectors of schemes. After finishing the first and second steps, the third step is to do consistency check, calculate values of weight coefficient for each scheme and do final decision.

We describe the first step in subsection: Establishment of a structural hierarchy. We describe the second step and the third step in subsection: Establishment of comparative judgments respectively. Specifically in subsection: Establishment of comparative judgments, we present the network scenario and its parameters.

### *A. Establishment of a structural hierarchy*

Two inputs are presented firstly. One is the importance evaluation of each criterion. Five criteria are involved here: scalability (S), key connectivity (K), resilience (R), storage overhead (M) and communication overhead (C). The other is importance evaluation of each scheme.

The importance evaluation of each criterion points out criteria establishing among the elements of the hierarchy by making a series of judgments based on pair wise comparisons of the criteria. For example, when we want to select an optimum key management scheme for army areas, choosers might say they prefer higher security and less normal nodes can be captured. Numerical priorities are derived from the decision makers' input.

In the next step, we present two types of matrix series. One is pairwise comparison matrix A for network scenario which is constructed based on each criterion's importance evaluation. The other one is pairwise comparison matrix B for criteria which is constructed based on each scheme's importance evaluation.

After constructing the two type matrix series, we can obtain two outputs. One is the weighted vector of criteria and the other one is the weighted vector of schemes. In the next section, the consistency check, calculating values of the weight coefficient for each scheme and final decision will be illustrated. In this section, we focus on explaining the matrix construction proceeds.

The formulation of AHP-based model for selecting the best key management scheme in the assumed WSN scenario is presented as shown in Algorithm 1. Based on the properties and mechanism of AHP, we provide a solution to evaluate the key management schemes in a mathematical analysis method. Our solution can be applied to select an optimum key management scheme

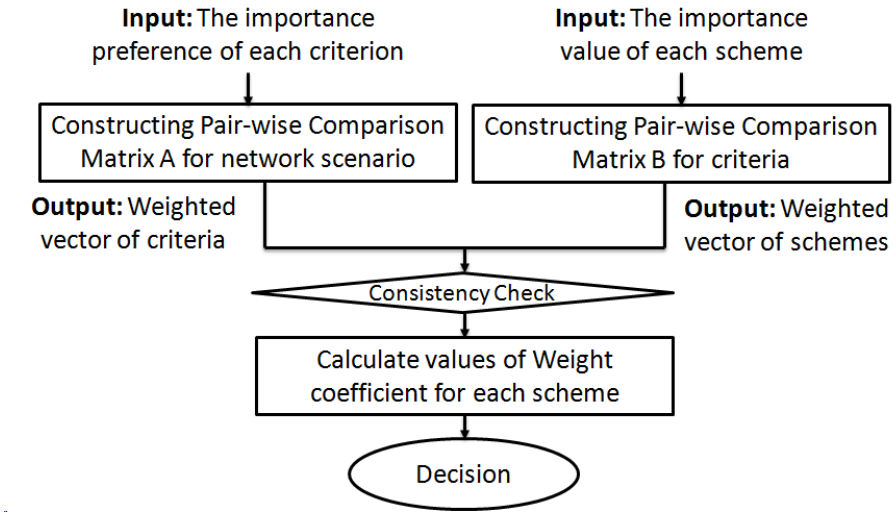



Figure 3. The inputs and outputs of our former proposal

within a particular network scenario. Basically, there are two steps for considering decision problems by AHP. Firstly, the two types of matrix series have been constructed based on the inputs.

1) One is pairwise comparison matrix $A = (a_{ij})_{6\times 6}$ for network scenario which is constructed based on each criterion's importance evaluation.
   In judgment matrix, we set $a_{ii} = 1$. Furthermore, if we set $a_{ij} = \eta$, then we set $a_{ji} = 1/\eta$.
   Here, $A = (a_{ij})_{6\times 6}$, $a_{ij} = w_i/w_j$, $w_i$ is the relative importance, $a_{ij} > 0$, $a_{ij} = 1/a_{ji}$, $a_{ii} = 1$, $i, j = 1, 2, \ldots, n$.
   The other one is pair-wise comparison matrix $B = (b_{ij})_{5\times 5}$ for the five criteria which are constructed based on each scheme's importance evaluation.
2) After constructing the two types of matrix series, we can obtain two outputs.
   One is the weighted vector of criteria $\overrightarrow{A}$ and the other is the weighted vector of schemes $\overrightarrow{B}$.
3) Then we can calculate the values of weight for each scheme $\overrightarrow{W}_\kappa = \overrightarrow{W_A} \cdot \overrightarrow{W_B}$, $\kappa = 1, \ldots, 6$.
   Finally, We can obtain the output of the decision of which scheme is the best choice $\overrightarrow{W}_{max} = \max(\overrightarrow{W}_\kappa)$.

**Algorithm 1** Our proposal

1: Input: importance values of each criterion $A = (a_{ij})_{6\times 6}$, importance values of each scheme $B = (b_{ij})_{5\times 5}$.
2: Output: the decision of the evaluation for the key management schemes $\overrightarrow{W} = (W_\kappa)_{1\times 6}$, $\kappa = 1, \ldots, 6$.
3: **while** Assumed network scenario: $\overrightarrow{A}\&\overrightarrow{B}$ **do**
4:  **while** the importance value of each criterion:$a_{ij}$ **do**
5:  Construct the pairwise comparison matrix $A$ ;
6:  Calculate the weighted vectors of the matrices $\overrightarrow{W_A}$;
7:  **end while**
8:  **while** the importance values of each key management scheme:$b_{ij}$ **do**
9:  Construct the pairwise comparisons matrix $B$;
10:  Calculate the weighted vectors of the key management scheme $\overrightarrow{W_B}$ ;
11:  **end while**
12:  **if** $\overrightarrow{W_A}\&\overrightarrow{W_B}$ **then**
13:  Calculate the values of weight for each scheme $\overrightarrow{W}_\kappa = \overrightarrow{W_A} \cdot \overrightarrow{W_B}$ ;
14:  **end if**
15:  Output the decision of which scheme is the best choice $\overrightarrow{W}_{max} = \max(\overrightarrow{W}_\kappa)$ ;
16: **end while**

### B. Establishment of comparative judgments

In this subsection, we first describe the network scenario and provide the matrix A, which is pairwise importance comparison of each criterion. Then both the parameters of the assumed network scenario and the series of matrix B are presented. The series of matrix B is pairwise importance comparison of each scheme.

We assume there is a scenario of judgment as follows: In [22], the government wants to enforce its homeland security using the WSN to aggregate the information on the borderline. In such a scenario, the perimeter surveillance is one of the most promising WSN applications. WSNs can be easily deployed permanently (e.g., public places) or on-demand (e.g., high risk events) in a very short time, with low costs and with little or no supporting communications infrastructure.

First of all, the sensor nodes must work at a low energy consumption to survive in a long time without energy supply and keep collecting and transmitting the information without breaking down. Under such a circumstance, Communication Overhead (C) becomes the most important criterion which should be considered be-

cause communication is the most energy-consuming. For instance, Mica2Dot has a 7.3MHz Atmel ATMEGA128L low-power micro-controller which runs TinyOS, 128KB of read-only program memory, 4KB of RAM, a 433MHz Chipcon CC1000 radio which provides a 19.2 Kbps data rate with an approximate indoor range of 100 meters [3].

Secondly, an attacker may capture part of sensor nodes or introduce its own malicious nodes inside the network, hence security must be taken into account in WSN design. Keys stored on a sensor node or exchanged over radio links should not reveal any information about the security of any links. Considering the Resilience (R), higher resilience means lower number of compromised links. Therefore, the resilience is an important issue in such a hostile environment.

For instance, as well as each pair-wise key coming from one node, node $S_i$ $(1 \leq i \leq N)$ stores the corresponding pair-wise keys for other N-1 sensor nodes in the WSN. Thus, each sensor $S_i$ stores a key-chain $KC_i = \{K_{i,j} | i \neq j, 1 \leq j \leq N\}$ of size $|KC_i| = N - 1$ out of $N(N-1)/2$ keys. However, not all $N-1$ keys are required to be stored in nodes' key-chain and not all $N-1$ keys are required to have a connected key graph. Thus, R is less important to C [3].)

Thirdly, Storage Overhead (M) is important because storage is necessary in order to support the store-and-forward operating principle. The data should be stored when several nodes run out of battery. And as a result, the network becomes partitioned. In this case, it is important not to lose the potentially measured data over a long period of time.

Finally, the size of the WSN is pre-determined in most of homeland security application so that the key connectivity (K) and scalability (S) is not an important issue for the government's judgments. And the location of nodes is usually fixed, which means each network scenario is assigned a scalability rank. Hence, key connectivity is more importance than scalability. Moreover, without key connectivity, the scalability will be affected due to the low communication efficiency [3].

As above, we conclude our importance is set in the increasing order of: (low) Scalability $<$ Key connectivity $<$ Storage overhead $<$ Resilience $<$ Communication overhead (high). From another aspect, we know that there are five levels in AHP. Their scale are claimed as: equal importance, weakly more important, strongly more important, very strongly more important and absolutely more important. They are described as follows:

- **Level 1** Two criteria are of **equal importance**. Storage Overhead and Resilience are of equal importance.
- **Level 2** This level which is between Level 1 and Level 3 means an intermediates value. Communication Overhead is a little more important than Storage Overhead. Resilience VS Key Connectivity: Because Storage Overhead has the same importance as resilience, storage overhead is a little more important than Key Connectivity.

TABLE III.
PAIRWISE COMPARISON JUDGMENT MATRIX OF THE FIVE CRITERIA

| | S | K | R | M | C |
|---|---|---|---|---|---|
| S | 1 | 1/3 | 1/7 | 1/5 | 1/9 |
| K | 3 | 1 | 1/2 | 1/2 | 1/3 |
| R | 7 | 2 | 1 | 1 | 1 |
| M | 5 | 2 | 1 | 1 | 1/2 |
| C | 9 | 3 | 1 | 2 | 1 |

- **Level 3** Metric $i$ is **weakly more important** than metric $j$. Key Connectivity is weakly more important than Scalability. Communication Overhead is weakly more important than Key Connectivity.
- **Level 5** Metric $i$ is **strongly more important** than metric $j$. Storage Overhead is strongly more important than Scalability.
- **Level 7** Metric $i$ is **very strongly more important** than metric $j$. Resilience is very strongly more important than Scalability.
- **Level 9** Metric $i$ is **absolutely more important** than metric $j$. Communication Overhead is absolutely more important than Scalability.

As the same as original pair-wise comparison values in AHP, the value between each two of the five levels means that it has an intermediates value. It is used to represent compromise between the levels list above. Reciprocal is suitable here for inverse comparison. The decision makers give their decision from quality aspect. They do not need the exact input. The decision makers need to give the relative importance of each two performances. Based on these relative importance items, we get the compared matrix.

Taking previous expert judgement of the five criteria into AHP-based method, we can obtain the specific levels of the above five criteria. Scalability (1) $<$ Key connectivity (3) $<$ Storage overhead (5) $<$ Resilience (7) $<$ Communication overhead (9). The most important thing in AHP is how to select items and how to give the framework of decision. First, we describe the relative importance of each of the five criteria. Then based on these the relative importance, a five level hierarchy decision process is described in Table III. As shown in Table III, we present the numerical based on the AHP pair-wise comparison table [4]. The criteria listed on the left are compared with each criterion listed on top one by one. Due to the priority of the existing alternative key management schemes, it relates to the assumed network scenario with definite comparison judgment matrix.

In judgment matrix, $a_{ii}$ is set to equal 1. Furthermore, we set $a_{ij}$ to equal $\eta$, then $a_{ji}$ equals $1/\eta$, where $A = (a_{ij})_{6\times 6}$, $a_{ij} = w_i/w_j$, $a_{ij} > 0$, $a_{ij} = 1/a_{ji}$, $a_{ii} = 1$, $i, j = 1, 2, \ldots n$. Next, we calculate the consistency ratio $CR = 0.0088 < 0.1$, which means that the pair-wise comparison judgment matrix of five criteria keeps consistency well [17].

From Table III, we normalize to obtain the relative weight or eigenvector of each rating scale. Using expert

TABLE IV.
MATRIX $B_S$: PAIR-WISE COMPARISON MATRIX OF THESE KEY MANAGEMENT SCHEMES' SCALABILITY METRIC

| | H.Chan *et al.* [8] | L.Eschenauer *et al.* [18] | C.Blundo *et al.* [9] | S. zhu *et al.* [19] | M. Shehab *et al.* [20] | Slijepcevic *et al.* [21] |
|---|---|---|---|---|---|---|
| [8] | 1 | 1 | 2 | 2 | 2 | 2 |
| [18] | 1 | 1 | 2 | 2 | 2 | 2 |
| [9] | 1/2 | 1/2 | 1 | 1 | 1 | 1 |
| [19] | 1/2 | 1/2 | 1 | 1 | 1 | 1 |
| [20] | 1/2 | 1/2 | 1 | 1 | 1 | 1 |
| [21] | 1/2 | 1/2 | 1 | 1 | 1 | 1 |

TABLE V.
RELATIVE WEIGHTS OF EACH METRIC FOR PAIR-WISE COMPARISON MATRIX OF THESE SCHEMES

| | $B_S Avg.$ | $B_K Avg.$ | $B_R Avg.$ | $B_M Avg.$ | $B_C Avg.$ |
|---|---|---|---|---|---|
| H.Chan *et al.* [8] | 0.25 | 0.25 | 0.049 | 0.273 | 0.180 |
| L.Eschenauer *et al.* [18] | 0.25 | 0.25 | 0.049 | 0.273 | 0.450 |
| C.Blundo *et al.* [9] | 0.125 | 0.125 | 0.095 | 0.265 | 0.192 |
| S. zhu *et al.* [19] | 0.125 | 0.125 | 0.269 | 0.041 | 0.069 |
| M. Shehab *et al.* [20] | 0.125 | 0.125 | 0.269 | 0.038 | 0.069 |
| Slijepcevic *et al.* [21] | 0.125 | 0.125 | 0.269 | 0.110 | 0.039 |
| | $\Sigma Avg = 1$ | $\Sigma Avg = 1$ | $\Sigma Avg = 1$ | $\Sigma Avg = 1$ | $\Sigma Avg = 1$ |

Choice software, the relative weights of Scalability (S), Key connectivity (K), Resilience (R), Storage Overhead (M) and Communication Overhead (C) are calculated, which are equal to 0.03, 0.119, 0.269, 0.218 and 0.352 respectively.

## IV. CASE STUDY

In this section, six key management schemes based on five criteria are compared. Because the importance scale of the five criteria can be various values, two case studies for further comparison are presented in this section.

We select six typical schemes: H.Chan *et al.* 2003 [8], L.Eschenauer *et al.* 2002 [18], C.Blundo *et al.* 1992 [9], S. zhu *et al.* 2003 [19], M. Shehab *et al.* 2005 [20] and S. Slijepcevic *et al.* 2002 [21] for the schemes comparison in next step. The six key management schemes are selected because each of them belongs to one kind of classification of KMs for WSN. Each of the six key management schemes has its own advantages and disadvantages. Both of their advantages and disadvantages are classified from the five criteria.

In our case study, we prove our method from three steps. The first step we assume one criteria preference and one network parameters, then we show our method step by step in details. This proceeding is presented in both subsection: Case Study 1 and subsection: Result of Case Study 1. The second step, we alter to another criteria preference as the alternation requirement from network scenario and calculate the result for analysis of the final values. We can see what is the alternation as the change of criteria preference. For the third step, we change the network size for further more explanation. This proceeding is presented in both subsection Case Study 2 and subsection: Result of Case Study 2. Finally, we compare the result of both case study 1 and case study 2 in subsection: Comparison between Case Study 1 and Case Study 2.

### A. Case Study 1

We assume the network and key's parameters as follows: In each 1 km$^2$ square unit area, for providing available WSN model, the relationship between communication distance $l$ and limiter power overhead $E$ of each sensor node is $E \propto l^n$ $(2 < n < 4)$, $n$ is effected by external influence and $n$ is usually set to 3 for calculation. Accordingly the communication radius of each node is set to 100 m [36]. Thus, the available nodes number is set to $N = 100$ for each 1 km$^2$ square unit area. Let $p$ denote the probability of sharing a key in pair-wise keys between any two nodes. Let $d = p \times (N-1)$ be the expected degree of a node.

L.Eschenauer *et al.* 2002 [18] has shown that: A key pool which has 10,000 keys means the key pool size $KP$ equals 10,000. When $KP = 10,000$, only need store 75 keys in a node's memory to ensure that the probability p can satisfy $p = 0.5$. $p$ means the probability that the nodes share a key in their key rings. If the pool size becomes ten times larger, for example, $KP = 100,000$, while the number of keys required for keeping the same probability $p = 0.5$ is only 250. The basic scheme is a key management technique which has the characters: scalable, flexible and be suitable for large networks. Thus, the key pool size $KP = 10,000$ keys, the keys number 75 keys and the probability $p = 0.5$ can be taken as an example in our case study 1.

In the key set up phase, each node $ID$ is matched with $N_p$. $N_p$ is randomly selected node identities with probability $p = 0.5$. $p = 0.5$ is always used for a qualified value for evaluation [6]. Thus we can get $N_p = 50$. At the beginning of the AHP evaluation, the matrix key distribution scheme generates a $m \times m$ key matrix for a WSN with size $N = m^2$. During key pre-distribution phase, each node is assigned a position $(i, j)$, receives both the keys in $i$-th column and the keys in $j$-th row of the key matrix as the key-chain, which totally has $2m$ keys. Here $m$ denotes the number of keys in master key list of a node and $m = \sqrt{N} = 10$. $t$ is the size of group in

TABLE VI.
QUOTED SYMBOLS IN CASE STUDY 1

| Quoted Symbols | Symbol's Name |
|---|---|
| $E$ | Power overhead |
| $l$ | Communication distance |
| $n$ | $l$'s exponent |
| $N$ | Nodes number |
| $p$ | Probability of two nodes share a key |
| $d$ | Expected degree of a node |
| $S$ | Key pool size |
| $m$ | Side of Key matrix |
| $t$ | Size of sub-group network |
| $\lambda$ | Size of adversary coalitions |
| $N_p$ | The number of each nodes stores a random set which dedicated pair-wise keys to achieve probability $p$ that two nodes share a key |

the assumed network scenario. If we assumes one group here, $t$ is set to 100. $\lambda$ is the size of adversary coalitions and equals 50.

All the quoted symbols in this section are concluded in Table VI. At the same time, the six key management schemes we have marked with black front in Table II.

For instance, the six key management schemes informed in our paper are listed in Table II. If we take their scalability into consideration, the basic numerical value of each key management scheme's scalability can be obtained from their original paper: $Value(S)$ [8] = 2, $Value(S)$ [18] = 2, $Value(S)$ [9] = 1, $Value(S)$ [19] = 1, $Value(S)$ [20] = 1, $Value(S)$ [21] = 1. Thus, we can obtain the pair-wise comparison matrix of these key management schemes' scalability value and we show the matrix $(B_S)$ as in the form of Table IV.

Accordingly, this matrix-Table IV is normalized to obtain the relative weight of eigenvector via the rating scale. As a consequence, the relative weights of key management scheme in H.Chan *et al.* 2003 [8], L.Eschenauer *et al.* 2002 [18], C.Blundo *et al.* 1992 [9], S. zhu *et al.* 2003 [19], M. Shehab *et al.* 2005 [20] and S. Slijepcevic *et al.* 2002 [21] are calculated and equal to 0.25, 0.25, 0.125, 0.125, 0.125 and 0.125, respectively. On the other hand, the consistency index CI is calculated and is equal to 0, which means that the matrix-Table IV passes consistency check. Namely, the matrix-Table IV keeps consistency well and the expert preferences are reasonable.

As the similar processing of scalability matrix $B_S$ calculation, we can go through a similar process on the other four criteria: key connectivity, resilience, storage overhead and communication overhead. Finally, the relative values of all the five criteria are calculated and summarized in Table V.

Then, as we obtain both the judgment matrix (Matrix $A$) and the matrixes for key management schemes with respect to each criteria's comparison (Matrix $B_S$, $B_K$, $B_R$, $B_M$ and $B_C$), we can calculate the final vectors of each key management scheme for the assumed WSN scenario. Recalling our overall weights, we can get a final value for each key management scheme now. The value for H.Chan *et al.* [8] is 0.175555. The solution of equations is as follows:

$$\overrightarrow{A} \cdot \overrightarrow{W_A} = \lambda \overrightarrow{W_A}$$
$$\overrightarrow{B} \cdot \overrightarrow{W_B} = \lambda \overrightarrow{W_B}$$
$$\overrightarrow{W}_{[8]} = \overrightarrow{W_A} \cdot \overrightarrow{W_B}$$

Thus, the value of H.Chan *et al.* [8] ($\overrightarrow{W}_{[8]}$ )is calculated out.

- With H.Chan *et al.* [8], $\overrightarrow{W}_{[8]} = 0.039 \times 0.25 + 0.119 \times 0.25 + 0.269 \times 0.049 + 0.218 \times 0.273 + 0.352 \times 0.180 = 0.175555$

Similarly, the value of the other five key management schemes are calculated in turns and concluded as follows:

- With L. Eschenauer *et al.* [18], $\overrightarrow{W}_{[18]} = 0.039 \times 0.25 + 0.119 \times 0.25 + 0.269 \times 0.049 + 0.218 \times 0.273 + 0.352 \times 0.450 = 0.270595$
- With C. Blundo *et al.* [9], $\overrightarrow{W}_{[9]} = 0.039 \times 0.125 + 0.119 \times 0.125 + 0.269 \times 0.095 + 0.218 \times 0.265 + 0.352 \times 0.192 = 0.170659$
- With S. Zhu *et al.* [19], $\overrightarrow{W}_{[19]} = 0.039 \times 0.125 + 0.119 \times 0.125 + 0.269 \times 0.269 + 0.218 \times 0.041 + 0.352 \times 0.069 = 0.125337$
- With M. Shehab *et al.* [20], $\overrightarrow{W}_{[20]} = 0.039 \times 0.125 + 0.119 \times 0.125 + 0.269 \times 0.269 + 0.218 \times 0.038 + 0.352 \times 0.069 = 0.124683$
- With S. Slijepcevic *et al.* [21], $\overrightarrow{W}_{[21]} = 0.039 \times 0.125 + 0.119 \times 0.125 + 0.269 \times 0.269 + 0.218 \times 0.110 + 0.352 \times 0.039 = 0.129819$

### B. Result of Case Study 1

Comparing the final value of the six schemes, we obtain the order of the six schemes' values. Their values decrease in the following order: L. Eschenauer *et al.* [18], H.Chan *et al.* [8], C. Blundo *et al.* [9], S. Slijepcevic *et al.* [21], S. Zhu *et al.* [19] and M. Shehab *et al.* [20]. Among the value of the six schemes, L. Eschenauer *et al.* scheme [18] has the biggest value and M. Shehab *et al.* scheme [20] has the least one.

The scheme with the biggest value means that it is the optimum scheme. The optimum scheme L. Eschenauer *et al.* [18] is superior to the traditional key pre-distribution schemes. Because it presents a new key management scheme for a large scale distribution sensor network. All such schemes must be extremely simple given the sensor-node computation and communication limitations. Their approach is scalable and flexible: trade-offs may occur between sensor-memory cost and connectivity, and design parameters can be adapted to fit the operational requirements of a particular environment.

The scheme with the least value means that it is not a suitable scheme for the assumed WSN scenario. We know that scheme M. Shehab *et al.* [20] is suitable for limited computation and energy capability sensor network. This proposed key generation algorithm is based on low cost hashing functions that enable the efficient key generation. Its key distribution protocol is also energy efficient. Thus, this scheme satisfies with the energy limitation problem

TABLE VIII.
ANOTHER RELATIVE WEIGHTS OF EACH CRITERION FOR PAIR-WISE COMPARISON MATRIX OF THE SIX SCHEMES

| | $B_S Avg.$ | $B_K Avg.$ | $B_R Avg.$ | $B_M Avg.$ | $B_C Avg.$ |
|---|---|---|---|---|---|
| H.Chan *et al.* [8] | 0.25 | 0.125 | 0.049 | 0.041 | 0.069 |
| L.Eschenauer *et al.* [18] | 0.25 | 0.25 | 0.269 | 0.273 | 0.450 |
| C.Blundo *et al.* [9] | 0.125 | 0.125 | 0.095 | 0.265 | 0.192 |
| S. zhu *et al.* [19] | 0.125 | 0.25 | 0.269 | 0.273 | 0.180 |
| M. Shehab *et al.* [20] | 0.125 | 0.125 | 0.269 | 0.038 | 0.069 |
| Slijepcevic *et al.* [21] | 0.125 | 0.125 | 0.049 | 0.110 | 0.039 |
| | $\Sigma Avg = 1$ | $\Sigma Avg = 1$ | $\Sigma Avg = 1$ | $\Sigma Avg = 1$ | $\Sigma Avg = 1$ |

TABLE VII.
FURTHER ONE MORE CASE ABOUT PAIRWISE COMPARISON JUDGMENT MATRIX OF THE FIVE CRITERIA

| | S | K | R | M | C |
|---|---|---|---|---|---|
| S | 1 | 1/3 | 1/7 | 1/5 | 1 |
| K | 3 | 1 | 1/2 | 1/2 | 3 |
| R | 7 | 2 | 1 | 1 | 7 |
| M | 5 | 2 | 1 | 1 | 5 |
| C | 1 | 1/3 | 1/7 | 1/5 | 1 |

of wireless sensor network. The trade-off between energy and security is the biggest problem in wireless sensor network, so it cannot satisfy the requirement in our assumed network scenario.

### C. Case Study 2

Both subsection A and subsection B in section 4 are the results of our case study 1. Case study 1 considers the preference setting for the evaluation criteria and the assumed WSN network scenario. For more clear explanation, further work with more discussion can be done well from two aspects: the first one is to change the criteria preferences even to do all the permutation of the criteria. Under other criteria preferences, we can see what is changed from the final optimum scheme result. Under the all permutation of criteria, we can obtain the different result of optimum scheme according to different preference.

The other one is to change the parameters of WSN network scenario, such as the network size and the key pool size. Accordingly, we can do analysis on the final scores of these key management schemes in wireless sensor network which can help us come to be familiar with these schemes and make the decision for selecting optimum scheme easily.

First, we analyze one more case on the pairwise comparison judgement matrix of the criteria. If the WSN is for civil use which can provide enough energy and can keep the advantage of WSN, we can obtain the preference of criteria as follows: (low) Scalability (1) = Communication overhead (1) < Key connectivity (3) < Storage overhead (5) < Resilience (7) (high). The judgement matrix of criteria preference is shown in Table VII accordingly.

Keeping the same network parameters of WSN network scenario, we can calculate the final value under the one more case study which is according to one more criteria preference. The final values of the six typical schemes are sorted in decreasing order: L. Eschenauer *et al.* [18] $= 0.1935$, H.Chan *et al.* [8] $= 0.17757$, S. Slijepcevic *et al.* [21] $= 0.1679$, C. Blundo *et al.* [9] $= 0.1636$, S. Zhu *et al.* [19] $= 0.1468$ and M. Shehab *et al.* [20] $= 0.1459$.

Both the best scheme and the worst scheme in this result is the same as case study 1. However the order of the six values has been changed. Scheme C. Blundo et al. [9] and scheme S. Zhu et al. [19] change their order to each other. Thus, we can see the affect from the changing of the criteria preference .

Then, we analyze the affection from WSN network parameters setting. Previous network size $N = 100$. If more nodes have been added in and we also want to keep the same probability $p = 0.5$ for the probability of that two nodes share a key, it is an interesting problem on the optimum scheme alteration for the current network scenario. As shown in L. Eschenauer *et al.* [18], it is worth mentioning that only $k = 75$ keys are needed for probability $p = 0.5$ that any two nodes can share a key in their key ring as $KP = 10,000$. Thus, we assume the new network scenario as follows: Network size $N = 1000$. As we know, there is the expected degree of a node $d = p \times (N-1) = 500$. Accordingly we get the successfully connected nodes number $Np = 500$, the size of adversary coalitions $\lambda = 50$, $t = 200$ which means five grids here. In matrix key distribution scheme, $m = 10$ as $m = \sqrt{N}$. We let all the schemes keep the same key pool $KP = 10,000$ as given in scheme L. Eschenauer *et al.* [18]. Then, the alternation has been showed in Table VIII.

### D. Result of Case Study 2

Under the criteria preference setting in Table III and the WSN network parameters setting in Table VIII, we apply our AHP-aid method to calculate the combining of both Table III and Table VIII. We can calculate the final value and come to the conclusion that scheme S. Zhu *et al.* [19] takes advantage of the other schemes. Here the optimum scheme is S. zhu *et al.* [19] which is different from previous optimum scheme L. Eschenauer *et al.* [18]. This is the effect of network nodes number alternation from $N = 100$ to $N = 1000$. Scheme S. zhu *et al.* [19] is an efficient security key management scheme for larger scale sensor network. It can reduce the communication overhead between each communication unit. Thus, the more larger network size the more obvious the advantages are. This can be shown to be the same as the original

TABLE IX.
PARAMETERS COMPARISON BETWEEN CASE STUDY 1 AND CASE STUDY 2

| | $n$ | $N$ | $p$ | $d$ | $KP$ | $k$ | $m$ | $t$ | $\lambda$ | $Np$ | Optimum scheme |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Case Study 1 | 3 | 100 | 0.5 | 50 | 10,000 | 75 | 10 | 100 | 50 | 50 | L. Eschenauer *et al.* [18] |
| Case Study 2 | 3 | 1000 | 0.5 | 500 | 100,000 | 250 | 100 | 200 | 50 | 500 | S. Zhu *et al.* [19] |

paper assumption [19]. It consistent to our new network requirement.

### *E. Comparison between Case Study 1 and Case Study 2*

In the above four subsections, we describe two groups network scenario. We called them Network scenario $Net1$ and Network scenario $Net2$, respectively. Firstly, we conclude both of the network scenario. Then Table IX is used to show clearly the parameters' value of the two network scenario. Lastly, we explain the different parameters' value between the two network scenarios.

**Network Scenario $Net1$:** Recall from Section IV.A, the parameters of network scenario and key management scheme have been presented. The parameters can be concluded in the following:

$p = 0.5$, $N = 100$, $Np = 50$, $d = 50$, $KP = 10,000$, $k = 75$, $m = 10$, $\lambda = 50$, $t = 100$.

**Network Scenario $Net2$:** L. Eschenauer *et al.* [18] scheme infers that if the pool size is ten times larger, for example, $KP = 100,000$, then the number of keys required is still only 250 for keeping the value $p = 0.5$ which is the same as in the first group network scenario. The basic scheme is a key management technique that is scalable, flexible and can be used for large networks. Then we can present another WSN scenario: we enlarge the key pool size and the network nodes number.

We refer to the key pool size from scheme L. Eschenauer *et al.* [18]: $KP = 100,000$ keys, only $k = 250$ keys is needed for probability $p = 0.5$ such that any two nodes can share a key in their key ring. The available nodes number is enlarged to $N = 1,000$. Because of the same probability $p = 0.5$ and assumed $N = 1,000$, we can obtain that $Np = 500$, $d = 500$, $t = 200$ (five grids for the hierarchical structure), $m = 100$.

Table IX concludes the parameters used in both case study 1 and case study 2. The value of $n$, $p$ and $\lambda$ are the same in both case studies as shown in Table IX. Keeping the value of $p$ as the same as precondition, the other parameters in case study 2 change to different values [18]. In both case studies, the basic changed values are the network size $N$ and the key pool size $KP$. The key pool size $KP$ is changed from 10,000 to 100,000 and the nodes number $N$ change from 100 to 1000. $k$ is changed as $KP$ changed. $Np$, $d$ and $t$ are changed. Because the changing of the size of network grids $t$ causes the number of network grids to change, $m$ is changed as the network grids is changed.

All the changed values above cause the two case studies to perform different final decision. Case study 1 selects L. Eschenauer *et al.* [18] scheme as the optimum scheme. Meanwhile, case study 2 selects S. Zhu *et al.* [19] scheme as the optimum scheme. From the two case studies, the relationship between our method decision and the changing of the parameters are drawn. Obviously, the quantitative decision from our method brings into correspondence with the original case situations.

## V. CONCLUSION

From the analysis, we can see all the key management schemes have their own shortcomings. For this reason, it is a very critical issue to select trustworthy and suitable key management scheme according to assumed scenario requests. Such evaluation analysis can help to provide some valuable information for designing the key management in WSN.

In this paper, we present a quantitative evaluation system for key management scheme which is based on the six aspects: scalability, key connectivity, resilience, storage overhead, processing overhead and communication overhead. We analyze it and show that this system can be used to select suitable key management scheme under assumed wireless sensor network scenario requirements. Furthermore, we show six typical key management schemes from the six classified aspects. Under assumed network scenarios, we can obtain the value order of the six schemes. Importantly, we obtain the best scheme and the worst one via their final calculated values.

Formalized decision should be made where there are a limited number of schemes choices. However each scheme has a number of attributes and it is difficult to formalize some of those attributes. Obviously, AHP-aided method can prevent subjective judgment errors and increase the likelihood that the results are reliable. AHP-aided method provides useful insight into the trade-offs embedded in a decision making problem.

## ACKNOWLEDGMENT

The authors would like to thank lab colleagues for carefully reading the paper and providing detailed suggestions on paper structure and comments on the English writing. Also many thanks to the anonymous reviewers in the JNW editorial committee for their insightful comments on this work.

## REFERENCES

[1] J. Lopez, J. Y. Zhou, Overview of wireless sensor network security, *Wireless Sensor Network Security, (J. Lopez and J. Y. Zhou Eds)*, IOS Press, 2008.

[2] Y. Jeong, S. Lee, Hybrid Key Establishment Protocol Based on ECC for Wireless Sensor Network, In proceeding of *the 4th international conference on Ubiquitous Intelligence and Computing*, Volume 4611, pages 1233-1242, Hong Kong, China, July, 2007.

[3] S. A. Camtepe, B. Yener, Key management in wireless sensor network, *Wireless Sensor Network Security, (J. Lopez and J. Y. Zhou Eds)*, IOS Press, 2008.
[4] T. L. Saaty, *The Analytic Hierarchy Process*, McGraw-Hill, New York, 1980.
[5] J. Hwang, I. Syamsuddin, Information Security Policy Decision Making: An Analytic Hierarchy Process Approach, In proceeding of *2009 Third Asia International Conference on Modelling and Simulation*, pages 158-163, May, 2009
[6] Y. Xiao, V. K. Rayi, B. Sun, X. Du, F. Hu, M. Galloway, A survey of key management schemes in wireless sensor networks, *Computer Communications, special issue on security on wireless ad hoc and sensor network*, Volumn 30, pages 2314-2341, September, 2007.
[7] C. J. Jia, Research on security of wireless sensor network, *PhD thesis, ZheJiang University*, July, 2008.
[8] H.Chan, A. Perrig, D. Song, Random key pre-distribution schemes for sensor networks, In proceeding of *IEEE Symposium on Security and Privacy*, pages 197-213, May, 2003.
[9] C. Blundo, A. D. Santis, A. Herzberg, S. Kutten, U. Vaccaro, and M. Yung, Perfectly-secure key distribution for dynamic conferences, In proceeding of *Advances in cryptology (CRYPTO 92)*, pages 471-486, August, 1992.
[10] D. Chakrabarti, S. Maitra, B. Roy, A key pre-distribution scheme for wireless sensor networks: Merging blocks in combinatorial design, In *ISC 2005 Proceedings, Lecture notes in computer science*, ISSN 0302-9743, Volumn 3650. Springer, pages 89-103, 2005.
[11] T. Ito, H. Ohta, N. Matsuda, T. Yoneda, A key pre-distribution scheme for secure sensor network using probability density function of node deployment, In proceedings of *the 3rd ACM workshop on Security of ad hoc and sensor networks*, pages 69-75, November, 2005.
[12] L. Gong, D. J. Wheeler, A matrix key distribution schemes, In *Journal of Cryptology*, Volumn 2-1. pages 51-59, 1990.
[13] B. Dutertre, S. Cheung, J. Levy, Lightweight key management in wireless sensor networks by leveraging initial trust, *Technical Report SRI-SDL-04-02, System Design laboratory, SRI International*, April, 2004.
[14] C. Fei. Pair-wise Key Management in Wireless Sensor Network, In *Journal of Computer Simulation*, Volumn 22-5, 2005.
[15] H. Soussi, M. Hussain, H. Afifi, D. Seret. IKEv1 and IKEv2: A Quantitative Analyses, In proceeding of *International Conference on Information Security(ICIS WEC'05)*, Istanbul, Turquie, 24-26 June, 2005
[16] E. H. Forman. Decision by Objective, *http://mdm.gwu.edu/Forman/DBO.pdf*.
[17] AHP (Analytic Hierarchy Process) Calculation software by CGI, *http://www.isc.senshu-u.ac.jp/ thc0456/EAHP/AHPweb.html*.
[18] L. Eschenauer, V. D. Gligor, A key-management scheme for distributed sensor networks, In proceeding of *ACM Conference of Computer and Communication Security*, pages 41-47, November, 2002.
[19] S. Zhu, S. Setia, S. Jajodia, Leap:Efficient security mechanisms for large-scale distirbuted sensor networks, In proceeding of *ACM Conference of Computer and Communication Security*, pages 62-72, October, 2003.
[20] M.Shehab, E.Bertino, A.Ghafoor, Efficient hierarchical key generation and key diffusion for distribution for distributed sensor networks, In proceeding of *IEEE International Conference of Sensor and Ad Hoc Communication and Network*, pages 76-84, September, 2005.
[21] S.Slijepcevic, M.Potkonjak, V.Tsiatsis, S.Zimbeck, M.B.Srivastava, On communication security in wireless ad-hoc sensor network, In proceeding of *IEEE WETICE*, pages 139-144, November, 2002.
[22] A. Casaca, D. Westhoff, Scenario Definition and Initial Threat Analysis, In *report of UbiSec and Sens*, Deliverable D0.1, June, 2006
[23] D.Liu, P.Ning, Location-based pairwise key establishments for static sensor networks, In proceeding of *2003 ACM Workshop on Security in Ad Hoc and Sensor Networks (SASN'03)*, pages 72-82, October, 2003
[24] B.Dutertre, S.Cheung, J.Levy, Lightweight Key Management in Wireless Sensor Networks by Leveraging Initial Trust, In *Technical Report SRI-SDL-04-02, System Design Laboratory, SRI International*, April, 2004.
[25] D. Huang, M. Mehta, D. Medhi, L. Harn, Location-aware key management scheme for wireless sensor networks, In proceeding of *ACM Workshop on Security of Ad Hoc and Sensor Network*, pages 29-42, October, 2004
[26] D. Hwang, B. Lai, I. Verbauwhede, Energy-memory-security tradoffs in distributed sensor networks, In proceeding of *ADHOC-NOW*, LNCS, pages 70-81, July, 2004
[27] R. D. Pietro, L. V. Mancini, A. Mei, Random key assignment for secure wireless sensor networks, In proceeding of *ACM workshop on Security of ad hoc and sensor networks*, October, 2003.
[28] S. A. Camtepe, B. Yener, Combinatorial design of key distribution mechanisms for wireless sensor networks, In *9th European Symposium on Research Computer Security*, pages 293-308, September, 2004
[29] M. Ramkumar, N. Memon, An efficient random key pre-distribution scheme, In proceeding of *Global Telecommunications Conference (GLOBECOM '04)*, 29 Nov.-3 Dec. 2004, 2004.
[30] G. Jolly, M. C. Kuscu, P. Kokate, M. Younis, A low-energy key management protocol for wireless sensor networks, In proceeding of *Eighth IEEE International Symposium on Computers and Communication*, pages 335-340, June 30-July 3, 2003
[31] A. Chadha, Y. Liu, S. K. Das, Group key distribution via local collaboration in wireless sensor networks, In proceeding of *IEEE International Conference of Sensor and Ad Hoc Communication Network*, pages 46-54, February, 2006
[32] A. Perrig, R. Szewczyk, V. Wen, D. Culler, J. D. Tygar, Spins: Security protocols for sensor networks, In *Wireless Networks Journal*, Volumn 8-5, 2002
[33] D. Liu, P. Ning, Efficient distribution of key chain commitments for broadcast authentication in distributed sensor networks, In *Network and Distributed System Security Symposium*, February, 2003
[34] M. Bohge, W. Trappe, An authentication framework for hierarchical ad hoc sensor networks, In *ACM WiSe*, pages 79-87, September, 2003
[35] M. Gabriela, C. Torres, Enegry Consumption in wireless sensor network using GSP, *Thesis for master degree, University of Pittsburgh*, 2006
[36] G. B. Zhou, Z. C. Zhu, G. Z. Chen and N. N. Hu, Energy-Efficient Chain-Type Wireless Sensor Network for Gas Monitoring, In proceeding of *The Second International Conference on Information and Computing Science*, pages 125-128, May, 2009

**Ruan Na** was born in AnQing, AnHui, China on 1986. She is currently a Ph.D. candidate at Kyushu University, Japan. She received her MS and BS degrees in communi-

cation from China University of Mining and Technology, China, in 2010 and 2007, respectively. Her research interests include security, wireless sensor network and communication systems.

**Yizhi Ren** received the B.S. degree in computer science from Anhui Normal University in 2004, and the M.S. degree and doctorate in computer software and theory from School of Software, Dalian University of Technology in 2006 and 2010, respectively. From 2008 to 2010, he was as a Joint Training PhD student in Kyushu University in Japan, supported by China Government Scholarship. Now, he worked for Software Engineering School of Hangzhou Dianzi University as an assistant professor. His main research interests include network security, social computing.

**Yoshiaki Hori** received B.E., M.E, and D.E. degrees on Computer Engineering from Kyushu Institute of Technology, Iizuka, Japan in 1992, 1994, and 2002 respectively. From 1994 to 2003, he was Research Associate at the Common Technical Courses, Kyushu Institute of Design. From 2003 to 2004, he was Research Associate at the Department of Art and Information Design, Kyushu University. From 2004, he was Associate Professor at the Department of Computer Science and Communication Engineering, Kyushu University. Since 2009, he has been Associate Professor of Department of Informatics, Kyushu University. His research interests include network security, network architecture, and performance evaluation of network protocols on various networks. He is a member of IEEE, ACM, and IPSJ.

**Kouichi Sakurai** received the B.S. degree in mathematics from the Faculty of Science, Kyushu University and the M.S. degree in applied science from the Faculty of Engineering, Kyushu University in 1986 and 1988 respectively. He had been engaged in the research and development on cryptography and information security at the Computer and Information Systems Laboratory at Mitsubishi Electric Corporation from 1988 to 1994. He received D.E. degree from the Faculty of Engineering, Kyushu University in 1993. Since 1994 he has been working for the Department of Computer Science of Kyushu University as Associate Professor, and now he is Full Professor from 2002. His current research interests are in cryptography and information security. Dr. Sakurai is a member of the Information Processing Society of Japan, the Mathematical Society of Japan, ACM and the International Association for Cryptology Research.

# Interference Analysis of TD-SCDMA System and CDMA2000 System

Chen Hao[1,2]
1 School of Electronic and Information Engineering, Tianjin University, Tianjin 300072, China
2 Computer Science and Information Engineering College, Tianjin University of Science & Technology, Tianjin 300222 ,China
Email: haochen111@yahoo.com.cn

Yang Tong
Tianjin Mobile Communications Co., Ltd., Tianjin 300052, China
Email: yangtong@tj.chinamobile.com

Teng Jian-fu
School of Electronic and Information Engineering, Tianjin University, Tianjin 300072, China
Email: jfteng@tju.edu.cn

He Hong
Tianjin Key Laboratory for Control Theory and Application in Complicated Systems,
Tianjin University of Technology, Tianjin 300384, China
Email: heho604300@126.com

***Abstract*—In this paper, the feasibility of co-channel coexistence of Time Division- Synchronous Code Division Multiple Access （ TD-SCDMA ） and Code Division Multiple Access 2000（CDMA2000）systems operating in a macro cell environment is investigated. The deterministic analysis and simulation method are used to evaluate the performance compromising of both systems. Based on a more efficient calculation scheme, a novel deterministic equation is proposed and used to provide a better interpretation of the relationship between aggressor and victim in the interference system. The evaluation and simulation results show consistency with the corresponding experiment results. Furthermore, the interference characteristics of Omni-antenna and smart antenna in TD-SCDMA and CDMA2000 are compared, providing an important guideline to reduce the interference of two systems.**

***Index Terms*—TD-SCDMA; CDMA2000; smart antenna**

## I. INTRODUCTION

Time Division-Synchronous Code Division Multiple Access and Code Division Multiple Access 2000 operated by China Mobile and China Telecom are the dominating mobile protocols[1][2], which are important to Chinese telecommunication industries. Frequency bands of TD-SCDMA are assigned to 1 880-1 920 MHz and 2 010-2 025 MHz, while CDMA2000 frequency bands are 1 920-1 980 MHz and 2 110-2 170 MHz. At 1 920 MHz, the two systems operate in the adjacent channels with a 0.825MHz protection band and their interference is inevitable. Therefore, it is important to evaluate the interference of two systems in the adjacent channels. Specific interferences[1][2][3]include: the CDMA2000 mobile stations interferes TD-SCDMA base and mobile stations; at the same time, TD-SCDMA base stations and mobile stations conflict with CDMA2000 base stations.

The specific form of interference is shown in Fig.1

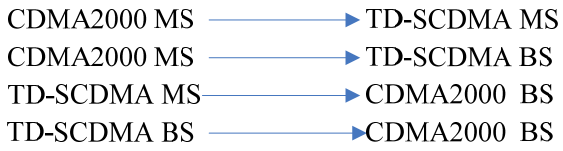



Fig.1 Specific forms of Interference

The interference study is based on the coexistence of both modes[4]. Such coexistence can increase the capacity and provide great flexibility for operators. Some previous studies examined the coexistence of UTRA-FDD and UTRA-TDD systems in adjacent channels by hierarchically overlapping coexistence of micro TDD cell and UTRA-FDD cells sharing the same frequency allocation[5] [6][7]. However, the coexistence of TDD and FDD systems operating in a macro cell environment and sharing the same frequency band has not been investigated yet. The focus of this paper is to estimate the capacity of such a coexistence of CDMA2000 and TD-SCDMA network and to evaluate its interference. The rest of the paper is organized as follows. Deterministic analysis[2] is discussed in section II. Simulation methodology[3] and scenario are introduced in section III. Simulation results and analysis are given in section IV. Section V concludes the paper.


doi:10.4304/jnw.7.1.101-109

## II DETERMINISTIC ANALYSIS

### *A. ACIR Approach[1]*

Deterministic analysis is usually used to analyze the worse interference cases. Using this method, we will use minimum Coupling Loss (MCL) data as the result. Based on these data, the interference analysis between the third-generation communication systems can be obtained.

Based on link budget, deterministic analysis is used to evaluate the effect of the interfering system on a single link of the victim system. Since the traditional deterministic equation may easily mislead the beginners, a novel deterministic equation is developed here, which enable us to easily calculate the relationship between aggressor and victim in interference system with a clearer understanding than the previous equation.

We now introduce some methodologies before describing the detailed analysis. The above method mainly calculates the ACIR result, which is obtained by using ACLR and ACS values (as listed in Tab. 1). ACIR can be evaluated by:

$$ACIR = \frac{1}{\frac{1}{ACLR} + \frac{1}{ACS}} \qquad (1)$$

Adjacent Channel Leakage power Ratio (ACLR) [2]: The ratio of the transmitted power to the power measured after a receiver filter in the adjacent RF channel.

Adjacent Channel Selectivity (ACS) [2]: Adjacent Channel Selectivity is a measure of a receiver's ability to receive a signal at its assigned channel frequency with the presence of a modulated signal in the adjacent channel. ACS is the ratio of the receiver filter attenuation on the assigned channel frequency to the receiver filter attenuation on the adjacent channel frequency.

ACIR stands for Adjacent Channel Interference Ratio which is defined as the ratio of the transmitted power to the power measured after a receiver filter centered on the adjacent channels. From Eq.(1) and Tab.1, the values of ACIR can be calculated as listed in Tab.2

Tab.1 ACLR and ACS requirements for TD-SCDMA and CDMA2000 based on 3GPP protocol

| Frequency Offset ∆ f/MHz | CDMA2000 MS ACLR | CDMA2000 BS ACS | TD-SCDMA MS ACLR | TD-SCDMA BS ACLR | TD-SCDMA MS ACS | TD-SCDMA BS ACS |
|---|---|---|---|---|---|---|
| | 30.5dB | 55dB | 38.5dB | 45.8dB | 38dB | 52.5dB |

Tab.2 ACIR values of TD-SCDMA and CDMA2000

| Interferer | CDMA2000 MS | CDMA2000 MS | TD-SCDMA MS | TD-SCDMA BS |
|---|---|---|---|---|
| Victim | TD-SCDMA MS | TD-SCDMA BS | CDMA2000 BS | CDMA2000 BS |
| ACIR (dB) | 29.8 | 30.5 | 38.5 | 45.3 |

Tab.3, we give the relevant data of co-existence and co-sited requirement in 3GPP protocols. Tab.3 is obtained based on Tab.1 and Tab.2

Tab.3 Co-existence and Co-sited requirement based on 3GPP protocol

| | CDMA2000 BS ACS /2.25MHz | TD-SCDMA BS ACLR | ACIR |
|---|---|---|---|
| Co-existence | 55dB | 70dB | 54.9dB |
| Co-sited | 55dB | 110dB | 55dB |

As seen in Tab.3, since the ACIR requirement of co-sited (or co-existence) is larger than that of the ACIR calculated between TD-SCDMA BS and CDMA2000 BS (54.9 dB is larger than 45.3 dB), the conventional layout can not meet the requirements of co-existence or co-station.

### *B MCL Method*

Traditional deterministic equation usually focuses on adjacent frequency, so the equation given in (2) may easily mislead the beginners.

$$MCL = P_{TX} - ACIR - I_r \qquad (2)$$

where

$P_{TX}$ : transmitter power of the base station(or mobile station)

$I_r$ : maximum acceptable interference level at the receiver

$MCL$ : minimum coupling loss

In fact, $P_{TX}$ actually represents interference dimension，while MCL denotes overcome interference dimension .Therefore, a new equation with clear physical meaning will be developed.

Based on the combination of interference between systems, the overall interference can be calculated, and then the MCL is computed. Adjacent frequency interference, spurious interference, intermodulation interference is existing in the victim bandwidth. They deteriorate the quality of communication. Blocking is out of the receiving bandwidth and it generates interference system receiver saturation and obstacles to the communication.

Normally, there is a great difference between interference strength. The total interference power is dominated by the strongest interference power. Even in the most extreme cases, that each interference power value is almost equal, then total interference power will increase 6.02 dB, but still consistent with (3) and (4) and in the margin of allowance.

$$I_{total} = 10\log(10^{\frac{I_{ACI}}{10}} + 10^{\frac{I_{SEI}}{10}} + 10^{\frac{I_{IMD}}{10}} + 10^{\frac{I_{Blocking}}{10}}) \quad (3)$$

$$CL \geq I_{total} \quad (4)$$

$I_{total}$ : total of interference power

$I_{ACI}$ : adjacent frequency interference power

$I_{SEI}$ : the spurious emissions power

$I_{IMD}$ : intermodulation power

$CL$ : coupling loss

When the ACI exists, the equation is simplified

$$I_{total} = I_{ACI} \quad (5)$$

When $CL = I_{total}$ , the CL is MCL.

Our study will only focus on Eq.(3),(4)and(5) to calculate MCL. The new equations are deduced to show in Eq. (6)

$$I_{ACI} = P_{TX} - ACIR - I_r \quad (6)$$

$CL \geq I_{ACI}$ , if and only if $CL = I_{ACI}$ , CL is equal MCL.

The physical meaning of the Eq. (6) is more clearer than that of Eq.(2)

The relationship among $I_r$ and NF, ROT , KTW is listed in Eq. (7)

$$I_r = 10 \times \log(10^{\frac{KTW+NF+ROT}{10}} - 10^{\frac{KTW+NF}{10}}) \quad (7)$$

Where NF：noise floor ROT: raise over thermal KTW: thermal noise power

Based on 3GPP protocol, PTX and bandwidth conversion values can be found. ACIR have been calculated in the table 2. The other parameters can be obtained from the protocol of 3GPP 25.942. MCL can be calculated from Eq. (6) and (7), and the values are summarized in the Tab. 5

Tab.4 Parameters of TD-SCDMA System and CDMA2000 System

| Item | Unit | CDMA2000 MS interfering TD-SCDMA MS | TD-SCDMA BS interfering CDMA2000 BS | CDMA2000 MS interfering TD-SCDMA MS | TD-SCDMA MS interfering CDMA2000 BS |
|---|---|---|---|---|---|
| $P_{TX}$ | dBm | 21 | 36 | 21 | 30 |
| ACIR | dB | 29.8 | 45.3 | 30.5 | 38.5 |
| bandwidth conversion | dB | 1.1 | -1.1 | 1.1 | -1.1 |
| KTW | dBm | -113 | -113 | -113 | -113 |
| NF | dB | 7 | 7 | 7 | 7 |
| ROT | dB | 3 | 3 | 3 | 3 |

Tab.5 MCL of TD-SCDMA System and CDMA2000 System

| Item | Unit | CDMA2000 MS interfering TD-SCDMA MS | TD-SCDMA BS interfering CDMA2000 BS | CDMA2000 MS interfering TD-SCDMA MS | TD-SCDMA MS interfering CDMA2000 BS |
|---|---|---|---|---|---|
| MCL | dB | 97.22 | 96.72 | 96.52 | 97.52 |

The propagation environments are considered in the Macro-cellular model.

The path loss of the propagation[1][8] can be evaluated in Eq. (8)

$$L = 128.1 + 37.6\log_{10} R \quad (8)$$

R is the base station- UE separation in kilometers

L: path loss ΔG：Antenna gain

LogF：Log Normal fade margin

Pathloss_Maro＝L＋LogF (9)

Tab. 6 Parameters of propagation

| Item | Unit | CDMA2000 MS interfering TD-SCDMA MS | TD-SCDMA BS interfering CDMA2000 BS | CDMA2000 MS interfering TD-SCDMA MS | TD-SCDMA MS interfering CDMA2000 BS |
|---|---|---|---|---|---|
| ∆G | dBi | -6.0 | 16.0 | 11.0 | 11.0 |
| L=∆G+MCL | dB | 91.22 | 112.72 | 107.52 | 108.52 |
| LogF | dB | 10 | 10.0 | 10.0 | 10.0 |
| Pathloss_Macro | dB | 101.22 | 112.72 | 117.52 | 118.52 |

R can be calculated in the Tab.7 ,according to Eq. (8) and (9)

Tab.7 R of Macro-cellular environment

| Item | Unit | CDMA2000 MS interfering TD-SCDMA MS | TD-SCDMA BS interfering CDMA2000 BS | CDMA2000 MS interfering TD-SCDMA MS | TD-SCDMA MS interfering CDMA2000 BS |
|---|---|---|---|---|---|
| R | km | 0.193 | 0.389 | 0.523 | 0.556 |

As deterministic analysis is appropriate for the worst case, the probability of the worst case is low[5]. So MS to MS, MS to BS and BS to MS can use conventional layout. If there is interference, adjusting the position of BS and narrowing the distance and service radius of BS can reduce or avoid the interference. However, BS to BS interference requires high, co-existence and co-sited scenario that is difficult to satisfy. Thus, it needs to adopt the following improvement.

1: The level of technology needs improvement and ACIR of equipments needs upgraded, especially the ACS of the BS of CDMA2000.

2: It is shown that adjacent channel interference is caused by the deterioration of the sensitivity volume of CDMA2000 BS. Narrowing CDMA2000 BS base station service radius and spacing improves the uplink signal of the MS power to resist the TD-SCDMA BS strong adjacent channel interference.

In summary, when CDMA2000 system and TD-SCDMA system are deployed in adjacent frequencies, ACI is the important and major part of the total interference. SEI, IMD and Blocking are more relaxing than ACLR and ACS requirements relative to the adjacent frequency interference. So just considering these two systems, SCI can satisfy the coexistence requirement, and SEI, IMD and Blocking requirements can be satisfied.

## III. Simulation Scenario

### *A Macro-cell Scenario [2]*

Since for the macro scenario a hexagonal cell structure is assumed, Monte-Carlo method has been chosen for evaluation. Each Monte-Carlo (MC) simulation cycle starts with the positioning of the receiver station (disturbed system) by means of an appropriate distribution equation for the user path. The interfering (mobile) stations are assumed to be uniformly distributed. The density of interferers is taken as parameter. To start up we assume that only the closest user of the co-existing interfering system is substance of the main interference power. At each MC cycle the pathloss between the disturbed receiver and the next interfering station as well as the pathloss for the communication links are determined according to the pathloss formula given in the next clause. Depending on the use of power control the received signal level C at the receiver station in the disturbed system is calculated. Finally the interference power I is computed taking into account the transmit spectrum mask and the receiver filter. C/I is then substance to the statistical evaluation giving the CDF.

### *B Pathloss formula[2][8]*

The pathloss calculation for the Macro Vehicular Environment Deployment Model is implemented to simulate the MS↔ BS case (10 dB log-normal standard deviation). Both 3000 m and 577 m cell-radii are considered. The simulation support an Omni directional pattern antenna. Considering TDD the mobile to mobile interference requires a model valid for transmitter and receiver antennas having the same height. In order to cover this case the outdoor macro model is based on path loss formula from H.Xia considering that the height of the BS antenna is below the average building height.

### *C Evaluation of FDD/TDD interference yielding relative capacity loss*

#### *C.1 Definition of system capacity[9]*

The capacity of the system is defined as the mean number of mobile stations per cell that can be active at a time while the probability that the C/I falls below a given threshold is below 5%[10]. All mobiles use the same service.

#### *C.2 Calculation of capacity*

A relative capacity loss is calculated as:

$$C = 1 - \frac{N_{multi}}{N_{\sin gle}} \qquad (10)$$

where $N_{\sin gle}$ is the maximum mean number of mobiles per cell that can be activated at a time in the single operator case, i.e. without adjacent channel interference. is the maximum mean number of mobiles

per cell that can be activated at a time in the multi operator case, i.e. with adjacent channel interference originating in one interfering system in an adjacent transmit band.

## IV. Simulation Parameters, Results and Analysis

### *A. Simulation parameters*

TD-SCDMA system and CDMA2000 system simulation parameters are listed in Tab. 8 and Tab.9

Tab. 8 TD-SCDMA system simulation parameters in Macro environment

| Parameter | UL | DL |
|---|---|---|
| Simulation times | Snapshot（>=800） | Snapshot (>=800) |
| MCL （dB） | BS-UE : 70 | BS-UE : 70 |
| | UE-UE :40 | UE-UE :40 |
| | BS-BS :45 | BS-BS :45 |
| Receiver antenna gain （dBi） | 11 | 11 |
| Log Normal Fade(dB) | BS-BS: 0 (LOS)<br>10 （NLOS） | BS-BS: 0 (LOS)<br>10 （NLOS） |
| | BS－UE：10 | BS－UE：10 |
| | UE－UE：0 (LOS)<br>10(NLOS) | UE－UE ： 0(LOS)<br>10(NLOS) |
| Reference sensitivity level (dBm) | -110 | -108 |
| Power Control | Based C/I | Based C/I |
| Power Control Error | 0% | 0% |
| Outage | $E_b/N_0$<-0.5dB | $E_b/N_0$<-0.5 dB |
| noise power(dBm) | －106 | －104 |
| Maximum base station transmit power (dBm) | | 36 |
| Maximum terminal transmit power (dBm) | 30 | |
| Power control range(dB) | 65 | 30 |
| User distribution | random uniform distribution | random uniform distribution |

Tab.9 CDMA2000 system simulation parameters in Macro environment

| Parameter | UL | DL |
|---|---|---|
| Simulation times | Snapshot（>=800） | Snapshot (>=800) |
| MCL（dB) | BS-UE : 70 | BS-UE : 70 |
| | UE-UE :40 | UE-UE :40 |
| | BS-BS :45 | BS-BS :45 |
| Receiver antenna gain （dBi） | 11 | 11 |
| Log Normal Fade(dB) | BS-BS: 0 (LOS)<br>10 (NLOS) | BS-BS: 0(LOS)<br>10(NLOS) |
| | BS－UE：10 | BS－UE：10 |
| | UE－UE :0(LOS )<br>10(NLOS) | UE－UE: 0(LOS)<br>10(NLOS) |
| Reference sensitivity level (dBm) | -110 | -108 |
| Power Control | based C/I | based C/I |
| Power Control Error | 0% | 0% |
| Outage | $E_b/N_0$<-0.5dB | $E_b/N_0$<-0.5dB |
| noise power(dBm) | －108 | －104 |

| | | |
|---|---|---|
| Maximum base station transmit power (dBm) | | 30 |
| Maximum terminal transmit power (dBm) | 21 | |
| User distribution | random uniform distribution | random uniform distribution |

### *B. The process of simulation*

According to scenario flowcharts, this paper uses the Monte-Carlo method and related software to stimulate the interference of TD-SCDMA system and CDMA2000 system. The Fig. 2, 3 and 4 show the simulation map, simulation parameters and simulation results, respectively.

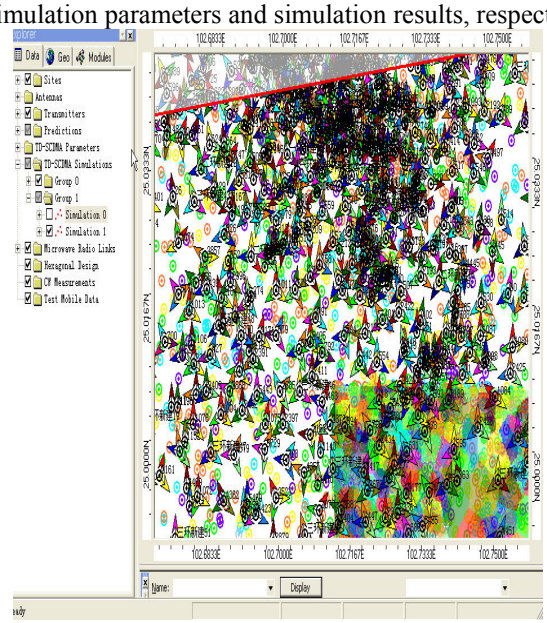


Fig. 2 simulation map

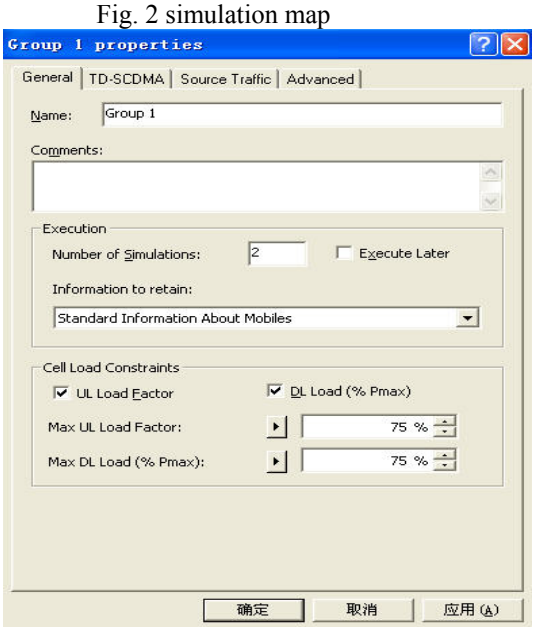



Fig.3 simulation parameters

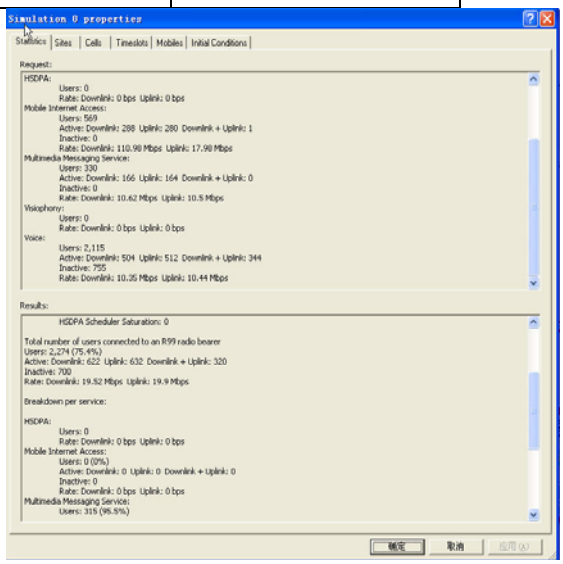



Fig. 4 simulation results

### *C. Simulation results and analysis*

*C.1 The work simulates interference of TD-SCDMA* MS to CDMA2000 BS and is divided into three simulation cases.

(1) In the omni-directional antenna and smart antenna[11][12], respectively

(2) Changing cell radius but not changing base station distance.

(3) Changing base station distance but not changing cell radius[13]

Simulation environment is the Macro-cell, and the whole area cell radius is 577m. There are three kinds of situations: the distance between operators is 0, 288 m and 577m, respectively.

In uplink, the number of users as a single TD-SCDMA system in 95% of the customer satisfaction degree reach 75%. According to data of simulation result, the graph can be drawn in Fig. 5 and 6.

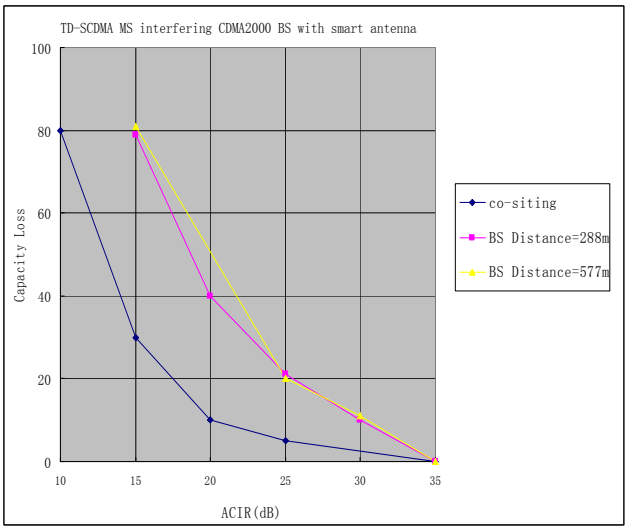



Fig.5 TD-SCDMA MS interfering CDMA2000 BS with smart antenna

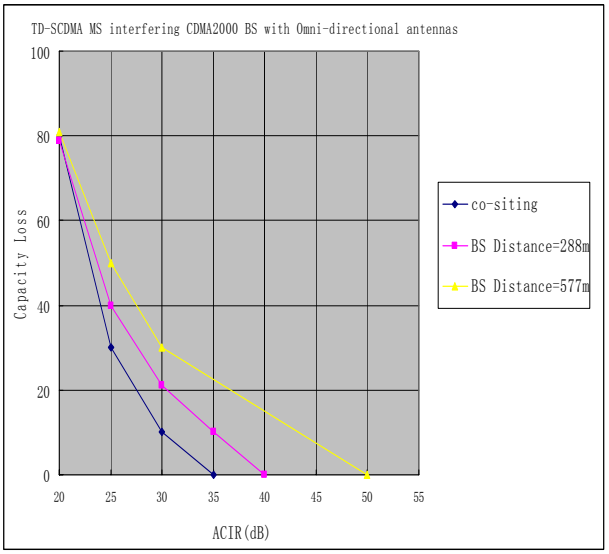



Fig.6 TD-SCDMA MS interfering CDMA2000 BS with Omni-directional antenna

Result Analysis:

(1) It can be seen that when ACIR increases, capacity loss continues to drop. The distances of BSs affect interference. For the same ACIR, the smaller are the distances of BSs, the smaller is capacity loss. As a result interference of systems is smaller.

(2) The two figures based on the use of smart antennas and omni-directional antennas are compared. For the same ACIR values, cellular capacity loss in using smart antenna is smaller than the cellular capacity loss in using omni-directional antenna. It illustrates that using smart system can inhibit interference of systems.

(3) Although the use of smart antennas can inhibit interference of systems, interference depression is still limited. A single TD-SCDMA terminal to CDMA2000 BS interference will have great inhibition in smart antenna. But the use of smart antenna can increase the number of TD-SCDMA terminals inside the cellar. Thus, improvement to overall interference of system is limited with smart antennas .

*C.2 This effort simulates different cell radius situations for smart antenna.*

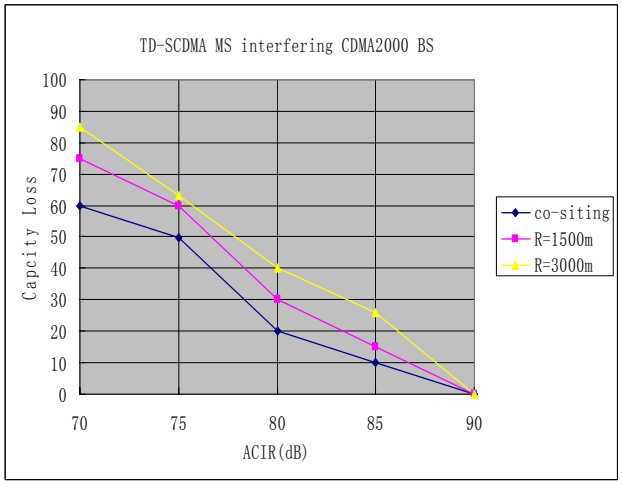



Fig.7 TD-SCDMA MS interfering CDMA2000 BS with different radius

Results:

Visibly, with the cell radius increasing, interference of TD-SCDMA MS to CDMA2000 BS increases, too. When the cell radius increases, TD-SCDMA MS in the edge of service covering the area needs more uplink power. It will maintain edge communication of TD-SCDMA system, Thus, TD MS forms larger interference to CDMA2000 BS.

According to the extreme worst results calculated by deterministic analysis methods, a certain degree of interference exists in the adjacent frequencies in CDMA2000 MS to TD-SCDMA MS, CDMA2000 MS to TD-SCDMA BS and TD-SCDMA MS to CDMA2000 BS. These interferences can be avoided and narrowed by adjusting BSs relative position. Considering the MS distribution and power control, simulation calculation is used to calculate the probability of interference interrupting communication. It can be found that in the typical cases, when the devices meet the 3GPP standard requirement, the probability is relatively low. So long as the site engineering design can analyze adjacent frequency interference area according to specific situations and turn the interfering system to secondary area, the systems can meet the co-existence need.

However, adjacent frequency interference of TD-SCDMA to CDMA2000 BS is large, Monte-Carlo simulation and deterministic analysis results show that the conventional site engineering is difficult to meet the two systems co-existence when 3GPP standards of equipment are used. To achieve the two systems coexistence or co-sited, the following measures can be taken.

Measure 1: the frequency of the two systems is reasonably arranged and system frequency interval is increased. By increasing the frequency interval, the greater ACIR of two systems can be realized. It will reduce MCL requirements. Increasing frequency interval also facilitate additional filter to provide a greater isolation between transmitter and receiver.

Measure 2: It increases system frequency of interval, avoiding co-sited. If using Omni-directional antenna, it should try to keep the adjacent sites of the two systems in the same direction. Using the antenna, decoupling can increase coupling loss. If different systems must be co-sited, antenna feeding system layout should be designed carefully. The methods such as spacing isolation and antenna decoupling are to be used to increase isolation loss between transceiver and transmitter.

## V. CONCLUSIONS

In this paper, we concentrate on interference analysis and coexistence study of TD-SCDMA and CDMA2000 systems operating in adjacent frequency bands. A thorough theoretic investigation and detailed system level simulation scenarios are presented. Most representative interference are evaluated. The results show that the interference between base stations is severe and some additional protection measures are needed. Meanwhile, TD-SCDMA system has the ability to resist interference to some extent, but it brings relatively severe interference to CDMA2000 system. It is also found that for TD-SCDMA system sites, different antennas and cell radius schemes have various impacts on CDMA2000 capacity, which should be carefully considered during TD-SCDMA deployment to ensure its coexistence with CDMA2000 system.

## . ACKNOWLEDGMENT

The title selection is mainly originated from Tianjin science and technology innovation special funds project(10FDZDGX00400) and Tianjin Key Laboratory for Control Theory and Application in Complicated Systems, Tianjin University of Technology, Tianjin 300384, China. The name of the project is “the research and development, demonstration and application of new generation mobile communication network coverage key technology”.

## REFERENCE

[1] 3GPP.TR25.942 V9 Universal Mobile Telecommunications System (UMTS); Radio Frequency (RF) system scenarios[S]. *3rd Generation Partnership Project (3GPP)*, 2010

[2] 3GPP. TR 25.945 V5 Universal Mobile Telecommunications System (UMTS); Radio requirements for low chip rate TDD option[S]. *3rd Generation Partnership Project (3GPP), 2007.*

[3] 3GPP. TR25.951V9 Universal Mobile Telecommunications System(UMTS); FDD Base Station(BS) classification[S]. *3rd Generation Partnership Project(3GPP), 2010.*

[4] 3GPP. TR25.141 V9 Universal Mobile Telecommunications System(UMTS); Base Station(BS) conformance testing(FDD)[S]. 3rd Generation Partnership Project(3GPP), 2010

[5] 3GPP TR 25.142 V9 Universal Mobile Telecommunications System (UMTS);Base Station (BS) conformance testing(TDD)[S]. *3rd Generation Partnership Project(3GPP),* 2010.

[6] J. E. Suris, L. A. DaSilva, Z. Han, and A. B. MacKenzie. “Cooperative game theory for distributed spectrum sharing,” *IEEE International Conference on Communications.* 2007: 5282-5287.

[7] R. Etkin, A. Parekh, and D. Tse. “Spectrum sharing for unlicensed bands,”*IEEE Journal on Selected Areas in Communications*, 2007,25(3):517-528.

[8] H. Boche and S. Stanczak. “Strict convexity of the feasible log-SIR region,” *IEEE Transactions on Communications*, 2008, 56(9):1511-1518.

[9] F.Baccelli, B. Blaszczyszyn, and F. Tournois. “ Spatial averages of coverage characteristics in large CDMA networks,”. *Wireless Networks*, 2002, 8(6):569-586.

[10] Chun Chung Chan and S.V. Hanly. “Calculating the outage probability in a CDMA network with Spatial Poisson traffic,”. *IEEE Transactions on Vehicular Technology.* 2001,50(1):183-204.

[11] Wan Choi and J. G. Andrews. “Downlink performance and capacity of distributed antenna systems in a multicell environment,”*IEEE Transactions on Wireless Communications*, 2007, 6(1):69-73 .

[12] A. Forenza, M. R. McKay, A. Pandharipande, R.W. Heath, Jr., and I. B.Collings. “Adaptive MIMO transmission for exploiting the capacity of spatially correlated channels,” *IEEE Transactions on Vehicular Technology, 2007,*56(2):619-630.

[13] J.G. Andrews, W. Choi, and R.W. Heath, Jr. “ Overcoming interference in spatial multiplexing MIMO cellular networks.”. *IEEE Wireless Communications Magazine*, 2007, 14(6):95-104.

[14] F. Baccelli, B. Blaszczyszyn, and P. Muhlethaler. “An ALOHA protocol for multihop mobile wireless networks,” *IEEE Transactions on Information Theory*, 2006, 52(2):421-436.

[15] D. Avidor, S. Mukherjee, J. Ling, and C. Papadias. “On some properties of the proportional fair scheduling policy,” *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications,* 2004: volume 2, 853-858.

[16] V. Chandrasekhar, J. G. Andrews, and A. Gatherer. “Femtocell networks: a survey,” *IEEE Communications Magazine,* 2008, 46(9):59-67.

[17] Kun Yang, Yumin Wu, and Hsiao-Hwa Chen. “QoS-aware routing in emerging heterogeneous wireless networks,” *IEEE Communications Magazine*, 2007, 45(2):74-80.

[18] Ghasemi and E. Sousa. “Spectrum sensing in cognitive radio networks: the cooperation-processing tradeoff,” *Wireless Communications and Mobile Computing*, 2007, 7(9):1049-1060.

**Chen Hao** has been a Ph.D candidate in School of Electronic and Information Engineering, Tianjin University. His current research interests include wireless communication , interference of different communication system and power control, etc. At the same time, he is also a teacher of Computer Science and Information Engineering College , Tianjin University of Science & Technology.

**Yang Tong** received his B.S. degree in Radio Engineering from Beijing University of Posts and Telecommunications and now is at the China Mobile Group Tianjin Co., Ltd. He is currently serving as a senior technical expert of network optimization, with years of wireless network optimization, and management experience. In 2006, in the quality of the wireless network international standard test close to customers sense, led the Network Optimization group to rank 7th of 237 operators around the world. At the same year, the paper, " the innovative application of Optical distribution system in large traffic emergency communications "by the Tianjin Science and Technology Progress Award for third place; during the Olympic Games telecommunication support period in 2008, focusing on the Beijing-Tianjin inter-city organization to complete the construction and optimization of high-speed rail, creating a optimized high-speed railway pioneer, won the title of tech Olympics and excellent workers; in 2009, responsible for high-speed Beijing-Tianjin inter-city railway network GSM Wireless Network program, won the Tianjin Commission of Science, China Mobile Group, and other science and technology innovation awards.

**Teng Jian-fu** received M.Sc. degree in telecommunication and electronic engineering from Tianjin University, China, in 1983. He received Ph.D. degree in electrical and electronic engineering from University of London, U.K. in 1988.

From 1988 to 2004, he worked in Tianjin University, where he became a vice dean of the graduate school and provost of the university. Since 2004, he worked as a vice president of Tianjin University of Commerce. Currently he is a vice president of Tianjin University of Technology. His research interests include filter theory and design, electronic circuit design and signal processing.

**He Hong** is Professor Tianjin University of Technology Institute of automation, the main research direction of signal and information processing and electromagnetic compatibility. worked as a vice president of Tianjin University of Commerce. Currently she is a vice president of Tianjin University of Technology. Her research interests include filter theory and design, electronic circuit design and signal processing.

# An Improved Localization Algorithm of Nodes in Wireless Sensor Network

Xiaohui Chen*
College of Computer and Information Technology, China Three Gorges University, Yichang, China
Email: chui@ctgu.edu.cn

Jing He, Bangjun Lei, Tingyao Jiang
College of Computer and Information Technology, China Three Gorges University, Yichang, China
Email: Edwin.ho-ctgu@hotmail.com

***Abstract*—Aiming at improving the precision of nodes' localization, this paper analyses the source of the localization error when using the least square algorithm, and proposes the principle to choose the benchmark anchor nodes in reducing the power of equation. Based on this principle, the algorithm choosing the nearest node as the benchmark anchor node in LSM and the algorithm choosing the synthetic nearest node as the benchmark anchor node in LSM are put forward. It is proved by the simulation in MATLAB that the improved LSM can effectively improve the precision of the nodes localization.**

***Index Terms*—wireless sensor networks, node localization precision, lest square algorithm**

## I. INTRODUCTION

Nowadays, wireless sensor network has been implicated in many domains, such as environmental monitoring fields, battlefields [1-4]. As the localization of the nodes is very important to make the application of WSN effectively, it is very important to improve the precision of the localization algorithms in WSN.

Generally, the sensor nodes are deployed randomly and their locations may not be acquired prematurely. The localization of the sensors has evoked a tremendous attention in these occasions. At the same time, the node energy consumption is one of the key factors which should be considered in WSN, so the algorithm of the localization must shorten its computing time.

The localization algorithms of the nodes in WSN are mainly divided into two methods range-free and range-based. The localization of range-free algorithm is on the basis of the network connectivity. Although both the cost and the consumption of its equipment are very low, the localization accuracy is very low, usually its' accuracy is only 40% of the communication radius. Additionally, the distribution of nodes can also influence the performance of localization algorithms. And the localization methods in range-based algorithms include time of arrival(TOA), angle of arrival(AOA), time difference of arrival(TDOA) and received signal strength(RSSI) [5]-[7]. AOA is used to get the coordinates of the sensor nodes by measuring the distance and the angles between unknown nodes and anchor nodes. RSSI is used to calculate the distance by calculating the energy diminishment. The characteristic of those algorithms is to get the geometric relation between the nodes and the coordinates in a physical way, and thus realize the nodes' localization.

In order to reduce the influence of the measuring distance error and the distance estimation error, great deals of methods have already been developed to solve the localization of nodes in WNS. Some existent localization algorithms adopt cycle accuracy method [8-13]. Such as Savarese proposed two localization algorithms: cycle accuracy-Cooperative ranging [8] and Two-Phase localization [9] that can decrease the influence of distance error; in 2002, Savvides [10] proposed n-hop multilateration primitive localization algorithm, where Kalman filtering technique is used to calculate the accurate coordinates circularly, it reduces the accumulation error; Bergamo averaged the measuring distance results to increase the localization accuracy with the attenuation of analogy signal [11]; in 2005, in order to improve the accuracy of localization, Guha used the method of non-convex constraints and time detecting to estimate the nodes' localization [12]; in 2006, Cao proposed a localization refinement algorithm based on Cayley-Menger determinant [13]; in 2007, based on Second-order Cone Programming, Srirangarajan solved the accuracy problem in the case that anchor position of the nodes is not accurate localization [14].

In this paper, an improved localization algorithm based on the least squares algorithm is proposed. Firstly, the paper analyses the reason of the extra error when subtracting the equation of distance between the benchmark anchor nodes and the ordinary nodes from the others in the LSM algorithm. Secondly, the paper proposes the principle of choosing the benchmark anchor nodes for reducing the errors by the distance' errors of

Manuscript received May 1, 2011; revised June 1, 2011; accepted July 1, 2011.

This researcher was supported by the Research and Development Project of Science and Technology of Yichang (Grant No.A2011-302-14); the National Natural Science Foundation of China (Grant No. 60972162); the National Natural Science Foundation of China (Grant No.41172298).

*Corresponding author: Xiaohui Chen


doi:10.4304/jnw.7.1.110-115

the benchmark anchor nodes. Lastly, the localization errors of these three algorithms are compared by using simulation in MATLAB, the coordinates' errors of the nodes decrease.

## II. THE MODEL OF NODES' LOCALIZATION

The localization in wireless sensor networks usually calculates the coordinates of nodes through measuring the distances between the nodes to be measured and the three anchors around. According to the neighboring nodes' localization information and the measuring distance among the nodes, the node coordinates can be calculated.

It respectively uses the three anchor nodes to be the centre of three circles, similarly uses the distance between unknown nodes and anchor nodes to be the radius of three circles, and the coordinates of this point which the three circles have intersected is exactly that of the node under test.

Assuming the coordinate of the unknown node to be (x, y); the coordinates of three anchor nodes are respectively set to be $(x_1, y_1)$, $(x_2, y_2)$, $(x_3, y_3)$; the distance between the unknown node and the three anchor nodes are set to be d1, d2, d3, the model of the localization figure is as shown in figure1.

According to this model we can get the equations as follow.

$$\begin{cases}(x_1-x)^2+(y_1-y)^2=d_1^2\\(x_2-x)^2+(y_2-y)^2=d_2^2\\(x_3-x)^2+(y_3-y)^2=d_3^2\end{cases}. \tag{1}$$

Subtracting the third equation from the first and the second in (1), we can get the following expression.

$$\begin{cases}a_1x+b_1y=c_1\\a_2x+b_2y=c_2\end{cases}. \tag{2}$$

where

$$\begin{cases}a_i=2(x_i-x_3)\\b_i=2(y_i-y_3)\\c_i=2(x_i^2-x_3^2)+2(y_i^2-y_3^2)-2(d_i^2-d_3^2)\end{cases},i=1,2.$$

Solving the equations simultaneously, the coordinate of the unknown node D can be obtained.

$$\begin{cases}x=\dfrac{b_1c_1-b_1c_2}{a_1b_2-a_2b_1}\\y=\dfrac{a_1c_2-a_2c_1}{a_1b_2-a_2b_1}\end{cases}$$

## III. LEAST-SQUARE ALGORITHM FOR LOCALIZATION REFINEMENT

We can get three circles by using the three anchor nodes to be the centre, and using the distance between the unknown node and the three anchor nodes to be the radius. These three circles can intersect into a point, and the coordinate of the unknown node can be obtained.

Because of the environmental influence on distance measurement, such as, the signal of wireless are mainly influenced by transmission medium, the multi-path transmission, signal reflections, antenna gain, etc, the error is produced. When the error appears, these three circles will not have one common intersection.

Therefore, we cannot get the coordinates of the nodes. In order to solve this problem, the least square algorithm has been used, which is very concise and practical to the WSN which attaches importance to energy consumption.

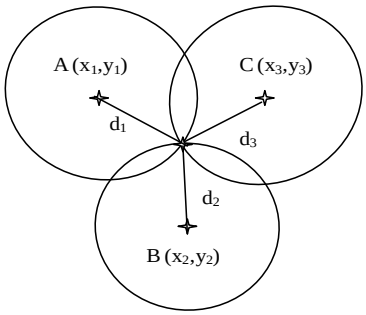



Figure1. The model of Nodes' localization

Assuming the coordinate of the unknown node to be (x, y), every anchor node coordinate is respectively given to be $(x_1, y_1)$, $(x_2, y_2)$, $(x_3, y_3)\ldots (x_n, y_n)$, then we can get the following expressions.

$$\begin{cases}(x_1-x)^2+(y_1-y)^2=d_1^2\\(x_2-x)^2+(y_2-y)^2=d_2^2\\\quad\cdots\\(x_n-x)^2+(y_n-y)^2=d_n^2\end{cases}. \tag{3}$$

Respectively subtracting the last equation from the others in (3), we get the following expression,

$$\begin{cases}2(x_1-x_j)x+2(y_1-y_j)y=(x_1^2-x_j^2)+(y_1^2-y_j^2)-(d_1^2-d_j^2)\\2(x_2-x_j)x+2(y_2-y_j)y=(x_2^2-x_j^2)+(y_2^2-y_j^2)-(d_2^2-d_j^2)\\\quad\cdots\\2(x_n-x_j)x+2(y_n-y_j)y=(x_n^2-x_j^2)+(y_n^2-y_j^2)-(d_n^2-d_j^2)\end{cases}$$

And we can use (4) to instead of it.

$$AX^*=B^*. \tag{4}$$

The $X^*$ can be solved.

$$X^*=(A^TA)^{-1}A^TB^*. \tag{5}$$

where

$$A=2\times\begin{bmatrix}x_1-x_j & y_1-y_j\\x_2-x_j & y_n-y_j\\\vdots & \vdots\\x_n-x_j & y_n-y_j\end{bmatrix}.$$

$$B^*=\begin{bmatrix}(x_1^2-x_j^2)+(y_1^2-y_j^2)-(d_1^2-d_j^2)\\(x_2^2-x_j^2)+(y_2^2-y_j^2)-(d_2^2-d_j^2)\\\vdots\\(x_n^2-x_j^2)+(y_{n-1}^2-y_j^2)-(d_n^2-d_j^2)\end{bmatrix}.$$

$$X^* = \begin{bmatrix} x \\ y \end{bmatrix}. \quad (6)$$

We can solve the linear equation and get the corresponding value of X as (6).

## IV. LSM_DS(CHOOSING THE BENCHMARK ANCHOR NODES OF THE NEAREST NODE)

Using the least square method can reduce the influence of the measurement error of the distance between the anchor node and the nodes to be measured. However, before using the least square method, we need to iterate all the equations of the distance between the anchor nodes and the nodes to be measured, and in the iteration process, we need to subtract all the equations from the benchmark equation. Because the measurement error of the distance in the benchmark equation has great impact on the accuracy of localization, it is important to choose the suitable benchmark node.

As the benchmark equation has a great part in the iteration equation, so it is practicable to choose the anchor nodes with the minimum measurement error of the distance as the benchmark node.

It can be presented by the (7) and (8).

$$\begin{cases} (x_1 - x)^2 + (y_1 - y)^2 = (d_1 + e_1)^2 \\ (x_2 - x)^2 + (y_2 - y)^2 = (d_2 + e_2)^2 \\ \quad .... \\ (x_n - x)^2 + (y_n - y)^2 = (d_n + e_n)^2 \end{cases}. \quad (7)$$

The simulation (with 100 ordinary nodes) results are shown by the Figure2.

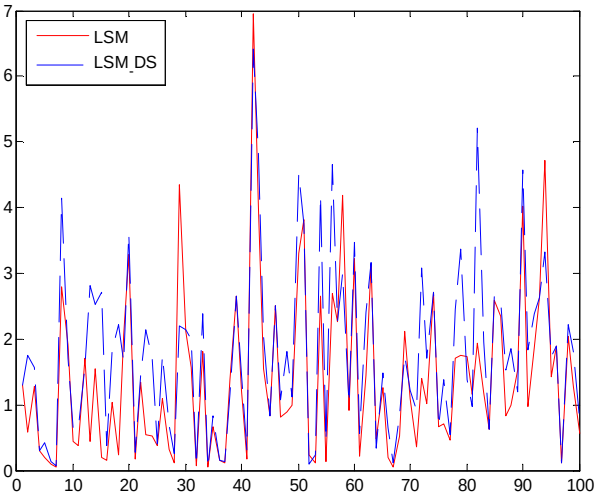



Figure2. Comparing the error between LSM and LSM_DS

Comparing the errors between LSM and LSM_DS, we get the results are shown in Figure.2. The ordinate represents the error; the abscissa represents the number of nodes; the solid line represents the error curve of LSM, and the dashed line represents the error curve of LSM_DS.

We can see from the Figure2 that the LSM_DS algorithm only improves a part of the precision, but the ensemble precision is not improved. It is incomplete to improve the localization precision by choosing the anchor nodes with the minimum measurement error of the distance as the benchmark node. Therefore, we must reconsider the choice of benchmark node.

## V. LSM_DR (CHOOSING THE BENCHMARK ANCHOR NODES OF THE MINIMUM RELATED DISTANCE NODE)

Supposing the node "J" is the benchmark node. As choosing the benchmark anchor nodes of the nearest node in the iteration process, the choice of reference equation J is the expression of the node whose distance is the shortest between the unknown node and the anchor node, Founding by actual simulation, this approach does not effectively reduce the node location error under the error rate in the same distance cases, so the choice of reference equation J needs further improvements.

### *A. The Analysis of Derivation Process*

According to expressions in part IV, when the error appears, the (4) and (5) are changed to (8) and (9).

$$AX = B^* + E. \quad (8)$$

$$X = (A^T A)^{-1} A^T (B^* + E). \quad (9)$$

where

$$B = B^* + E.$$

$$E = \begin{bmatrix} e_1^{'} \\ ... \\ e_n^{'} \end{bmatrix}.$$

And the (10) can be transformed to (10).

$$X = (A^T A)^{-1} A^T B + (A^T A)^{-1} A^T E. \quad (10)$$

In order to expound clearly, we suppose the fellows:

$$F = (A^T A)^{-1} A^T.$$

Then, we can get:

$$X = F(B + E) = X^* + FE$$
$$X - X^* = FE.$$

And both sides are computed the Euclidean norm, we can get:

$$\left\| X - X^* \right\|_2 = \left\| FE \right\|_2.$$

By using Cauchy-Schwarz inequality, the following inequality can be obtained.

$$\left\| FE \right\|_2 = \sqrt{\sum_{i=1}^{n} \left| \sum_{k=1}^{n} a_{ik} e_k \right|} \le \sqrt{\sum_{i=1}^{n} (\sum_{k=1}^{n} |a_{ik}| \|e_k|)^2}$$
$$\le \sqrt{\sum_{i=1}^{n} \left[ (\sum_{k=1}^{n} |a_{ik}|^2)(\sum_{k=1}^{n} |e_k|^2) \right]}$$
$$= \left\| F \right\|_F \left\| E \right\|_2.$$

Based on the above conclusion, we can get:

$$\left\| X - X^* \right\|_2 = \left\| FE \right\|_2 \le \left\| F \right\|_F \cdot \left\| E \right\|_2.$$

When the Euclidean norm of error vector E is the minimum, the Euclidean norm of vector FE can be smaller, that is to say:

$$\left\|X - X^*\right\|_2 \propto \left\|E\right\|_2 .$$

Because the Euclidean norm of vector (X-X$^*$) equals that of EF, and the Euclidean norm of vector (X-X$^*$) can be transformed to (11).

$$\begin{aligned}\left\|X - X^*\right\|_2 &= \left\|\begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} x^* \\ y^* \end{bmatrix}\right\|_2 = \left\|\begin{bmatrix} x - x^* \\ y - y^* \end{bmatrix}\right\|_2 . \\ &= \sqrt{(x - x^*)^2 + (y - y^*)^2}\end{aligned} \quad (11)$$

According to (11), the Euclidean norm of vector (X- X$^*$) is just the error between the calculating value of the node and the real value of it. Based on the above proof, the lower the error measurement is, the higher the recognition precision can be got. So a theorem can be given.

**Theorem 1:** If the Euclidean norm of E is the smallest, the error between the calculating value of the node and the real value of it can be lower.

So the problem is transformed to find an anchor node, the measurement error of which can make the Euclidean norm of E be the minimum.

In (7), when the equation of benchmark node minus the equation of other nodes, the right expression of (7) is:

$$B = \begin{bmatrix} (x_1^2 - x_j^2) + (y_1^2 - y_j^2) + (d_1 + e_1)^2 - (d_j + e_j)^2 \\ (x_2^2 - x_j^2) + (y_2^2 - y_j^2) + (d_2 + e_2)^2 - (d_j + e_j)^2 \\ .... \\ (x_n^2 - x_j^2) + (y_n^2 - y_j^2) + (d_n + e_n)^2 - (d_j + e_j)^2 \end{bmatrix}.$$

In order to discourse clearly, let the following expression be:

$$\begin{cases} b_i = (x_i^2 - x_j^2) + (y_i^2 - y_j^2) + (d_i^2 - d_j^2) \\ e_i^{'} = (2d_i e_i - 2d_j e_j) + (e_i^2 - e_j^2) \end{cases} i = 1,2,3...,n . \quad (12)$$

So the B can be transformed to (13).

$$B = \begin{bmatrix} c_1 + (2d_1 e_1 - 2d_j e_j) + (e_1^2 - e_j^2) \\ c_2 + (2d_2 e_2 - 2d_j e_j) + (e_2^2 - e_j^2) \\ .... \\ c_n + (2d_n e_n - 2d_j e_j) + (e_n^2 - e_j^2) \end{bmatrix}. \quad (13)$$

Let

$$B = B^* + E .$$

where

$$B^* = \begin{bmatrix} c_1 \\ ... \\ c_n \end{bmatrix}, E = \begin{bmatrix} e_1^{'} \\ ... \\ e_n^{'} \end{bmatrix}.$$

As we know, measurement error is proportional to the distance, and let $e_i = \gamma_i d_i$ .

So the (12) can be transformed to (14).

$$\begin{aligned} e_i^{'} &= (\frac{2}{\gamma_i} e_i^2 - \frac{2}{\gamma_j} e_j^2) + (e_i^2 - e_j^2) \\ &= (\frac{2}{\gamma_i} - 1) e_i^2 - (\frac{2}{\gamma_j} - 1) e_j^2 \end{aligned} . \quad (14)$$

In the measuring range, the impact factor caused by the distance measurement errors can be approximately set to the same value, so let $\gamma_i = \gamma_j = \gamma$ .

Equation (14) can be transformed to (15).

$$e_i^{'} = (\frac{2}{\gamma} - 1)(e_i^2 - e_j^2) . \quad (15)$$

And the error vector can be got.

$$E = (\frac{2}{\gamma}) \begin{bmatrix} e_1^2 - e_j^2 \\ ... \\ e_n^2 - e_j^2 \end{bmatrix}.$$

Because:

$$\begin{aligned} \left\|E\right\|_2 &= \left\| (\frac{2}{\gamma}) \begin{bmatrix} e_1^2 - e_j^2 \\ ... \\ e_n^2 - e_j^2 \end{bmatrix} \right\| \\ &= (\frac{2}{\gamma}) \left\| \begin{bmatrix} e_1^2 - e_j^2 \\ ... \\ e_n^2 - e_j^2 \end{bmatrix} \right\| \\ &= \sqrt{(\frac{2}{\gamma}) \sum_{i=1}^{n} (e_i^2 - e_j^2)^2} \end{aligned} . \quad (16)$$

And both the right and left of (16) are squared.

$$\left(\left\|E\right\|_2\right)^2 = (\frac{2}{\gamma}) \sum_{i=1}^{n} (e_i^2 - e_j^2)^2 .$$

$$\left\|E\right\|_2 \propto \left(\left\|E\right\|_2\right)^2 \propto J = \sum_{i=1}^{n} \left| e_k^2 - e_j^2 \right| .$$

According to the Theorem 1, only if the Euclidean norm of E can be the smallest, Node identification accuracy will be higher.

*B. Algorithm process*

- Step1: Find the all anchor nodes in the perception of distance
- Step2: For each anchor node, compute its distance from the rest nodes, and get its $J_i$ .
- Step3: Choose the minimum from $J_i, i = 1,2...n$ and its corresponding node is the benchmark node.
- Step4: Degree reduction.
- Step5: Refinement using the least square algorithm.

We choose the anchor node with minimum related distance as benchmark anchor node. In the LSM_DR algorithm, the anchor node which has the smallest $J_i$ is chosen to be the benchmark node, and the simulation is shown by the Figure3.

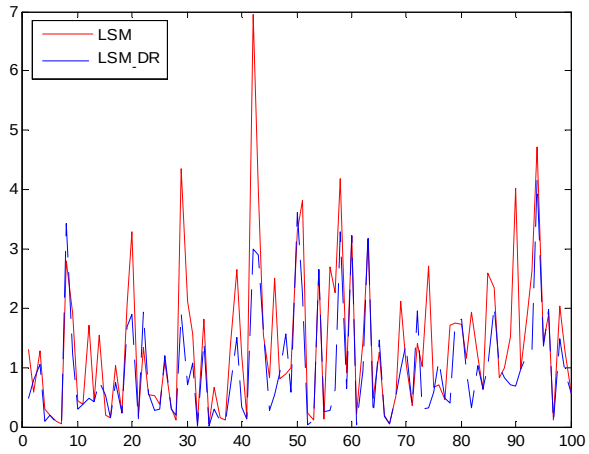



Figure3. Comparing the error between LSM and LS_DR.

In Firure3, the ordinate represents the error; the abscissa represents the number of nodes; the red solid line represents the error curve of LSM; the dashed line represents the error curve of LSM_DR. As shown in Figure3, we can see that the majority of nodes' localization error is reduced.

## VI. Simulation

By using random deployed 100 ordinary nodes with seven anchor nodes around them, the distances between the ordinary node and the anchor nodes are unequal and disproportionate, the distance error is a random positive number and it is less than the 30% of the distance. The simulation by using LSM, LSM_DS and LSM_DR about localization is shown as the Figure4 and the Table1.

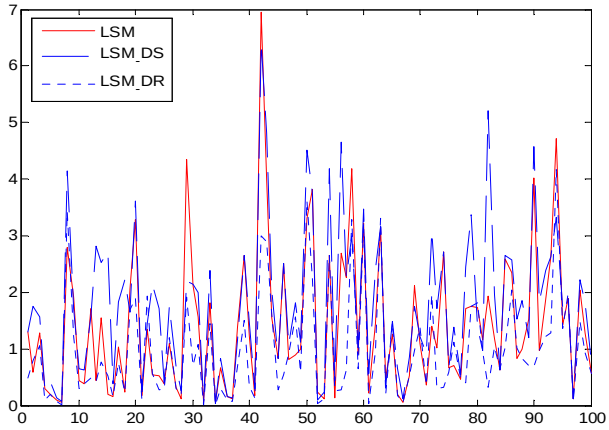



Figure4. Comparing the errors among LSM, LS_DS and LS_DR.

As shown in Figure4, the ordinate represents the error; the abscissa represents the number of nodes; the red solid line represents the error curve of LSM; the dashed line represents the error curve of LSM_DS; the dotted line represents the error curve of LSM_DR.

TABLE I.
THE ERRORS OF THE THREE METHODS

| Algorithm | LSM | LSM_DS | LSM_DR |
|---|---|---|---|
| Average error | 1.4044 | 1.7691 | 0.9784 |

As shown in table I, we can see that the average error of LSM_DR is the smallest among the three algorithms; the simulation has proved that the LSM_DR can obtain more accurate localization while making the algorithm concise.

## VII. Conclusions

This paper aims at improving the localization precision of nodes when using the least squares algorithm in WSN. Firstly the paper analyses the reasons why the extra error exists when subtracting the equation of distance between the anchor nodes and the ordinary nodes from the others in the LSM algorithm, and then proposes the principle of choosing the benchmark anchor nodes by using the distance' errors of the benchmark anchor nodes in order to reduce the errors. The localization errors of these three algorithms are compared by using simulation in MATLAB, the result shows that the coordinator's errors of the nodes decrease, and the improved algorithm proposed in this paper can effectively improve the precision of the nodes localization.

## Acknowledgment

This research was supported by the Research and Development Project of Science and Technology of Yichang (Grant No.A2011-302-14); the National Natural Science Foundation of China (Grant No. 60972162);the National Natural Science Foundation of China (Grant No.41172298).

## References

[1] K. Lorincz, D. Malan, T.R.F. Fulford-Jones, A. Nawoj, A. Clavel, V.Shnayder, G. Mainland, M. Welsh, S. Moulton, Sensor networks for emergency response: challenges and opportunities, Pervasive Computing for First Response (Special Issue), IEEE Pervasive Computing ,2004.

[2] T. Gao, D. Greenspan, M. Welsh, R.R. Juang, A. Alm, Vital signs monitoring and patient tracking over a wireless network, Proceedings of the 27th IEEE EMBS Annual International Conference, 2005.

[3] G. Wener-Allen, K. Lorincz, M. Ruiz, O. Marcillo, J. Johnson, J. Lees, and M. Walsh, Deploying a wireless sensor network on an active volcano. Data-Driven Applications in Sensor Networks (Special Issue), IEEE Internet Computing, 2006.

[4] He T, Huang CD, Blum B.M, range-free localization schemes in large scale sensor networks, Proc of the 9th Annual Int'l Conf on Mobile Computing and Networking. San Diego: ACM Press, 2003:81-95.

[5] Qun, Wan, Wanlin, Yang, A fast algorithm for wireless sensor networks localization based on metric least squares scaling, IET International Conference on Wireless Mobile and Multimedia Networks Proceedings, ICWMMN 2006.

[6] Born, Alexander; Timmermann, Dirk; Bill, Ralf A distributed linear least squares method for precise localization with low complexity in wireless sensor networks Distributed Computing in Sensor Systems - Second IEEE International Conference, DCOSS 2006, Proceedings.

[7] Zenon Chaczko, Ryszard Klempous, Jan Nikodem, Michal Nikodem, Methods of Sensors Localization in Wireless Sensor Networks, Proceedings of the 14th Annual IEEE

International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS07), 2007.
[8] Savarese C, Rabaey JM, Beutel J, Localization in distributed ad-hoc wireless sensor network, Proceeding of the 2001 IEEE International Conference on Acoustics, Speech, and Signal Salt Lake, 2001.
[9] Avvides A, Park H, Srivastava M.B, The bits AND flops of the N-hop multilateration primitive for node localization problems, Proceeding of the 1st ACM International Workshop on Wireless Sensor Networks and Applications, Atlanta, 2002.
[10] Savarese C, Rabaey JM, Langendoen K, Robust Localization Algorithms for Distributed Ad Hoc Wireless Sensor Networks, Proceedings of the USENIX Technical Annual Conference, Monterey, USA, 2002.
[11] Bergamo P, Mazzini G.: Localization in sensor networks with fading and mobility. Processing of the 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, NJ, USA, 2002.
[12] Guha S, Murty R N, Sirer E G, A unified node and event localization framework using non-convex constraints, Proceeding of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing, IL USA, 2005.
[13] Cao M, Anderson B D O, Morse A.S, Sensor network localization with imprecise distances, Systems and Control Letters, 2006.
[14] Srirangarajan S, Tewfik A H, Luo Z Q, Distributed sensor network localization with inaccurate anchor positions and noisy distance information. IEEE International Conference on Acoustics, Speech and Signal Processing, HI, USA, 2007.
[15] Xiaohui Chen, Jing He, Jinpeng Chen, An Improved Localization Algorithm for Wireless Sensor Network, Intelligent Automation and Soft Computing, Vol. 17, No. 5, pp. 507-517, 2011.
[16] Xiaohui Chen, Jing He, Jinpeng Chen, Localization of Node in Wireless Sensor Network Based on Conjugate Algorithm, 2011 International Conference on Computer Science and Service System, 2011.

**Xiaohui Chen** was born in Sichuan province, China in 1967, he received his master's degree from Huazhong University of Science and Technology (HUST) , Wuhan, China, in 1996, He is currently an associate professor in the college of computer and information technology, China Three Gorges University. His research interests include wireless sensor networks, data mining, and intelligent control.

**Jing He** was born in Hubei province, China in 1987, he received his bachelor's degree from Huazhong University of Science and Technology (HUST) , Wuhan, China,in 2010, and he is seeking for his master's degree in the college of computer and information technology , China Three Gorges University now. His research fields include wireless sensor networks, embedded system.

**Bangjun Lei** was born in 1973. he got his Ph.D title from TUDelft, the Netherlands In 2003. He is professor at the College of Computer and Information Technology, China Three Gorges University. He is active in low-level image processing, 3-D imaging and computer vision. He enjoys developing practical multimedia applications.

**Tingyao Jiang** was born in 1969. He received the Ph.D. degree in College of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China, in 2004. He is a professor in the College of Computer and Information Technology, China Three Gorges University. His research interests include fault-tolerant computing, network security, software engineer.

# Malicious Nodes Detection in MANETs: Behavioral Analysis Approach

Yaser khamayseh, Ruba Al-Salah, Muneer Bani Yassein
Jordan University of Science and Technology, Dept of Computer Science
Irbid, 22110, Jordan
yaser@just.edu.jo, ruba_3cs@yahoo.com, masadeh@just.edu.jo

***Abstract*— The increased popularity and usage of wireless technologies has opened the doors for new emerging applications in the domain of networking. One emerging and promising areas is the domain of Mobile Ad Hoc Networks (MANETs). A mobile ad hoc network is a collection of wireless mobile nodes that form a dynamic network without the need for infrastructure or centralized points.**

**The dynamic nature of ad hoc networks presents many security challenges. Secure routing is a promising area for achieving better security for the network by protecting the routing protocols against malicious attacks. Several secure routing protocols have been proposed in the literatures that were successful in avoiding and preventing some types of security attacks in MANETs. However, MANETs are still vulnerable to other types of attacks. Hence, there is a need for introducing an efficient mechanism to detect malicious nodes.**

**In this paper, a new mechanism is proposed that improves the performance of routing protocol against the malicious attacks. Since a malicious node behaves in abnormal ways, this mechanism proposes observing nodes behavior such as nodes' mobility, and avoiding communication through these nodes which may lead to more secure routing.**

***Index Terms*— Malicious Nodes, MANETS, Behavioral Analysis.**

## I. INTRODUCTION

A mobile ad hoc network is a network that consists of mobile nodes that communicate wirelessly and does not have an infrastructure. Nodes in mobile ad hoc network communicate through either single-hop or multi-hop modes. Therefore, in this environment, each node acts as a router as well as a host.

A major function in mobile ad hoc networks is the route discovery process [1], in which a route from a specific source to a specific destination is discovered in order to transfer data packets via this discovered route. Many routing protocols such as DSR [2] and AODV [3] have been proposed in the literature for route discover and data transmission. In the data transmission phase, each intermediate node in the network participates in forwarding data and control packets to other nodes. Most ad hoc routing protocols such as AODV and DSR were not originally designed to be secure against malicious attacks; as they rely on simple implicit trust-your-neighbors relationships [4].

However, and due to the increase popularity of the ad hoc networks, some nodes may act in the network in a negative way. These nodes are called malicious nodes, and may perform attacks in the network to jeopardize the network resources. Due to the dynamic nature of the MANETs topology and the absence of infrastructure, MANETs are more vulnerable to attacks [5]. This dynamic structure of nodes may disturb the trust relationship among nodes. The lack of central points makes the detection process of attacks difficult as it is hard to monitor the traffic in a dynamic and large scaled network [5]. All these characteristics of MANETs allow the attackers to easily target the network and savatage its resources by disturbing and jamming the communication between legitimate nodes. Malicious nodes can perform adversarial attacks that can damage the basic aspects of security, such as integrity, confidentiality and privacy [7]. Security is more critical in some applications such as military, law enforcement, and rescue missions [6]. Consequently, security in MANETs has attracted further attention [8].

### *A. Security Goals*

In MANET, all networking functions, such as routing and data transmission, are performed by the nodes without the need for a central point to control and organize the resource management process. Therefore, security is very challenging [5] [7]. Security vulnerabilities for a network consist of the following aspects: Confidentiality, integrity, authentication, non-repudiation, and access and usage control.

### *B. Attack Patterns on Ad Hoc Routing Environment*

Attacks in MANETS can be classified into two types, passive and active attacks [5]. In the passive attacks, attackers do not disrupt the normal routing operations; it listens to the routing traffic to acquire some valuable information. On the other hand, an active attacker injects packets into the network, eavesdrops and tries to compromise the network resources by performing a Denial Of Service (DOS) attack [8].

Encryption techniques and mechanisms are used to encrypt the sent messages and therefore prevent the passive attackers from understanding the content of the message even if it gets it. On the other hand, the active attacks are more serious and can cause severe problems since an attacker may drop, inject, and fabricate messages. To avoid this type of attacks, the best solution is to detect the attackers (malicious nodes) and avoid them [8]. Common attack patterns identified in MANETS are [9]: Tunneling, Spoofing, Blackhole attack, Wormhole attack, Routing table overflow attack,


doi:10.4304/jnw.7.1.116-125

As mentioned before, due to the highly dynamic nature of MANETs, they are more vulnerable to attacks. Thus, several solutions based on securing the routing protocols have been proposed such as: Security-Aware Routing Protocol (SAR) [4], Secure Ad hoc On-Demand Distance Vector Routing Protocol (SAODV) [8], Secure Routing Protocol (SRP) [12], Authenticated Routing for Ad hoc Network (ARAN) [14], Ariadne [15].

The main goal of most of the above mentioned proposed protocols is to ensure security in the communication process between nodes and preclude any attack patterns. However, none of these protocols succeeded in preventing all kinds of attacks. Therefore, there is need to design a secure routing protocol that overcomes the shortcomings of these protocols.

We note that the malicious node attacks on the network are recrudescent and cannot be limited. A common solution to handle this limitation is to detect and avoid the malicious nodes in the network and hence communicating via reliable routes [17]. Moreover, we also note that a malicious node may have abnormal behavior compared to normal nodes in the network such as it may have higher mobility than other nodes that enables it to perform its suspicious actions. Therefore, the node reliability increases as the node dwell in a specific region for a long time [17]. Thus, avoiding communications via malicious nodes could improve the network throughput. Besides the nodes' mobility, there are other factors that must be taken into account in order to detect and avoid harmful nodes such as node traffic patterns.

The main goal of this paper is to enhance and revamp the performance of the secure routing protocols for MANETs by detecting and avoiding malicious nodes. The proposed scheme identifies malicious paths between the source and the destination nodes. A malicious path is a path that may contain one or more malicious nodes.

The rest of the paper is organized as follows: Section 2 presents a brief review of the existing secure routing protocols in mobile ad hoc networks and some of the techniques used to secure MANETs. Section 3 describes the proposed scheme. Section 4 presents the experimental results. Finally, Section 5 concludes the paper and outlines future work.

## II. Related Work

A considerable amount of research has been conducted in the area of MANETs security [23, 24, 25]. Some proposed schemes introduce new routing protocols that take security into account and hence, these secure routing protocols can prevent some types of attacks. Other proposed schemes can detect and deal with malicious nodes. The following sections discuss some of the techniques that have been proposed concerning security in MANETs.

### *A. Ad Hoc Secure Routing Protocols*

The current routing protocols for MANETs, such as AODV and DSR, have vulnerabilities that allow can permit various attacks on the network. Therefore, several secure routing protocols were proposed in literature that uses security as a metric in order to have more reliable routes.

SAR [4] is implemented in top of the reactive routing protocol AODV. SAR introduces a new security metric in the route discovery and maintenance operations that allows the node to capture and enforce cooperative trust relationships.

In SAR, the source broadcasts a trust level to its neighbors using the Route Request (RREQ). Then, the intermediate nodes can either rebroadcast this RREQ packet or drop it. The intermediate node only rebroadcast if the node itself has the required security level. The same applies to the Route REPly (RREP) packets. In case of receiving multiple RREP packets, the source picks the path of the first RREP packet [4]. Consequently, the path between the source and destination may not be the shortest but it is secure enough.

The main challenge of this approach is the definition of the trust level [10]. Security level could be derived from the organization hierarchy, such as ranks in the army or a company. Furthermore, to achieve secure routing, SAR needs other mechanisms to ensure the messages integrity and the identity of the nodes [11].

The main drawback of this approach is that malicious nodes may not adhere to the protocol specification. For example, if malicious node does not satisfy the security level required in the RREQ packet decided to rebroadcasts the RREQ packet instead of dropping it. In such case, according to the sender's requirements, the arrival of RREQ at the destination node does not necessarily guarantee that the traversed path is secure. The authors of [4] did not provide a mechanism to prevent such behavior.

SRP [12] is secure routing protocol that is based on the reactive routing protocol DSR. SRP requires a Security Association (SA) between the source and the destination nodes. SA is used to authenticate both RREQ and RREP packets through using Message Authentication Code (MAC). In SRP, the source initiates a RREQ packet that contains request sequence number and random request ID. Those identifiers are later used to calculate the MAC with the shared secret key (KS, T) [12]. In addition, the route request packet stores the identities of the traversed intermediate nodes.

To limit flooding in SRP, intermediate nodes keep track of the RREQ forwarding rate for each neighbor. It rebroadcast RREQ for neighbors with smallest rate. However, this approach may yield another type of attacks [13]. For example, SRP reward selfish nodes by prioritizing their requests over other active non-malicious nodes. SRP is also vulnerable to DOS, tunneling and wormhole attacks [13].

In [14], the authors proposed on demand Authenticated Routing for Ad hoc Networks (ARAN) protocol. ARAN uses certificates to ensure authentication, message integrity, and non-repudiation of routing messages. ARAN is developed on top of AODV. The source node broadcasts a signed RREQ packet that is authenticated by all intermediate nodes from the source to destination.

ARAN uses a trusted third party with public key that is known to all nodes. The third trusted party signs the certificate for all nodes in the network. The certificate contains the node's IP address, its public key, and time stamp of the certificate issuing and expiring times [14].

The RREQ packets are verified and signed by all its recipients. Finally, the destination node replies to the first not taking into account path optimality. Moreover, in the case of route maintenance ARAN applies the same procedure. Route Error (RERR) packets are signed and verified similar to the RREQ and RREP packets.

Although ARAN prevents spoofing attacks, however, it is vulnerable to the wormhole attack [10]. DoS attacks [15], and the tunneling attacks [16]. Furthermore, a considerable network overhead is injected in the network as a result of certifying all route request packets.

The authors in [15] proposed the Aridane protocol that is based on DSR reactive routing protocol. Ariadne signs routing messages using digital signatures. To identify the sender, MAC is appended to the packet.

Ariadne protocol specifies a mechanism for securing route maintenance operation; it ensures the validity of route error messages for the broken links in MANETs.

Ariadne is vulnerable to the invisible node attack [16]. Furthermore, Aridane protocol is vulnerable to other fabrication attacks, which cause the construction of nonexistent routes [10]. To overcome the shortcomings of these, endairA [9] has been proposed. endairA is similar to Aridane. The main difference is that endairA signs RREP packets instead of the RREQ ones.

In endariA [10], each node (namely node A) monitors its successor (namely node B) in the source route and checks whether node B forwards to node C all packets.

SAODV [8] is a secure routing protocol developed on top of the reactive protocol AODV. In SAODV there is a central key management unit that provides each node with a public key. Moreover, for message authentication, it uses digital signatures and it uses hash chains to protect hop counts. Once a source node has a RREQ message, it uses the hash function to calculate the top hash value, once a node receives the RREQ message; it verifies the hop count and the signature before updating its route table. Once the destination receives the RREQ message, it prepares and signs a RREP message. In case of like failure, a RERR message is generated as in the original AODV. The RERR messages are secured with digital signatures. SAODV is vulnerable to DOS, tunneling and wormhole attacks [13].

### *B. Schemes for Detecting Malicious Nodes*

The authors in [17] proposed a novel trust computation and management system, called TOMS. The proposed model introduces the concept of community, in which a node that is a central and all its one-hop neighbors establish a community. The authentication between the central node and its neighbors is achieved through cryptographic techniques. Once a new node joins a community, the central node assigns an initial trust value to this new node. The trust value depends on two factors. First, the time that node $n_i$ dwell in another community. Second, the past activity record of the node.

TOMS employs technique (called Trust Assistant Policy (TAP)) to provide an efficient mechanism for the central nodes in evaluating its neighboring nodes' trusts. Furthermore, TAP is used to detect the central nodes that are malicious.

The authors in [18] enhance the security of DSR protocol by enhancing the trust-based route selection mechanism. The selection of the best route is based on two scenarios. In the first scenario, the average of the trust values for all nodes in a route is computed. In the second scenario, it estimates the nodes' trust values according to the average of the past experiences values. Each node in the modified DSR protocol consists of five modules.

The authors in [19] propose a protocol that calculates the reputation of a node by observing the node's behavior, the observed value is later altered based on additional observations from other nodes. The estimated reputation values are exchanged among all nodes in the network. The main goal of this protocol is to detect selfish nodes. In this protocol, node *i* observes the behavior of its neighboring node *j* and stores the acquired information into a local table called Neighbor Reputation Table (NRT). The NRT contains information regarding one hop neighbor nodes. Moreover, a Global Reputation Table (GRT) is also used to stores information about all nodes in the network.

The authors in [10] proposed an approach that monitors, detects, and isolates selfish nodes. It deals with both directed (unicast) and broadcast packets.

## III. PROPOSED SCHEME

The proposed protocol, named Trust Scheme, aims to detect malicious nodes in MANETs and then avoids transmitting messages through them. Therefore, nodes transmit data via trusted paths to enhance network performance. The Trust Scheme evaluates the behavior of all nodes by establishing a trust value for each node in the network that represents the trustworthiness of each one. It calculates the trust value of a node by directly observing the behavior of the node and then passing this value with other observations from other nodes in the network. The behavior of the neighboring nodes is used as an indication to distinguish between normal nodes (nodes that act as expected) and malicious nodes.

Each node in the network observes the behavior of its neighbors. It observes node's mobility, number of neighbors each node has, number of packets generated and forwarded by the neighboring nodes, and the past activity of the node. Those parameters are then used to determine which nodes are misbehaving in the network. Then, the observer node builds a table called Local Trust Table (LTT) that records a local trust value for each neighboring node estimated from these observations. We note that only using the trust value in LTT may not be an accurate measure to decide that a node is a malicious one. Therefore, each node constructs a Neighbor Trust Table (NTT) that stores the trust value for all nodes not only the neighbors but also the far ones.

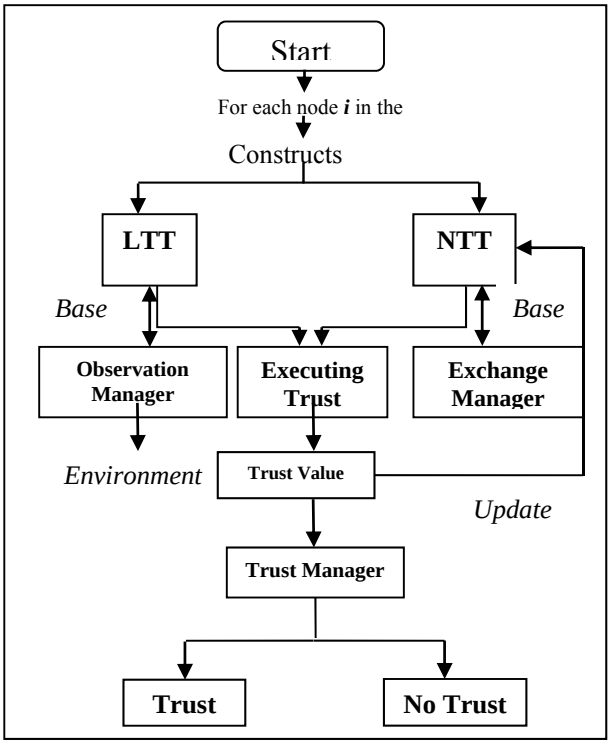



Figure 1: Trust Scheme Framework

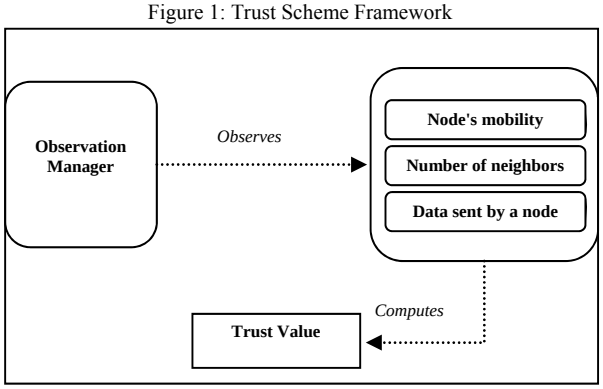



Figure 2: Observation Manager for Node i

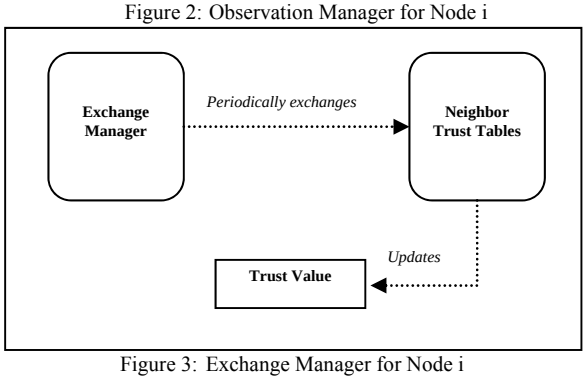



Figure 3: Exchange Manager for Node i

For a given node *i*, all nodes participate in computing its trust values in NTT by exchanging NTTs between neighbor nodes. Figures 1, 2 and 3 present the general framework of the Trust Scheme. Each node in this scheme consists of three modules: Observation Manager, Exchange Manager and Trust Manager.

The following sections discuss how to estimate the local and global trust values.

### *A. Local Trust Table (LTT)*

As mentioned above, each node in the network constructs a local trust table that keeps track of the information observed directly about each neighbor in order to calculate the trust value for a node. The LTT for a node contains an entry only for a node's neighbors.

Every node calculates the local trust value for its neighbors as shown in the following formula:

$$\text{local}_{\text{trust}} = \text{a1*mobility} + \text{a2*nbrs} + \text{a3*data}_{\text{sent}} + \alpha 4^{*}\,\text{old}_{\text{trust}} \quad (1)$$

Where, $a1, a2, a3$ *and* $a4$ are tuning constants that are determined during simulation. The sum of these constants equals 1. And, *nbrs* is the number of neighbors each node has, $data_{sent}$ is the forwarded and generated packets sent by the observed node and the $old_{trust}$ is the last trust value of the node; it presents the past activity of a node. Mobility is measured based on the number of changes in the one-hop neighborhood of a node. Figure 4 shows a simple scenario. Node A directly observes the behavior of its one-hop neighbors; B, C and D. It stores the values of the observed parameters in its local trust table.

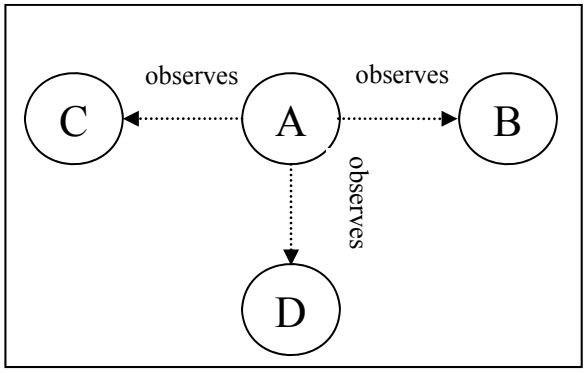



Figure 4: A simple scenario

The structure of LTT for node A is shown in Table 1. It includes entries for nodes B, C and D, and their behavior values. Node A computes a local trust value for each node based on the stored values as described in Formula 1.

TABLE 1: LTT FOR NODE A

| Node | *Mobility* | *nbrs* | $data_{sent}$ | $old_{trust}$ | $local_{trust}$ |
|---|---|---|---|---|---|
| **B** | 0.5 | 0.8 | 0.34 | 1.3 | 2.94 |
| **C** | 0.4 | 0.6 | 0.56 | 1.64 | 3.2 |
| **D** | 0.3 | 0.2 | 0.18 | 1.05 | 1.73 |

### *B. Neighbor Trust Table (NTT)*

LTT stores only information about one-hop nodes. Every node also has a Neighbor Trust Table (NTT) that stores the trust values for all nodes in the network. Table 2 shows the structure of NTT for node A. Its structure is simple. It contains entries for all nodes in the network and their global trust values. B, C and D are direct neighbors for node A. E and F are not neighbors. So,

NTT do not have local trust values for them. This table distributed among neighbors in order to compute and update the trust value for each node. If the node is a direct neighbor, its trust value is calculated using the local trust value and the remote information that is obtained from neighbor nodes.

The new trust value that is stored in NTT will be:

$$new\ trust = wl * local_{trust} + wr * remote \quad (2)$$

Where,

$wl$ and $wr$ are weights and $local_{trust}$ has higher weight ($wl$).

For far nodes that are not neighbors for a node, the new trust value will be:

$$new\ trust = remote \quad (3)$$

The remote information is:

$$remote = w1 * T_{est} + w2 * T_{rec}$$

Where $T_{est}$ is the value stored in NTT, and $T_{rec}$ is the recommendation ($local_{trust}$) received from a neighbor different from the owner of the table.

TABLE 2: NTT FOR NODE A

| Node | Local trust | Test | Trec | Globaltrust |
|---|---|---|---|---|
| B | 2.94 | 0.5 | 1.02 | 0.65 |
| C | 3.2 | 0.9 | 1.4 | 0.83 |
| D | 1.73 | 0.87 | 0.34 | 0.380 |
| E | X | 1.2 | 2.3 | 0.81 |
| F | X | 0.45 | 1.67 | 0.546 |

### C. Exchanging the NTT

As mentioned above, every node of the network constructs an local trust table (LTT) that keep track of the trust values for one hop neighbors, and a Neighbor Trust Table (NTT) that keep tracks of trust relation with neighbors and far nodes. The information in the NTT is exchanged with neighbors. The Trust Scheme allows scattering the trust values for all nodes throughout the network. And thus, all nodes participate in deciding which nodes considered as malicious not only one node has the decision.

The information in the NTT is not broadcasted over the network. A node sends its own table only to one hop neighbors. Every node receives an updated version of the table from its neighbors, computes new trust values, and then it sends the updated NTT to its neighbors.

## IV. SIMULATION RESULTS

In order to evaluate the performance of the Trust Scheme, the Network Simulator (NS2) version 2.32 with wireless extension is used. NS-2 is a discrete event simulator targeted at networking research [20]. It is one of the most popular simulation packages used in the literature because of its availability (free license), and possibility to implement and test new protocols and applications [19]. NS-2 is a commonly used package to simulate several schemes (such as routing and multicast protocols) in wired and wireless domains as it supports several networking protocols and standards [19]. Also, it implements modules for wireless stations, which can move in a 2D environment. To simulate the network, an OTcl script is used to initiates the event scheduler, and sets up the network topology using the network objects.

### A. Simulation Environment

In the simulation experiments, the underlying scheme is used with DSR. Moreover, a network with 1000m x 800m area and 25, 50 and 100 mobile nodes was simulated. The simulation time is 1000 seconds. The mobile nodes move within the network space according to the Random Waypoint model. The communication patterns used are 6 Constants Bit Rate (CBR) connections with a data rate of 10 packets per second. 10%, 20%, 30% and 40% of the total number of nodes were chosen randomly as malicious nodes. Those malicious nodes drop, alter and forward packets.

### B. Performance Metrics

For evaluating the Trust Scheme against the standard DSR and RP [19], three commonly used metrics are used to evaluate the performance of the three protocols [21], and [22]: Packet Delivery Ratio (PDR), Routing Overhead, and Average Delay. Moreover we will use the True vs. False detected malicious nodes to show the strength of the proposed scheme.

TABLE 3: SIMULATION PARAMETERS

| | |
|---|---|
| Simulator | NS-2 |
| Simulation duration | 1000 seconds |
| Simulation area | 1000m x 800m |
| Number of nodes | 25, 50, 100 nodes |
| Transmission range | 250 m |
| Movement model | Random waypoint |
| Traffic type | CBR |
| Data size | 512 bytes |
| Packet rate | 10 pkt/sec |
| Maximum speed | 5 m/s |
| Pause time | 60 s |

### C. Simulation Results

The Trust Scheme is studied using the above defined metrics. The standard DSR and RP are also evaluated. The standard DSR does not have a mechanism to detect and avoid malicious nodes. It considers all nodes as trusted. RP only detects selfish nodes that do not participate in transmitting packets in order to save their energy. It does not detect the other types of malicious nodes. All the graphs are plotted from the data averaged from the 30 runs.

As we discussed when defining the performance metrics, packet delivery ratio is measured by dividing the total number of data packets received, by the total number of data packets originated by the sources. Packet delivery ratio presents an important metric for performance evaluation of a mobile ad hoc routing protocol because the number of data packets successfully delivered depends mainly on path availability, which in turn depends on how effective the underlying routing algorithm is in a mobile scenario.

Figures 5, 6 and 7 plot the packet delivery ratio of the Trust Scheme, RP and the standard DSR in the presence of malicious nodes with different number of mobile nodes. A higher value of PDR indicates that most of the packets are being delivered and is a good indicator of the protocol performance as well.

The graphs demonstrate that the Trust Scheme always performs better than the standard DSR and RP in the three scenarios. This is because in the Trust Scheme, the malicious nodes are detected and hence avoided. This means that only trusted paths are used for transmitting packets. Thus, most sent packets delivered to the desired destination. In RP, only the selfish nodes are detected, so it performs lesser than the Trust Scheme and better than the standard DSR.

Figures 8, 9 and 10 depict the routing overhead for the three protocols that are generated by the malicious nodes. They demonstrate that the routing overhead significantly increase as the percentage of malicious nodes increases in the network. The standard DSR has the highest overhead.

Trust Scheme has an almost equivalent routing overhead when compared with RP but at less number of mobile nodes. However, Trust Scheme still performs better than RP for the routing overhead when there are more mobile nodes and hence more number of malicious nodes. The reason is that, in this scheme, the routing packets are reduced since nodes do not rebroadcast the routing packets received from the malicious nodes, they immediately drop any routing packets received from malicious nodes and also the other two protocols will keep flooding route discovery to all neighbors. This brings many transmissions of control packets.

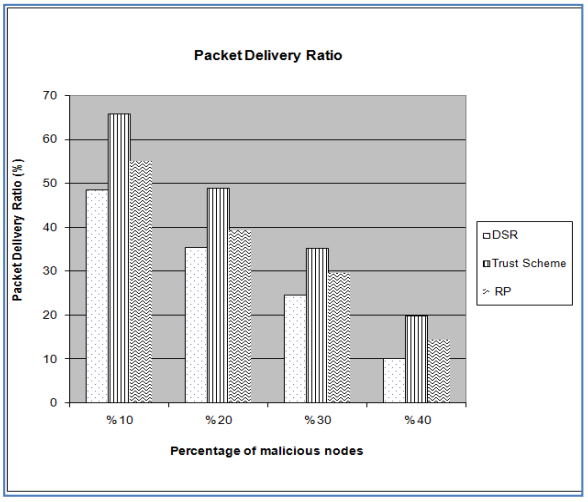



Figure 5: Packet delivery ratio for 25 mobile nodes

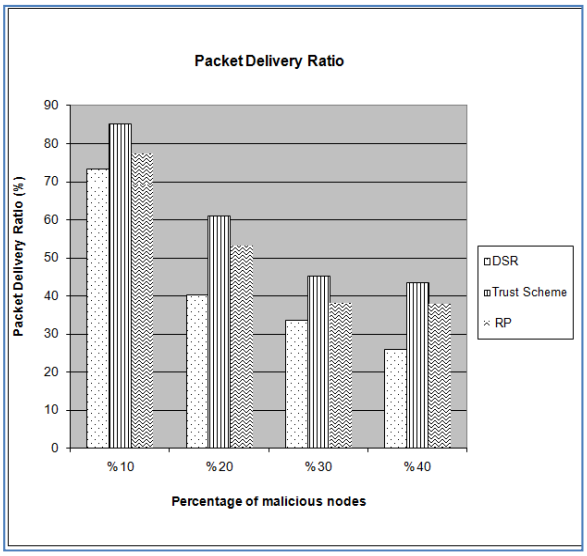



Figure 6: Packet delivery ratio for 50 mobile nodes

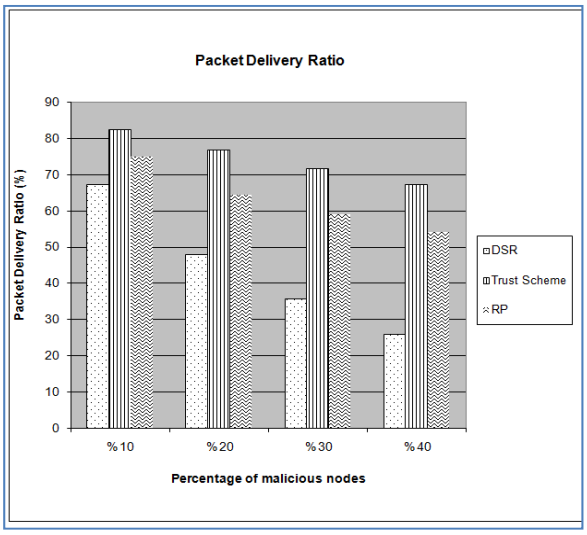



Figure 7: Packet delivery ratio for 100 mobile nodes

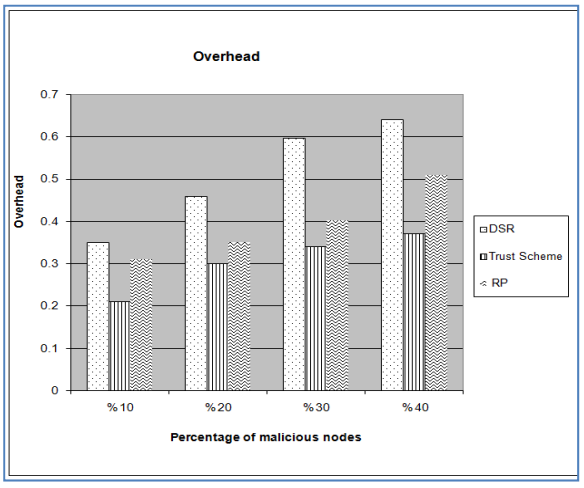



Figure 8: Overhead for 25 mobile nodes

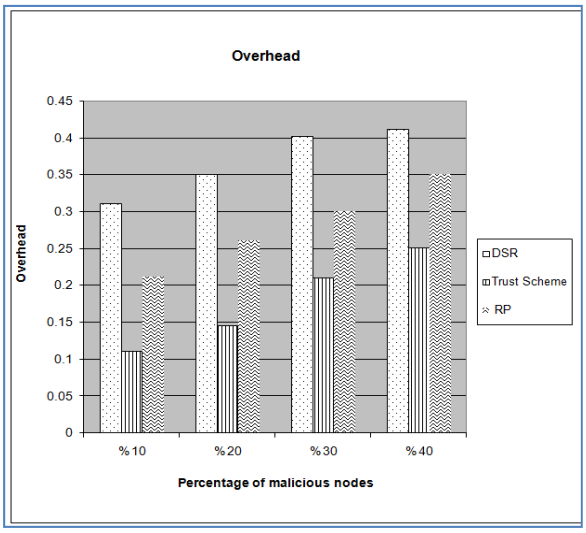



Figure 9: Overhead for 50 mobile nodes

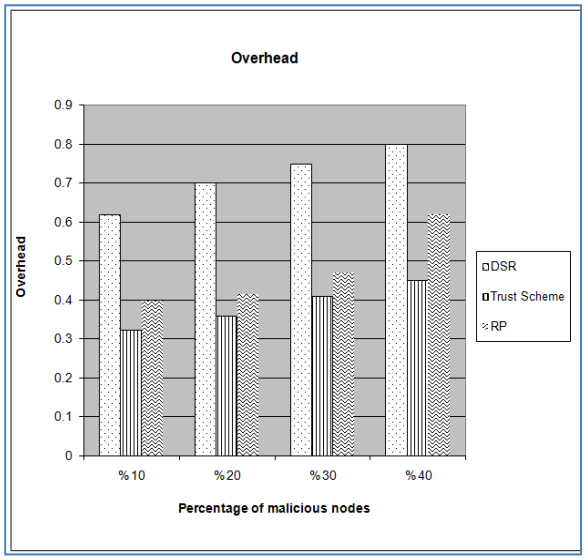



Figure 10: Overhead for 100 mobile nodes

Figures 11-13, show the simulation results of the performance metric, average delay. As can be seen from the figures, the trust scheme always outperforms the other two protocols in terms of the average delay.

As the routing overhead increases the delay experienced by data packets increases. However, as seen in Figures 8 to 10, it is evident the trust schemes achieves lower overhead and as a results it achieves lower packet delivery times.

Now, we present the behavior of Trust Scheme and RP in terms of the true number of detected malicious nodes and the false number of detected malicious nodes.

Figures 14-16 show the true number of detected malicious nodes. As shown, Trust Scheme was successful in detecting more number of malicious nodes in the three scenarios.

On the other hand, Figures 17-19 show the number of false detected malicious nodes. Again, Trust Scheme outperforms RP. It detects less number of false malicious nodes.

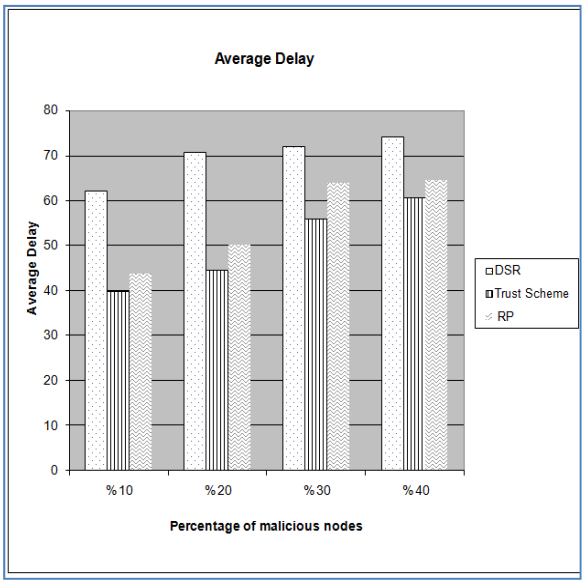



Figure 11: Average Delay for 25 mobile nodes

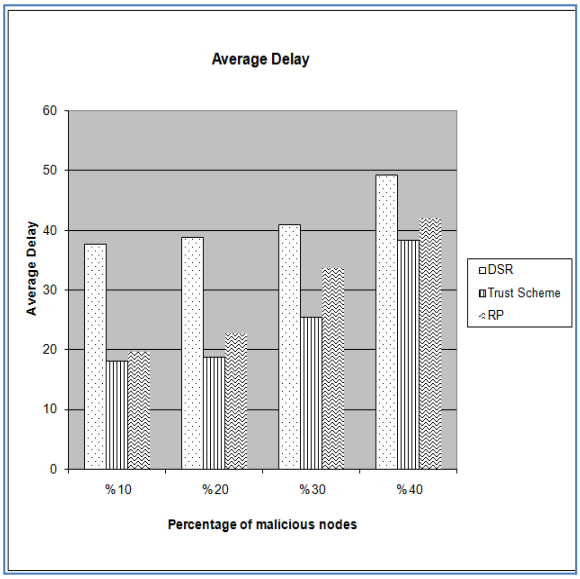



Figure 12: Average Delay for 50 mobile nodes

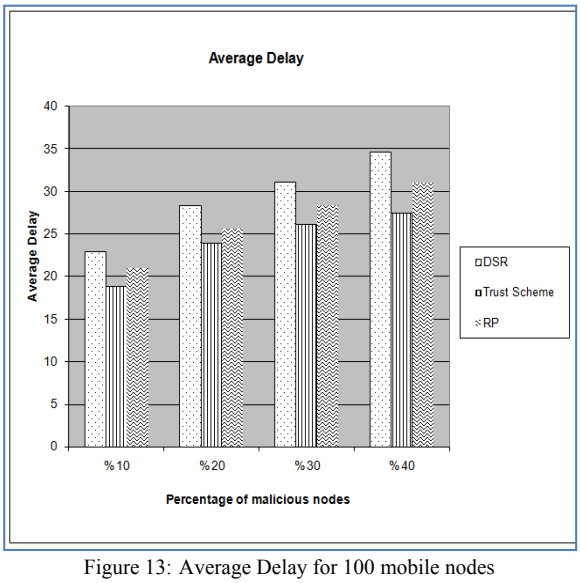



Figure 13: Average Delay for 100 mobile nodes

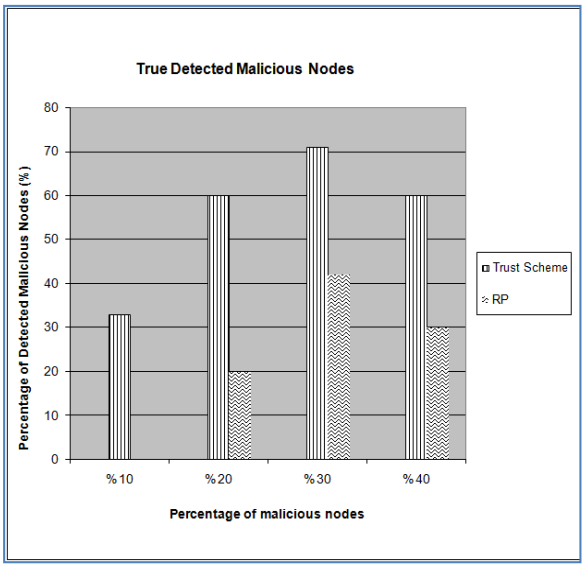



Figure 14: True detected malicious nodes for 25 mobile nodes

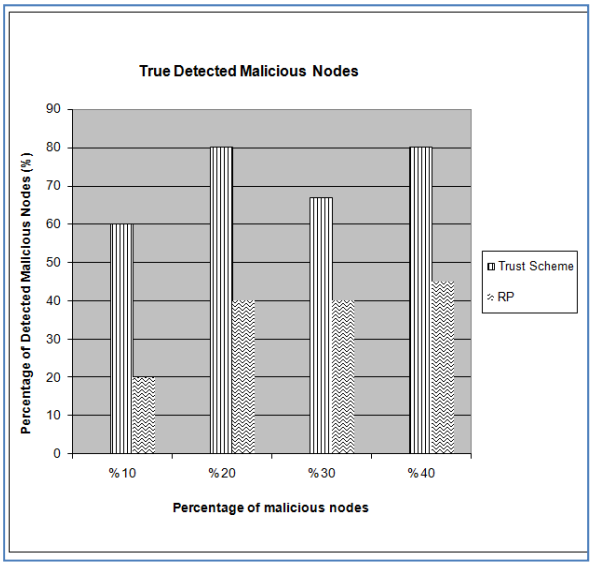



Figure 15: True detected malicious nodes for 50 mobile nodes

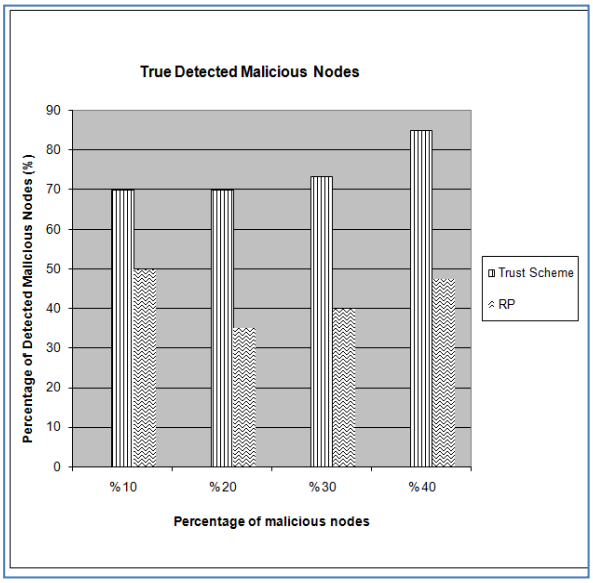



Figure 16: True detected malicious nodes for 100 mobile nodes

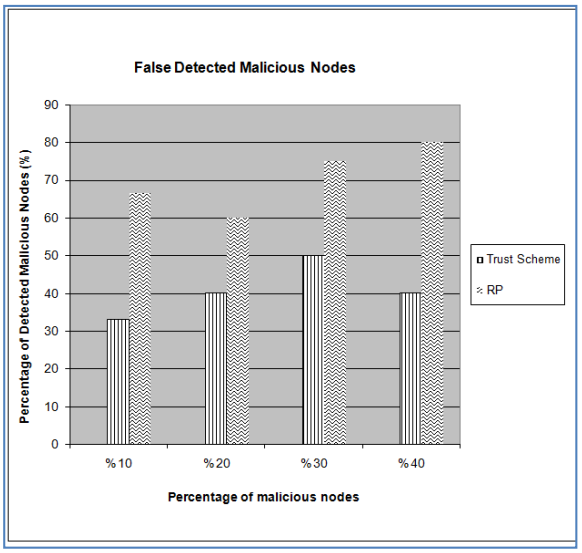



Figure 17: False detected malicious nodes for 25 mobile nodes

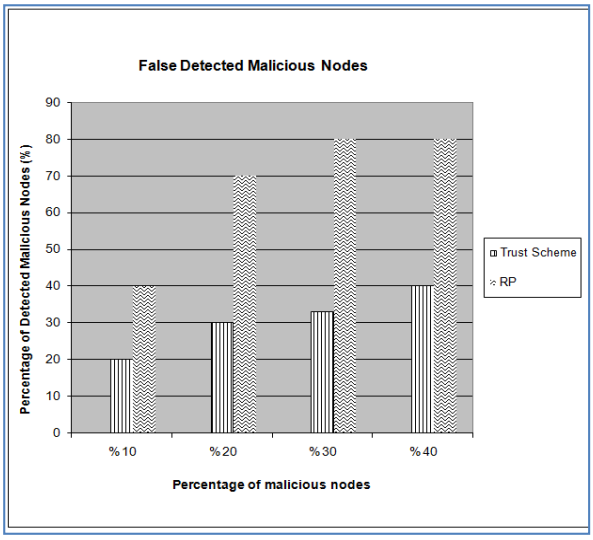



Figure 18: False detected malicious nodes for 50 mobile nodes

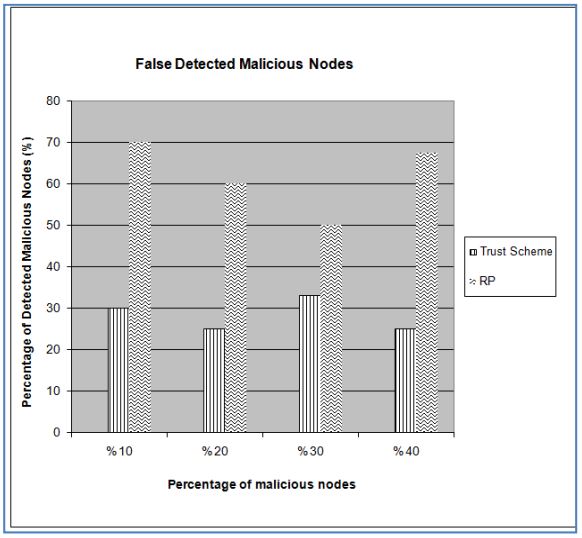



Figure 19: False detected malicious nodes for 100 mobile nodes

## V. Conclusion and Future Work

Securing mobile ad hoc networks is a challenging task. In this paper, we have proposed a new approach that detects the misbehaving nodes that intend to jeopardize the network resources and affects its performance. This approach proposes observing the behavior of mobile nodes depending on different factors. Based on these factors, each node in the network can recognize the malicious nodes and thus prevent them from participating in the communication.

The performance of the proposed scheme is evaluated using NS-2 simulator and compared the performance with the standard DSR and Reputation protocol. In the standard DSR all nodes participate in the communication; it does not differentiate between malicious and normal nodes. On the other hand, RP only detects and avoids selfish nodes that do not always participate in forwarding packets via the network in order to save its energy.

The Trust Scheme outperforms the other two protocols and improves the network efficiency. In the presence of variable number of malicious nodes and different numbers of mobile nodes, Trust Scheme outperforms DSR and RP in all performance metrics. The routing overhead is considerably low, resulting in higher packet delivery ratio as well as less delay in packet delivery.

Research in the area of MANETs security is still active. Further work is needed to enhance the performance of the secure routing protocols for MANETs.

The evaluation of Trust Scheme and the other two protocols, DSR and RP, discussed in this work with some more performance metrics must be considered as future research work. Also, different scenarios can be used such as evaluating the same protocols using different mobility speed, and different pause time. Moreover, the Trust Scheme can be extended to include observing more factors such as the accuracy of the received packets. The observer node observes the correctness of the packets received from a specific node and thus detecting the misbehaving nodes based on the number of received and inaccurate packets.

## References


[1] Gowsiga, S. and Manavalasundaram, V. An Efficient and Secure Route Discovery for Mobile Ad Hoc Networks, Proceedings of the International Conference, Computational Systems and Communication Technology, 2010

[2] Hu, Y., Johnson, D. and Maltz, D. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR), <draft-ietf-manet-dsr-09.txt>, 2003.

[3] Belding-Royer, E., Das, S., and Perkins, C. Ad hoc On-Demand Distance Vector (AODV) Routing, IETF Internet Draft, www.ietf.org, 1997.

[4] Kravets, R., Naldurg, P. and Yi, S. Security-aware Ad Hoc Routing for Wireless Networks, In The ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC01), Long Beach, CA, 2001.

[5] Batra, S., Goyal, P. and Singh, A. A Literature Review of Security Attack in Mobile Ad-hoc Networks, International Journal of Computer Applications, 11-15, 2010.

[6] Inomata, A., Mambo, M., Md, S., Okamoto, T. and Rahman, M. Anonymous Secure Communication in Wireless Mobile Ad-hoc, In Proceedings of the First International Conference on Ubiquitous Convergence Technology, 131-140, 2006.

[7] Haas, Z. and Zhou, L. Securing Ad Hoc Networks, IEEE Network Magazine, 24-30, 1999.

[8] Hu, Y. and Perrig, A. A Survey of Secure Wireless Ad Hoc Routing, IEEE Security & Privacy, 28-39, 2004.

[9] Lu., Q. Vulnerability of Wireless Routing Protocols, Technical Report, University of Massachusetts Amherst, 2002.

[10] Bouamama, M., Djenouri, D., Jones, D., Merabti, M. and Mahmoudi, O. On Securing MANET Routing Protocol Against Control Packet Dropping, IEEE International Conference on Pervasive Services, 2007.

[11] Hongbo Z. A Survey on Routing Protocols in MANETs, Technical Report MSU-CSE-03-08, 2003.

[12] Haas, Z. and Papadimitratos, P. Secure Routing for Mobile Ad Hoc Networks, In The SCS Communication Networks and Distributed Systems Modeling and Simulation Conference CNDS02, San Antonio, Texas, 2002.

[13] Argyroudis, P. and O'Mahony, D. Secure Routing for Mobile Ad hoc Networks, IEEE Communications Surveys & Tutorials, 2-21, 2005.

[14] Dahill, B., Levine, B., Royer, E. and Shields, C. Aran: A Secure Routing Protocol for Ad Hoc Networks, Technical Report UMass Tech Report 02-32, 2002.

[15] Hu, Y., Johnson, D. and Perrig, A. Ariadne: A Secure On-demand Routing Protocol for Ad Hoc Networks. In The 8th annual international conference on Mobile computing and networking MobiCom '02, 12–23, 2002.

[16] Ramachandran, P. and Yasinsac, A. Limitations of On Demand Secure Routing Protocols, Proceedings of the 2003 IEEE Workshop on Information Assurance, 2003.

[17] Boukerche, A. and Ren Y. Modeling and Managing the Trust for Wireless and Mobile Ad hoc Networks, IEEE International Conference on Communications, 2129 – 2133, 2008.

[18] Alsaadi, M. and Qian, Y. Performance Study of a Secure Routing Protocol in Wireless Mobile Ad Hoc Networks, 2nd International Symposium on Wireless Pervasive Computing, IEEE Computer Society, 425-430, 2007.

[19] Bella, G., Costantino, G. and Riccobene, S. Evaluating the Device Reputation through Full Observation in MANETs, Journal of Information Assurance and Security, 458-465, 2009.

[20] The Network Simulator NS-2, http://www.isi.edu/nsnam/ns/, accessed 5 August 2009.

[21] Liu, J., Liu, Z. and Sangi, A. Performance Comparison of Single and Multi-Path Routing Protocol in MANET with Selfish Behaviors, World Academy of Science, Engineering and Technology, 2010.

[22] Chen, Y. and Nasser, N. Enhanced Intrusion Detection System for Discovering Malicious Nodes in Mobile Ad hoc Networks, IEEE Communications Society subject matter experts for publication in the ICC 2007 proceedings, 2007.

[23] S.Kannan, T. Maragatham, s. Karthik, V.P Arunachalam, "A Study of attacks, Attack Detection and Prevention Methods in Proactive and Reactive Routing Protocols", In Medwell journals, volume 5, page No.: 178-183, 2011.

[24] S. Umang, B.V.R. Reddy, M.N. Hoda, "Enhanced intrusion detection system for malicious node detection in ad hoc routing protocols using minimal energy consumption", In ITE journals, volume 4, page(s): 2084 - 2094, 2010.

[25] G. S. Mamatha, Dr. S. C. Sharma, "A New Combination Approach To Secure MANETS Against Attacks", In International Journal of Wireless & Mobile Networks (IJWMN), volume 2, page(s):1-10, 2010

**Yaser Khamayseh** was born in Irbid/ Jordan in 1977. He received his Bachelor degree in computer science from Yarmouk University, Irbid, Jordan, in 1998. He finished his master's in computer science at the University of New Brunswick, Canada in 2001. And he finished his PhD in computer science at University of Alberta, Canada in 2007.

He is an assistant professor of computer science at Jordan University of Science and Technology since 2007. He has More than 12 years of experience in research and teaching in the field of data communication and computer networks. His research interests include simulation and modeling, wireless networks, performance evaluation, evolutionary computation, and image processing.

Dr. Khamayseh is member of IEEE and has received several awards. He is a member of technical programs of several journals and conferences. He served as a reviewer for several conferences and Journals

**Muneer Masadeh Bani Yassein** received his B.Sc. degree in Computing Science and Mathematics from Yarmouk University, Jordan in 1985 and M. Sc. in Computer Science, from Al Al-bayt, University, Jordan in 2001. And PhD degrees in Computer Science from the University of Glasgow, U.K., in 2007, He is currently an assistant professor in the Department of Computer science at Jordan University of Science and Technology (JUST), His research include the development/analysis the performance probabilistic flooding behaviors in MANET optimizations and the refinement of routing algorithms for mobile device communications in heterogeneous network environments.

# Wake-Up-Receiver Concepts - Capabilities and Limitations -

Matthias Vodel, Mirko Caspar and Wolfram Hardt
Chemnitz University of Technology, Chemnitz, Germany
Email: {vodel, mica, hardt}@cs.tu-chemnitz.de

***Abstract*— A promising idea for optimising the power consumption of mobile communication devices represents the usage of an additional ultra-low-power receiver unit, which is able to control the main transceiver in order to reduce the standby power consumption of the overall system. Such a *Wake-Up-Receiver (WuRx)* unit senses the medium and switches on the communication interfaces in case of an external request. Otherwise, all system components for the network communication are completely switched off. Especially in the domain of resource-limited and embedded devices, WuRx technologies enable novel communication paradigms.**

**But on the application layer, not all scenarios allow the efficient usage of such WuRx technologies. Dependent on environmental parameters, technological limitations and conceptual requirements, different strategies are necessary to ensure energy-efficient system operation.**

**In this article, we present a critical analysis of the capabilities and the conceptual limitations of WuRx approaches. Therefore, we identify critical parameters for WuRx concepts, which limit the efficiency in real world scenarios. Our goal is to classify sufficient fields of application. Furthermore we evaluate the influences of these parameters on the system behaviour. In addition, we introduce heterogeneous energy harvesting approaches as an efficient way for the system optimisation. The proposed technologies are capable to be integrated into small-sized wireless sensor platforms and prolong the system uptime significantly.**

**The presented simulation results are focusing on actual smart metering scenarios and wireless sensor networks. Based on these measurements, we were able to apply further optimisation steps within the system configuration on the application layer. In this context, we focus on application-specific key issues, like the trade-off between measurement quality and quantity, the usage of data buffering approaches and QoS capabilities.**

***Index Terms*— Wake-Up-Receiver, WuRx Technology, Energy-Efficiency, Wireless Sensor Networks, Energy Harvesting, Communication Strategies, Asynchronous, Resource-limited Systems, Embedded Systems, Critical Parameters, Routing, Topology Optimisation**

## I. Introduction

Wireless communication is omnipresent in all areas of our everyday life. Due to the technological enhancements, the device costs shrink and at the same time the hardware capabilities increase substantially. At the same time, limited energy resources still represent one the most challenging bottlenecks within mobile devices. In order to provide a sufficient battery runtime for the mobile devices, energy-efficient hardware components and operational concepts are essential. Especially in the field of mobile devices, the integrated communication interfaces and the interface management consume a huge part of the available energy resources.

*Figure 1* illustrates the problem. The exemplary measurements are based on wireless sensor network node with an ultra low power TI MSP430 microcontroller [1]. The diagram visualises the microcontroller power consumption during the different operational modes during a sensor communication scenario.

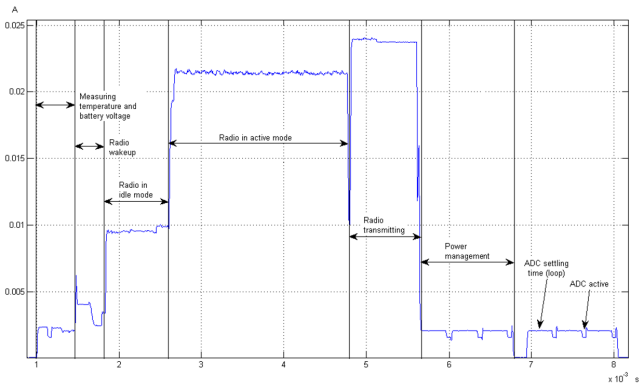



Figure 1. MSP430 working load in different operational modes, starting with the sensor measurement, the radio initialisation, radio listen mode (most power consuming), the transmission and the power management mode for preparing the deep sleep mode [2].

In order to solve or bypass the problem, the research focus on the optimisation of the transmission characteristics on the different communication layers [3][4]. Accordingly, during the last decade, engineers try to find an application-specific trade-off between the network latency and the average power consumption of the transceiver. In contrast to these traditional optimisation approaches, wake-up-receivers (WuRx) pursue another concept and enable a different communication paradigm.

If we take a look on standard communication scenarios, most of the time, the transceiver is waiting for external requests or incoming data. During that time, the listen mode consumes a lot of energy. Especially in the field of low-power communication standards, like *ZigBee*, *Z-Wave* or *Bluetooth*, listening and receiving data is more power consuming in comparison to the required energy for sending data. To eliminate this kind of conflict, *Wake-up Receivers* (*WuRx*) represent a dedicated, energy-efficient


doi:10.4304/jnw.7.1.126-134

receiver unit, which replaces the main transceiver during listening periods. The WuRx is able to detect special wake-up signals from its environment and accordingly wakes up the main transceiver only on demand. The WuRx concept is illustrated in *figure 2* and *3*. In many cases, engineers only have to combine an off-the-shelf transceiver with an additional WuRx to realise a more efficient communication platform.

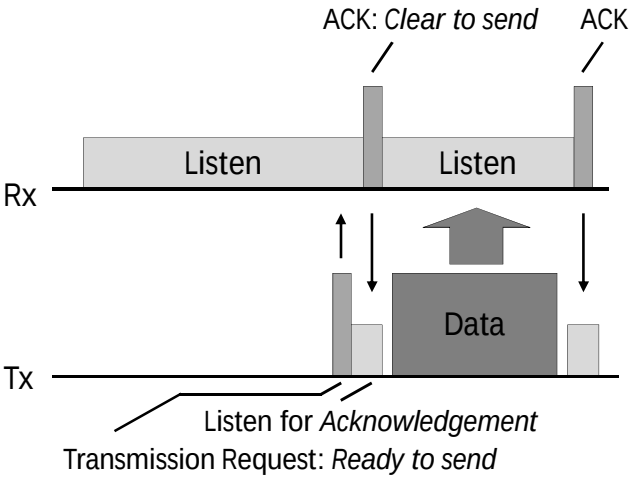



Figure 2. Conventional communication process between transmitter (*Tx*) and receiver (*Rx*). The receiver has to sense the medium. Dependent on the transmission parameters and the characteristics/duty cycle of the used radio standard, this listening mode consumes a lot of energy.

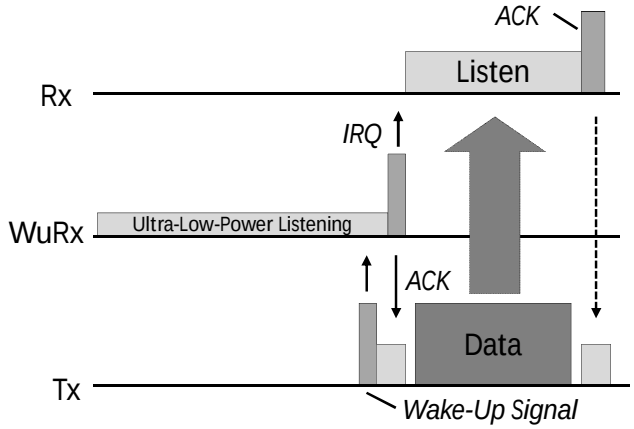



Figure 3. Wake-up receiver communication. The main network interface is waiting in a standby mode and will be activated by the WuRx on demand. The WuRx is designed as an ultra-low-power component, which only has to detect a predefined, coded wake-up signal from its environment.

Especially in the field of long term monitoring scenarios, like smart metering or next generation wireless sensor networks, such approaches promise enormous capabilities for energy-saving approaches [5][6]. Complex synchronisation procedures on both hardware or software level are not required anymore. Based on the WuRx technology, we change the way of communication from a synchronised, timer-based transmission to an asynchronous operation with wake-ups on demand (shown in *figure 4*).

Accordingly, we switch from a protocol-based duty-cycling to a WuRx-controlled duty-cycle. Thus, WuRx technologies provide powerful features for developing distributed, autonomous systems in an asynchronous application environment.

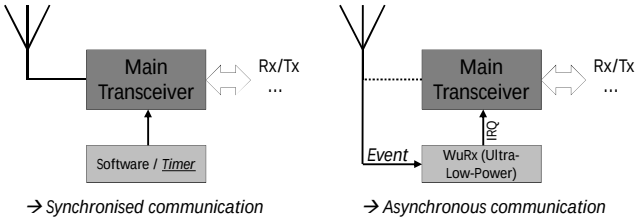



Figure 4. Different communication concepts with and without WuRx. A traditional application scenario on the left site requires any kind of time synchronisation in order to coordinate sleep-/transmission timings. The WuRx application on the right site does not need such a synchronisation technique. Special wake-up events activate the system on demand.

## II. Related Work

The research for WuRx technologies started several years ago with first approaches to modify off-the-shelf mobile devices. In [7], the authors upgrade a given handheld with a conventional low-power transceiver module, which operates as a wake-up unit for the main 802.11 WiFi interface. The power consumption of the additional receiver averages 7 mW in the receiving mode.

The work of [8] focuses on a radio-triggered hardware component to enable wake-up capabilities (see *Figure 5*). A primary aim for this approach was the avoidance of software-based synchronisation and scheduling techniques. Thus, the proposed operation concept creates asynchronous network behaviour in order to respond specific trigger events on demand. Now, the system is able to optimise the sleep and stand-by times without additional costs on the software level.

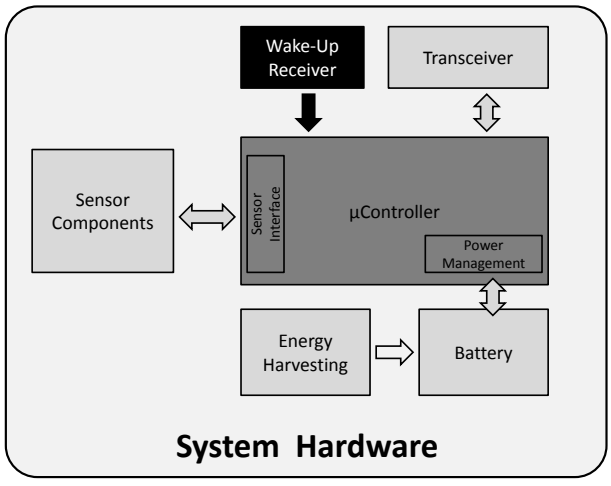



Figure 5. Typical system hardware architecture of an embedded system with a wake-up receiver unit.

Further research optimises different key parameters within the wake-up receiver units. This includes the power consumption, the frequency band [9] and the receiver

sensitivity [10]. Related off-the-shelf WuRx are operating in the ISM band at 2.4 GHz, 868 MHz as well as on the 125 KHz low frequency band for RFID-based applications. The power consumption ranges from 20 to 700 $\mu W$ for the normal listening mode. In [11], the authors present a 2 GHz WuRx with 52 $\mu W$ and -72 dBm sensitivity.

The problem of traditional, electronic WuRx technologies correlates with the respective system design and the required energy for each hardware component. To avoid these disadvantages, *nanett* [2] researches for an NEMS-based wake-up receiver approach (*Nanoelectromechanical System*). In contrast to related WuRx technologies, the introduced nanett project is based on a pure mechanical resonator, which generates the wake-up interrupt for the microcontroller and the main transceiver. The most important benefit of this approach is the operation without complex and energy-consuming electronic circuits. Accordingly, the goal is a power consumption of less than 10 $\mu W$ in the listening mode.

## III. System Optimisation

In order to prolong the system uptime for any kind of autonomous application scenario, several researchers try to combine WuRx technologies with additional energy harvesting components in the system architecture [12][13]. In general, we differ thermal, vibration and solar energy converter units, shown in *figure 6*.

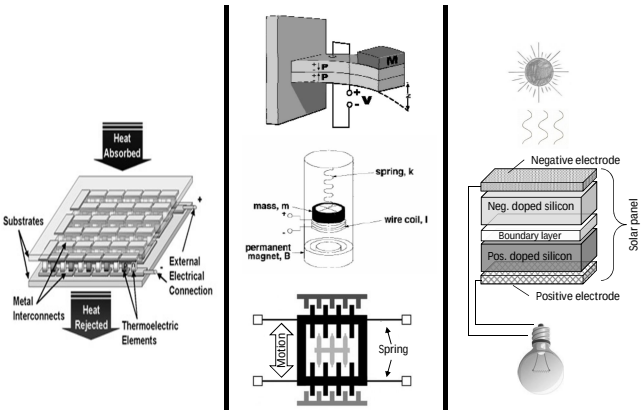



Figure 6. Typical energy harvesting technologies - Thermoelectric converter (left); Photovoltaic converter (right). Vibration-based converter elements (center) can be separated into: Piezoelectric harvesters (top); Electromagnetic harvesters (center); Electrostatic harvesters (bottom).

In this context, one challenge represents the efficient usage of the harvested energy as well as the stored energy resources (illustrated in *figure 7*). There are three possible energy paths. The first one from the harvester units directly to the application, another one from the harvesters to a given energy storage and the energy path from the available energy storages to the application. Dependent on the actual harvesting level, a power management unit has to decide about an optimal energy path. Furthermore, such information about the actual harvesting level are also helpful on the application layer. Based on these meta-information, the application is able to adapt its operational mode or the scheduling scheme for the data measuring as well as the communication tasks. In case of a high energy budget, the measuring interval will be increased. On the other side, in case of a critical energy level or minimal harvesting capabilities, the network communication will be reduced and the measured data sets are temporarily stored in a local buffer.

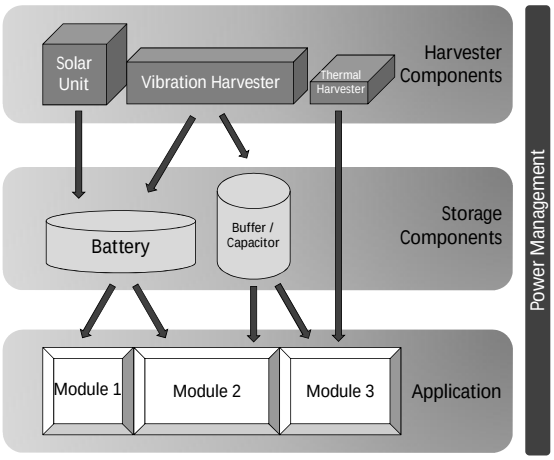



Figure 7. The power management is responsible for an efficient usage of the available energy resources and the harvesting components. In combination with an intelligent application design and modern WuRx technology, the system availability increases significantly.

## IV. Critical System Parameters

Based on the introduced wake-up receiver capabilities and the mentioned approaches for optimising the system design, we are able to identify and to define key parameters for a sufficient and efficient usage of WuRx capabilities [14].

### A. Operational modes on the application layer

The system specification on the application layer represents the most important and most critical parameter for any kind of WuRx technology. The WuRx is able to hold the system in a *ready-to-receive* mode. The power consumption of the main transceiver can be minimised. In consequence, only asynchronous and event-based application scenarios are capable for the usage of WuRx technologies. Thus, the entire, distributed system operates with predefined trigger-events for waking up the nodes. In more detail, we define two dedicated operational modes for a given distributed, WuRx-enabled system.

The first one represents a local and autonomous mode, in which all communication interfaces are disabled. Each node is sensing its environment with specific sensors components. If required, data can be stored within a local data buffer. In case of an emergency (e.g. system failure or critical battery level), each node is able to establish an ad hoc communication channel to its neighbourhood.

The second operational mode is responsible for the transmission of events or measured data. Accordingly, this mode includes all possible communication procedures like the network exploration, network optimisation, the

route path calculation or the exchange of basic system information. The mode is initialised by a wake-up signal, which will be broadcasted through the entire topology.

*Figure 8* illustrates these two modes and represents the basic operational concept of every asynchronous application scenario with WuRx capabilities.

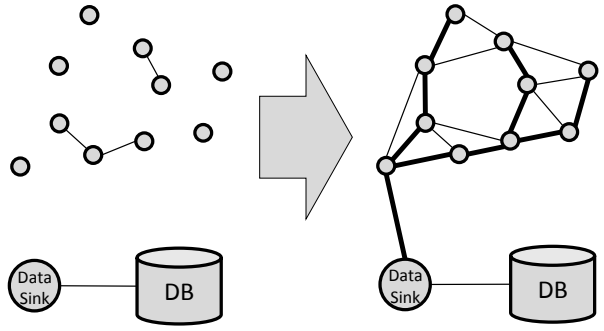



Figure 8. Two dedicated operational modes for an asynchronous, WuRx-enabled application scenario. On the left site the autonomous operational mode without network communication (excepting peer-to-peer transmission channels in case of an emergency). On the right side the communication / network mode (in this case to a predefined data sink with external database uplink). The initial network exploration provides a suboptimal topology (represented by the thin lines), the optimised network topology is illustrated by the fat lines.

During the runtime, the overall, distributed system has to manage different tasks. Dependent on the system specification, the task schedule and the mapping of each task to predefined system components is essential. In our WuRx scenario with two dedicated operational modes, the tasks are organised as illustrated in *figure 9*.

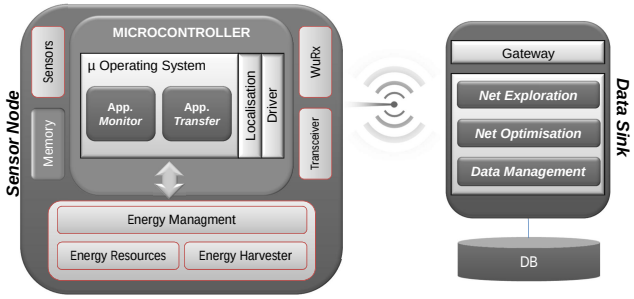



Figure 9. Based on the WuRx capabilities and the two dedicated operational modes, all global coordination task are controlled by the data sink. This includes the net exploration to find all available nodes, the net optimisation to minimise the communication paths and the communication overhead. Each sensor node is only responsible for the management of all energy resources and the sensor data measurements. In order to optimise the measurement quality, the nodes try to calculate its own position in relation to other nodes. Asynchronous communication events are initialised by the data sink.

### B. Data per Time Ratio

The second critical system parameter for efficient WuRx scenarios represents the ratio of data volume and time. All the mentioned WuRx technologies are strongly influenced by the given network load and the respective information distribution over the time. A huge amount of data or continuous data transmissions destroy the conceptual benefits of WuRx technologies. Wake-up concepts aim the energy-efficient handling of asynchronous trigger-events and minimal data. In consequence, our goal is to reduce the network load and to increase the periods of an autonomous operation without active network communication. In order to optimise this ratio, the developers are able to adjust several parameters, which have a direct influence on the data per time radio.

#### 1) Trade-off Measurement Quality/Quantity:

Dependent on the application scenario, the developer has to decide about the required measurement quality regarding to the number of data sets and the size of each data set during a time interval. In this context, little variations could have a huge impact on the system availability. Each measuring sample of the given sensors consumes energy. Local data aggregation and data fusion techniques help us to shrink the data volume, but also take computing time and energy.

#### 2) Data Buffer and Energy Resources:

In many cases, the given WuRx application scenario has to monitor its environment. Accordingly, the measured data sets have to be stored within a local data buffer during the autonomous operational mode. The size of this buffer is limited. In consequence, the size has a direct and strong influence on the data per time ratio.

A small buffer size limits the maximum period for an autonomous measuring or otherwise limits the measuring quality. On the other side, large data buffers increase the overall transmission time and generate bottlenecks during a multi-hop transmission. In case of an emergency, it is more difficult to backup the complete buffer without data loss.

In this context, the available energy resources still represent a key factor for the system behaviour. Additional energy harvesting components in the system architecture, like thermal, vibration or solar energy converter units, help us to extend the energy resources [12][13].

#### 3) Real-time Aspects, QoS:

Each application scenario also implies further requirements regarding to the maximum time delay between the detection and the processing of an event. This also includes the decision about a centralised or a distributed data handling. In this context, we also have to consider the possible prioritisation of specific events, like abnormal measurements or emergency calls. Such *quality of service (QoS)* aspects or in some cases real-time aspects have a huge impact on the system behaviour and the system availability.

### C. Network topology and Routing

With focus on the second operational mode with an interconnected topology, the size and the distribution of the network topology represents another key parameter. In order to wake-up a huge number of nodes, it takes much more time to explore the topology than operating in a small size measuring system with 20 or less nodes. The complexity for calculating efficient route paths increases significantly. During this time, all transceivers have to be active. Advanced topology optimisation approaches [15] and lightweight routing algorithms [16] are essential to handle these challenges.

In this context, the spatial distribution of the nodes represents another important parameter. In case of a high node density, the communication between two nodes could influence the WuRx units of other nodes. Robust transceiver and WuRx hardware components as well as capable communication standards and protocols minimise these kinds of disturbances.

## V. Simulation and Analysis

Within the *nanett* project [2] our research focuses on the evaluation of critical system parameters for WuRx applications.

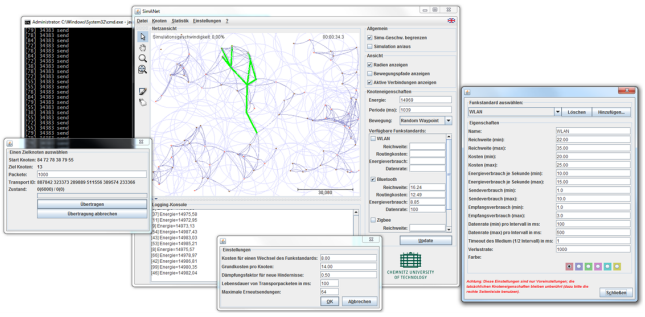


Figure 10. The SimANet framework. The GUI includes several configuration and administration tools. SimANet focuses on high-level simulation scenarios in mid- and large-scaled network topologies.

In a first step, we simulate the influences of the main critical parameters. For this purpose, we use the *SimANet* framework [17][18], which provides a lot of features for a functional analysis in the context of mobile ad hoc and sensor networks. SimANet allows the abstract modelling of wireless communication standards and interfaces. Also several movement models and an energy model are available. Based on the SimANet core, we create a sensor network topology with and without WuRx capabilities. *Figure 10* show the GUI-based version of SimANet. Furthermore, SimANet is also runnable in a massive parallel computing environments like the *CHiC* Cluster (*Chemnitz High Performance Computing Linux Cluster*). CHiC provides more than 512 AMD Opteron 64bit computing cores with 2.6GHz and each 4GB of memory. The network infrastructure uses a high-speed fiber optics *Infiniband* topology.

Accordingly, SimANet provides an excellent scalability to run different kinds of simulation scenario. Regarding to the proposed results, SimANet operates on a normal PC workstation with an 3Ghz Pentium IV and 2GB of memory.

For all the simulation, the used energy model operates with logical energy units. Each process for the handling and transmission of data decreases a nodes energy resources dependent on parameters like the used radio standard or the required electromagnetic field strength for reaching the selected node. The used values for the simulation scenarios are based on a real-world 802.15.4 low power transceiver and an conventional, off-the-shelf WuRx unit. The specified transmission characteristics are mapped to the abstract communication interface model of SimANet [16].

Within the following simulation scenario, we focus on the operational modes and on the data per time ratio. Furthermore, we want to find an optimal trade-off for these parameters.

First of all, we analyse a single hop communication in order to take a closer look on the transmission timings and the standby power consumption of the nodes. The diagrams visualise the differences between a conventional, non-buffered communication and WuRx-enabled systems. All relevant information for interpreting the results are shown in *Figure 11*.

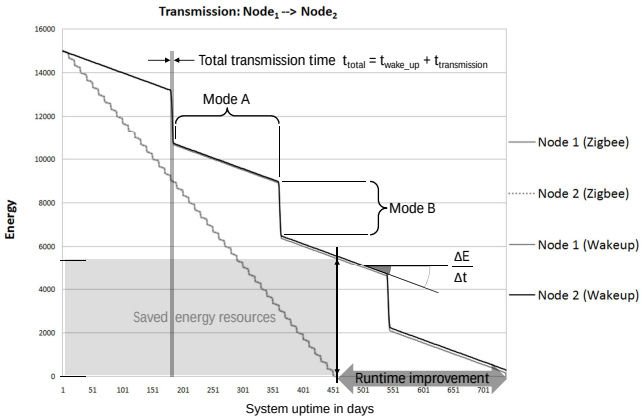



Figure 11. Single hop communication example. Comparison between a conventional ZigBee communication and buffered communication scenarios both with and without WuRx capabilities. The different operational modes for the local data measuring and the data transmission are marked. Our goal is to maximise the system availability and to minimise the power consumption during the autonomous measuring *mode A*, represented by the term $\frac{\Delta E}{\Delta t}$.

*Figure 12* represents the compressed results of four different communication scenarios.

The conventional, continuous transmission provides a maximum battery lifetime of 260 days. The second scenario with a local buffering and the compressed transmission prolongs the lifetime to 300 days. An additional WuRx provides features for switching off the main transceiver. The optimisation level depends on the WuRx technology and reaches up to 300 additional days of operation. As we can see, for a single hop communication, a long autonomous operational period provides the best results.

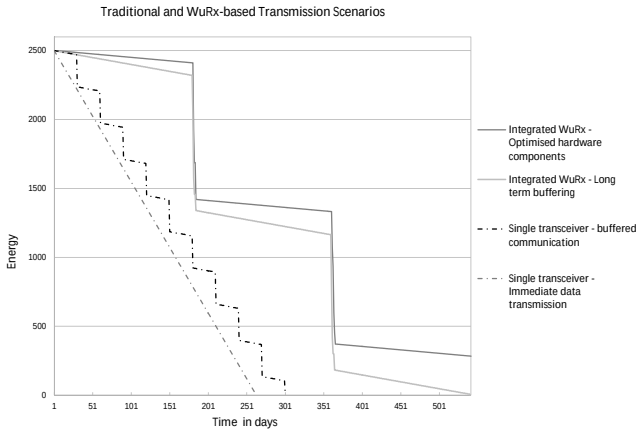



Figure 12. Comparison of different communication concepts. The initial charge state of each node is assumed with 2500 logical units. The results include a conventional communication with a single ZigBee transceiver, a buffered ZigBee communication as well as two WuRx-based scenarios.

Furthermore, we have to analyse the situation in case of a larger topology. Here, other effects influence the results. Routing procedures in a multi-hop environment, limited data throughput in combination with local data bursts and interferences have to be considered.

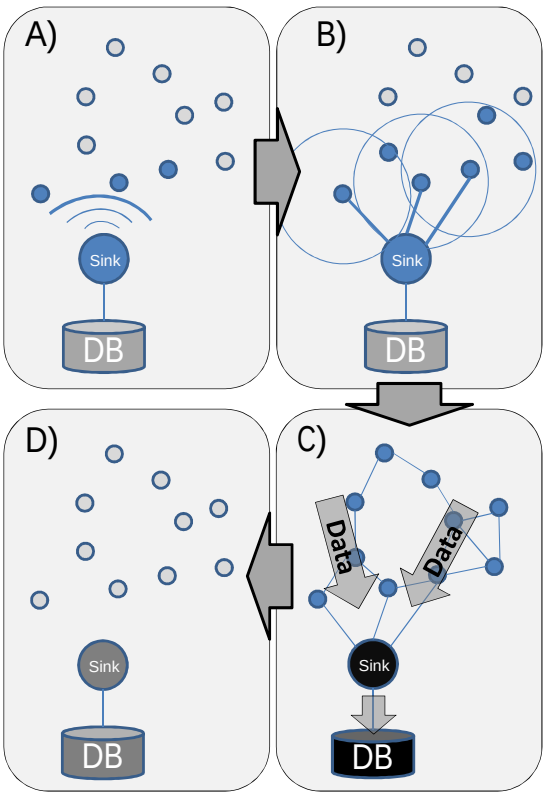



Figure 13. Application scenario - operational schedule with four steps. A) initial wake-up signal from the data sink in order to collect data from each sensor node. B) recursive wake-up signal from active nodes to explore the topology. C) Multi-hop data transmission of the measurement data over the established network topology. D) Communication finished, return to the autonomous mode without network infrastructure and waiting for new WuRx events.

Therefore, the next scenarios focus on several long term measurements within a multi-hop communication topology. The simulations include a uniform and a random distributed network topology with 20 nodes (see *Figure 14* and *15*) and predefined energy resources for each node of 2500 units. We used a simple hop-optimised, reactive routing protocol and no further topology optimisation approaches.

The test scenario is illustrated in 13 and represents a typical data gathering application in the wireless sensor network domain. A predefined data sink collects the measured sensor data from each nodes in a cyclic period. In contrast to conventional communication concepts, our simulation implements the two proposed operational modes for WuRx-enabled devices. Thus, all the node are sensing the environment in a autonomous mode without network connection. The wake-up signal for the dedicated communication mode is initialised by the data sink. After receiving this signal, the nodes activate their main radio transceiver. Accordingly, they are able to establish a network infrastructure to transmit the measurement data to the sink. Finally, if the communication process has completed, the nodes switch off the main transceiver and return to the autonomous mode. Hence, the power management only has to supply the sensor components and the WuRx unit. The overall system power consumption is reduced to a minimum.

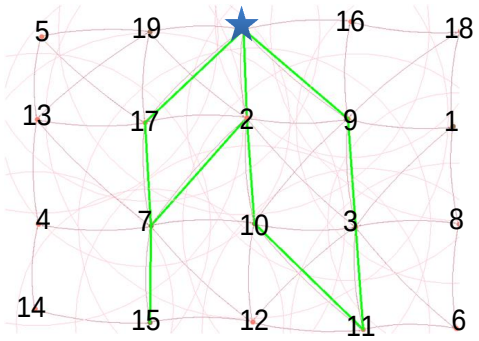



Figure 14. Uniform node distribution. The data sink is marked as star.

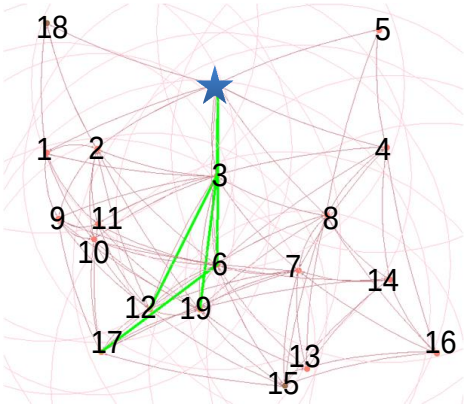



Figure 15. Random node distribution. The data sink is marked as star.

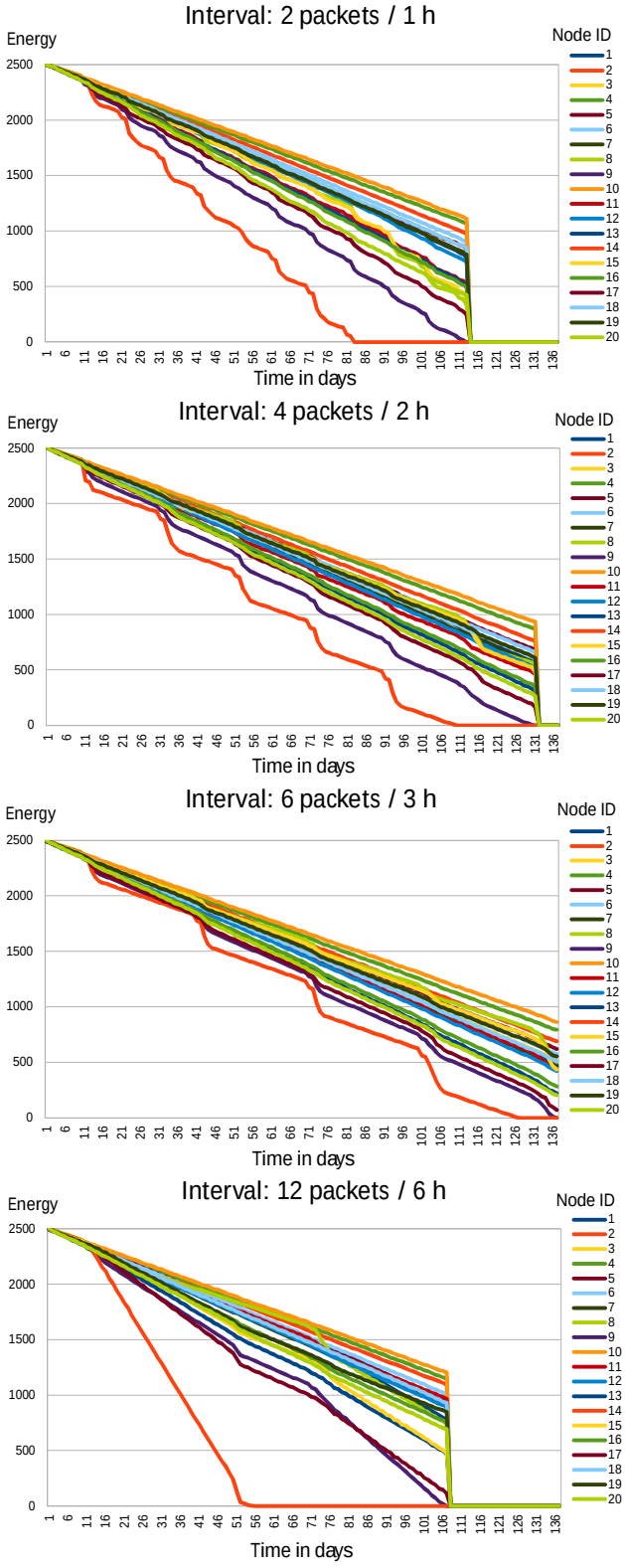



Figure 16. Simulation results for the uniform node distribution.

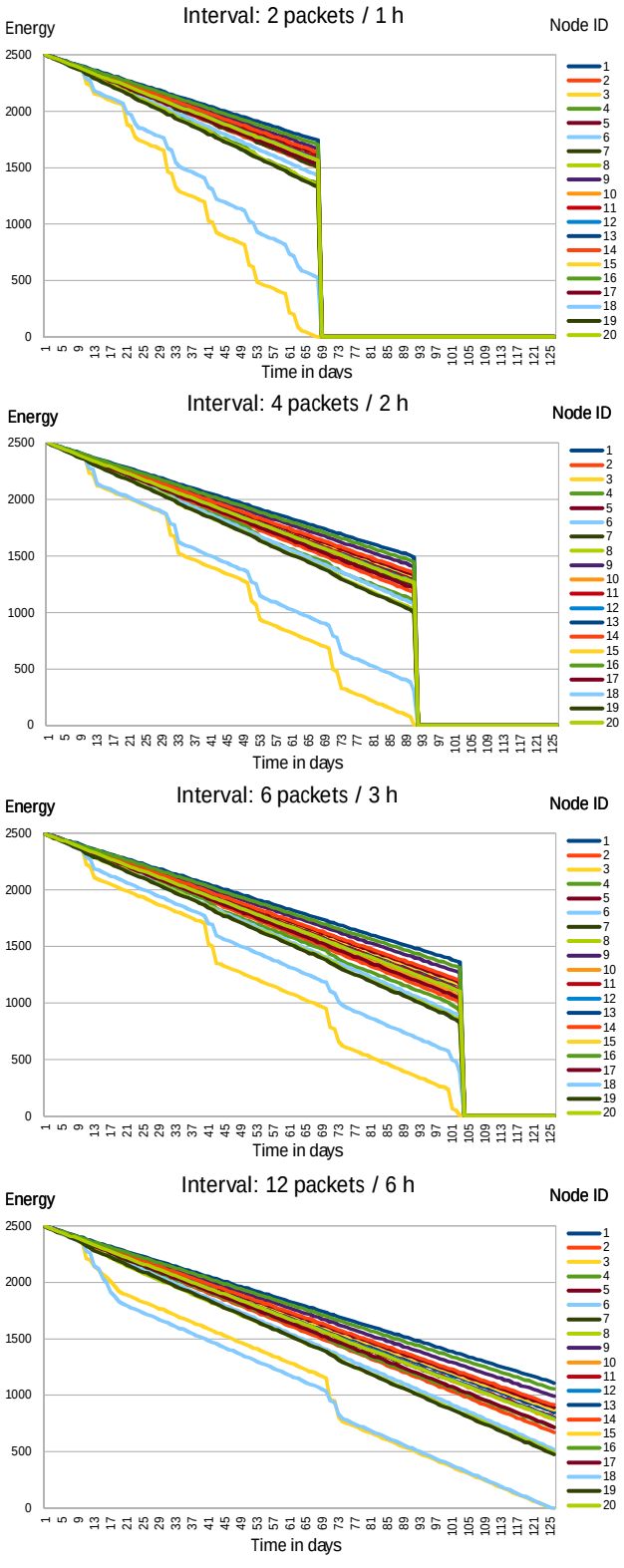



Figure 17. Simulation results for the random node distribution.

During the tests cycles, we varied the local buffer size of the nodes and the transmission interval to the data sink. We started with a scenario of 2 Packets every hour and increasing this interval up to 6 hours with 12 packets. Accordingly, the data per time ratio for a long time window remains constant. Based on different ratio between the two operational modes, we analyse the system uptime. Further indicators for the systems efficiency represent the entire system availability and the spatial distribution of the available energy resources on the nodes (*Figure 16* and *Figure 17*).

Both simulation scenarios provide interesting results. First of all, the diagrams clarify the trade-off between local buffering approaches and data transmission cycles. For the uniform node topology, the transmission scheme of 6 buffered packets every 3 hours provides the best results. In contrast to that, the random node distribution provides the best result for a buffering scheme of 12 packets in 6 hours. Accordingly, the overall system availability varies between 70 and 136 days. In other words, an optimal configuration is able to double the system availability. Anyway, in cases of a continuous data transmission or real-time requirements for the event handling, the system is not able to benefit from the mentioned WuRx technologies.

Another interesting simulation output represents the energy distribution in the networks. Especially in the uniform topology, the data forwarding process generates a lot of network traffic within a few nodes around the data sink. Based on these data bursts, the energy resources of the corresponding nodes decrease rapidly and the topology splits into disjunctive sets. Within the random topology, this problem strongly depends on the node distribution. In any case, advanced routing approaches for WuRx-enabled application scenarios are essential and part of our actual research work.

## VI. Conclusion

In this paper we discuss the capabilities of WuRx technologies and the requirements for an efficient usage. We define sufficient application scenarios and critical system parameters. Besides environmental parameters and technological restrictions of the hardware system, the application itself has a significant influence on the efficiency of WuRx technologies. We clarify, that the existence of an asynchronous, event-triggered application scenario represents the key requirement for an efficient usage of a WuRx unit. In this context, we introduce a separation into two different operational modes in order to use the benefits of a WuRx unit. The first one represents the local, autonomous operation and the second one is the interconnected operation within the network topology. Based on these modes, we simulate the differences between several communication scenarios with and without WuRx capabilities. Our simulation results confirm the impact of the critical parameters on the system availability.

Further research work focuses on adapted routing and topology optimisation approaches in the context of WuRx applications. Enhanced simulation scenarios with several hundred nodes will help us to evaluate these approaches. Further WuRx improvements of the power consumption and the wake-up latency will be done within the *nanett* project.

## Acknowledgement


This research work is supported by the *nanett* project (*nano system integration network of excellence*) in cooperation with the *Bundesministeriums fuer Bildung und Forschung FKZ 03IS2011A*.

## References

[1] C. Nagy and F. Eady. *Embedded Systems Design Using the TI MSP430 Series*. Academic Pr Inc, Oktober 2003.

[2] nanett Germany. nano system integration network of excellence. `http://www.nanett.org`, Mai 2011. [Research Project - Online reference].

[3] B.P. Otis. *Ultra-Low Power Wireless Technologies for Sensor Networks*. PhD thesis, University of California, Berkeley, 2005.

[4] Y.H. Chee. *Ultra-Low Power Transmitters for Sensor Networks, SCHOOL = University of California, Berkeley, YEAR = 2006*. PhD thesis.

[5] N.M. Pletcher. *Ultra-Low Power Wake-Up Receivers for Wireless Sensor Networks*. PhD thesis, University of California, Berkeley, 2008.

[6] M. Vodel, M. Lippmann, M. Caspar, and W. Hardt. A Capable, High-Level Scheduling Concept for Application-Specific Wireless Sensor Networks. In *Proceedings of the World Engineering, Science and Technology Congress (ESTCON2010) - 4th International Symposium on Information Technology (ITSIM)*, pages 914–919, Kuala Lumpur, Malaysia, June 2010. IEEE Computer Society.

[7] E. Shih, P. Bahl, and M. Sinclair. Wake on Wireless: An Event Driven Energy Saving Strategy for Battery Operated Devices. In *Proceedings of the 8th Annual International Conference on Mobile Computing and Networking*, pages 160–171. ACM, 2002.

[8] L. Gu and J.A. Stankovic. Radio-triggered Wake-up Capability for Sensor Networks. In *Proceedings of the 10th Real-Time and Embedded Technology and Applications Symposium*, pages 27–36. IEEE, May 2004.

[9] P. Le-Huy and S. Roy. Low-Power 2.4GHz Wake-Up Radio for Wireless Sensor Networks. In *Proceedings of the International Conference on Wireless & Mobile Computing, Networking & Communication*, pages 13–18. IEEE, 2008.

[10] M.S. Durante and S. Mahlknecht. An Ultra Low Power Wakeup Receiver for Wireless Sensor Nodes. In *Proceedings of the Third International Conference on Sensor Technologies and Applications*, pages 167–170. IEEE, 2009.

[11] N.M. Pletcher, S. Gambini, and J.M. Rabaey. A 2GHz 52$\mu$W Wake-Up Receiver with -72dBm Sensitivity Using Uncertain-IF Architecture. In *Proceedings of the International Solid-State Circuits Conference*, pages 524–525, San Diego, USA, 2008. IEEE.

[12] S. Beeby, M. Tudor, and N. White. Energy Harvesting Vibration Sources for Microsystems Applications. *Measurement Science and Technology*, 17(12):175–195, 2006.
[13] S. Beeby, M. Tudor, and N. White. Energy Harvesting Vibration Sources for Microsystems Applications. In *Proceedings of the 2nd ACM International Workshop on Wireless Sensor Networks and Applications (WSNA)*, pages 11–19, San Diego, USA, September 2003. ACM.
[14] M. Vodel, M. Caspar, and W. Hardt. Critical Parameters for an Efficient Usage of Wake-Up-Receiver Technologies. In *Proceedings of the International Conference on Computer Applications and Network Security (ICCANS2011)*, pages 100–105, Male, Maldives, 2011.
[15] M. Vodel. *Topology Optimisation in Mobile Ad Hoc and Sensor Networks: Evalutation of Distributed Algorithms for Self-Organising Systems*. Vdm Verlag, September 2008.
[16] M. Vodel. *Radio Standard Spanning Communication in Mobile Ad Hoc Networks*. PhD thesis, Chemnitz University of Technology, Germany, 2010.
[17] M. Vodel, M. Sauppe, M. Caspar, and W. Hardt. A Large Scalable, Distributed Simulation Framework for Ambient Networks. *Recent Advances in Information Technology and Security - Journal of Communications*, 2009.
[18] M. Vodel, M. Sauppe, M. Caspar, and W. Hardt. The SimANet Framework. In *Proceedings of the International Conference on M4D: Mobile Communication Technology For Development (M4D)*, pages 88–97, Karlstad, Schweden, 2008.

**Dr. Matthias Vodel** was born in Germany in 1982. He received the German Diploma degree (equal to M.Sc.) in Computer Science from the Chemnitz University of Technology with the focus on computer networks and distributed systems in 2006. In 2010, he received his Ph.D. degree in Computer Science from the Chemnitz University of Technology / Germany. In his dissertation, he proposes a novel concept in the field of radio standard spanning communication. For his thesis, he received the "Commerzbank Award" This includes a cooperative cross layer routing approach and advanced topology optimisation strategies for mobile Ad Hoc and sensor networks. Currently, he works as a postdoctoral research fellow at the Department of Computer Science, Chair of Computer Engineering at Chemnitz University of Technology, Germany. Actual research projects focus on optimisation strategies in distributed sensor and actuator systems. Additional fields of interest are network security topics, protocol engineering and embedded systems.
During a research exchange at the King Mongkut's University of Technology Northern Bangkok / Thailand in May 2008, Dr. Vodel received the best paper award for the conference paper "EBCR - A Routing Approach for Radio Standard Spanning Mobile Ad Hoc Networks".

**Mirko Caspar** was born in the former German Democratic Republic in 1980 and received the German Diploma degree (equal to M.Sc.) in Computer Science, focuses electrical engineering and embedded systems, from the Chemnitz University of Technology, Germany, in 2006.
He is currently research assistant at the Department of Computer Science, Chair of Computer Engineering at Chemnitz University of Technology, Germany where he also is pursuing his Ph.D. degree under guidance of Prof. Dr. Hardt. His special field of interests are Test-Automation for Embedded Systems, Organic/Pervasive Computing and Radio-Standard-Spanning Communications. For his Diploma/Master thesis, Mr. Caspar received a special award within the "SAX-IT Nikolaus-Joachim- Lehmann-Preis"

**Prof. Dr. Wolfram Hardt** is professor for computer science and head of the Computer Engineering Group at the Chemnitz University of Technology. He was born Germany 1965 and received the German Diploma degree (equal to M.Sc.) in Computer Science in 1991 from the University of Paderborn. Accordingly, Prof. Hardt received the Ph.D. degree in Computer Science from the University of Paderborn in 1996.
From 2000 to 2002 he was chair of the Computer Science and Process Laboratory at the University of Paderborn / Germany. After that he worked as chair of the operating systems Dept. at the University of Kassel / Germany. Since 2003 Prof. Hardt became chair of the computer engineering Dept. at the Chemnitz University of Technology / Germany. He is editor of a scientific book series about self-organising embedded systems and has published more than 70 papers.
Prof. Hardt is member of the Association for Electrical, Electronic and Information Technologies (VDI/VDE), the Association for Computer Science (GI) and the Association for Computing Machinery. Since 2006 he is committee member of the DATE conference - "Design Automation and Test in Europe"; Topic: System Synthesis and Optimization. His research interests include Hardware/Software Co-Design, Organic Computing and Reconfigurable Hardware.

# An Improved Adaptive Routing Algorithm Based on Link Analysis

Jian Wang
Sichuan University / College of Computer Science, Chengdu, china
Email: wj_98@163.com

Xingshu Chen and Dengqi Yang
Sichuan University / College of Computer Science, Chengdu, china
Email: { chenxsh@scu.edu.cn, dengqiyang@163.com}

***Abstract*—DHT (Distributed Hash Tables) has been applied to the structured P2P system to achieve information retrieval and positioning efficiently. KAD is a large-scale network protocol based on the XOR metric in practice, which uses DHT technology to improve network salability without central server. However, the increasing malicious pollutes routing tables to reduce seriously the query performance. Thus, an improved adaptive algorithm based on social network is proposed in order to improve routing table updating algorithm. Firstly, the data structure of routing table is adjusted to store value of centrality and prestige. Secondly, the request nodes can adaptive select nodes to send messages. Then when a find process is terminated, the node will calculate the two values for all participating nodes using the corresponding centrality and prestige algorithms based on XOR metric. Finally, the node updates the routing table depend on the above result. The above algorithm was implemented in an open source project named LibTorrent to test effectiveness. This experiment last a month to verify the change of the search success ratio in a KAD network with about 30% malicious nodes. The results show that the optimized adaptive routing algorithm can effectively resist the attack for routing table and improve the search success ratio of the node. Moreover, this lightweight algorithm is conducive to the deployment in practice without extra network burden.**

***Index Terms*—DHT, KAD, routing algorithm, Link Analysis**

## I. INTRODUCTION

P2P system can be subdivided into centralized P2P system, unstructured P2P and structured P2P system according to the overlay networks of organizing [1]. The first generation centralized P2P system disappeared from view because of inherent weaknesses. As a result of DHT technology, structured P2P system such as CAN、Kademlia、Tapestry and Chord, makes information retrieval and location have been greatly improved efficiency compared with unstructured P2P system. Hence, structured P2P is a main direction development of P2P system. KAD is a kind of DHT technology based on XOR metric, which has been widely adopted in practice file shared software，such as eMule/aMule[2]、BitTorrent DHT、Azureus[3]、MainLine and so on.

KADEMLIA is widely used routing protocol based on XOR metric, which has some obvious advantages compared with others [4].The distance between two nodes in the network is obtained using the XOR calculation. Before a node joins the DHT network, it requires a random 160 bit (or 128 bit) number as their identity using hash algorithm (MD5 or SHA1). When a node wants to share a file, it also uses the same hash algorithm for the file with any length to generate a 160 bit digest a value as the key word. A pair including key word with the publisher information (IP and Port) will be released to store in k nodes whose id values closest to the key word. To speed up the retrieval efficiency, each node in the network will create and maintain a routing table that stores a part of the information for all nodes. The routing table for KAD is organized by some k buckets [5]. When a node starts a request, it will pick up α nodes whose ID is closest to the object file hash value from the routing table to send the request parallel. Responders pick up β nodes whose values are closest to key value in their routing tables and back to the requester with messages including β nodes. Then, the requester selected some closest targets from α*β nodes depending on metric distance. The search process will be repeated iteratively until the target is found or failure. This failure means not more closer nodes can be found. This prefix matching mechanism ensure each search can at least halve the XOR distance，and the maximum number of search times not exceed $\log_2 N$, N denotes the number of nodes in the network. This algorithm made a better trade-off in performance and overhead.

## II. RELATED WORK

Some previous papers focus on how to improve the lookup performance [6, 7]. Sergio Marti et al propose SPROUT algorithm to route using social links, which can increase the probability of successful routing [8].Liu et al present a approach to build a interest-based communities, which can short the deliver path and improve the routing efficiency [9].George Danezis and Prateek Mittal present the Sybilinfer algorithm for labeling the nodes in social

Manuscript received March 5, 2011; revised June 3, 2011; accepted August 12, 2011
Corresponding author, Email addresses: wj_98@163.com (Jian Wang)


doi:10.4304/jnw.7.1.135-140

network to identify the Sybil[10].Yu et al analysis reliability among nodes depend on social link to determine which are Sybil in the network. They suggest using the SybilGuard algorithm to limit the influence of the Sybil attacks. Liu et al begin to study the nodes in the routing table based on KAD network, they present a social link based on IP distance Metric to improve routing performance in KAD network [11]. But, this algorithm is not suitable for the KAD network based on XOR metric. Moreover, a node with internal IP address can't be properly ranked.

## III. ROUTING ALGORITHM DESCRIPTION

The main difference between DHT and traditional network is that each node in DHT was been seen as a client and server. Each node not only can download/upload files, but also provide route services. Therefore, the performance of routing table will play a vital role for node search. The routing table in each node has provided the redundancy to response to frequent churn that nodes free join or leave the network. In order to maintain the availability, each node needs to interact with other nodes and update the corresponding entry in the routing table. Each routing table in a node saves part of the nodes' information whose distance from himself between $2^i$ and $2^{i+1}$, say d= [$2^i$, $2^{i+1}$). The i denotes the changes range from 0 to 160，say i= [0,160]. An entry in routing table called a K-bucket can store no more than k nodes information. According to the found time, those nodes are arranged. The node with first seen was placed on the head, and latest seen was placed on the tail in each bucket.

### A. State model for a node

Each node will create and update routing table, along with node joins network, sends request, provides service and leaves network. It is necessary to model a node in order to understand behavior. It shows the state transition for a node in figure 1.

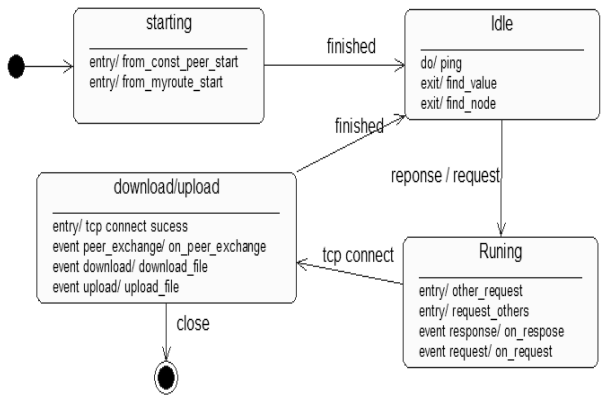



Figure 1. the node process transition

It can be seen from above figure, each node in starting phase, idle phase, running phase and download/upload phase requires obtain information by interacting with other nodes to complete routing table crate and update. The main operations for interaction among nodes are following:

- Starting state: It means a node issues a request to require a node in the DHT network in order to join the network. The responder can provide 20 nodes information in back message, and the requester inserts 20 nodes into the routing table. Then the node continues to fill the routing table after it sends some messages to find its own to the 20 nodes and receive the back messages. Meanwhile, the 20 nodes update the routing table with requester information. How to ensure the reliability of the fixed nodes is the premise of the node can successfully join the network. Practically, some fixed nodes instead of selected nodes randomly always are used as guide nodes to prevent nodes from isolation attack. Although this method is simple and effective, it increases the dependence of these fixed nodes and borrows the bottleneck and single point of failure problem. Therefore, the really matter is how to reduce the dependence on the fixed nodes to improve the join efficiency of nodes initialization process.
- Idle state: The term indicates that the node is in the absence of requesting or providing services to other nodes. The node in this state is only to randomly select some nodes from the routing table to ping. If no response messages, the target nodes will be removed by requester from the routing table, which ensures that most nodes are online. Unfortunately, taking to account the efficiency, the node is only to select part of nodes to detect. In other words, not all nodes in the routing table at a time online.
- Running state: Nodes request or provide service for other nodes in network. The nodes issue request or response for the request message that want to find a value or node.
- Download/upload state: Access to the information for the target nodes who own the source files, request nodes will start to download and upload process. When nodes have finished the download, they will announce to K nodes whose id is closest to the file's hash value with store operation. The pair with node IP and Port will be stored in the k different locations so that others can find the value with high probability.

### B. Four Operations for a node

Node in the idle state, running state will call or receive RPC (Remote Process Call) operation to achieve the maintenance of its own routing table. Four operations including in RPC are Ping, Find Node, Find Value and Store. Their functions as follow [13]:

- Ping: The function is to detect whether a specified node is still online.
- Find Node: The function is to ask a node return k nodes closest to a required ID from the routing table.

- Find Value: The operation is similar to Find Node. The function is to ask nodes whether own the key value as same as require value.
- Store: The function is to inform K nodes to store <key, Value> information as to others search.

The node who has received the one of above process calls may update the routing table. But the updating operation will be successfully depends on the following figure.

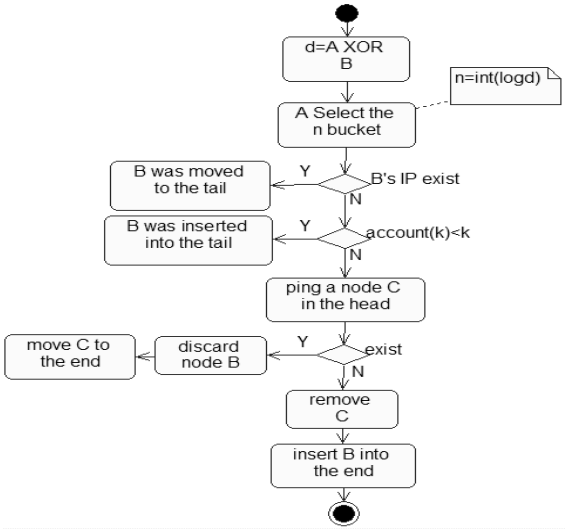



Figure 2. the process after receiving the RPC operation

The process can ensure the node fill corresponding bucket in routing table as soon as possible. When the number of a bucket reach limit and new request come, the node will ping another node that is first to be inserted into the bucket early.

## IV. THE IMPROVEMENT FOR ROUTING TABLE MAINTENANCE MECHANISM

### *A. Analysis of the current problems*

From the above analysis, the routing table maintenance mechanism is relatively efficient. It fully takes into account the dynamic and overhead for interactive with nodes. However, there are two problems for the mechanism as following:

- Has received the RPC command, the node only to test the node that in a bucket head. Its means whether the node will update the routing table depends on the head node rather than others in a bucket. Besides, the node can randomly select nodes in some buckets that haven't updated any node to ping on a regular basis. Obviously, it is not feasible that ping all nodes when the node need to update. Considered an extreme case, a new node can't be inserted into a bucket even if the other k-1 nodes are not online. This is not unsafe for a node if some malicious nodes launch the routing attacks. Because, these malicious nodes with multiple IP can stick to ping a victim node so that make themselves added into the routing table. At the same time, when a malicious node has been moved to the head in a bucket, the attacker can prevent new nodes from inserting into the bucket.
- A requester sends the request for finding value or find node, and responder randomly select α nodes from routing table. A requester will update the corresponding node information without checking. This means that a responder can more easily make requester use some victim nodes update the routing table. The search process will be repeated until one (or more) node is found or no closer nodes are returned. It is easy to known that a node find a target node after 1/2log2N (N is the number of all nodes) on average. Thus, a node may has received 1/2log2N*α malicious nodes information for a search process.

The maintenance of routing table should be improved to keep more reliable nodes in the routing table. Thus, it is the point that how to assess the reliability of node. It is more likely for a reliably node to provide search service in routing process. As a result, the improvement strategy of routing table maintenance consists of two parts in this paper: node evaluation algorithm and the node adaptive selection algorithm.

### *B. routing improvement algorithm based on link analysis*

DHT network and social relationship network is very similar. It is helpful to assess the node in the DHT with link analysis in the social network. If nodes are viewed as nodes, the activity or relationship between nodes can be represent as links, which the DHT network model will be become a direction grapy. The characteristics of nodes, such as behavior, authority, position and so on, will be easy to study based on the grapy. Actually, both centrality and prestige are measures of prominence of a node in the network [12]. The higher of degree of a node, it's more reliable for the other nodes in the network. To explain the term of node reliability, the definitions of centrality and prestige are introduced in this paper.

Node prestige: A prestige node is object of extensive links as a responder. Generally, a requester sends message to the node with higher prestige value, which gets the correctly back nodes with higher probability.

Node centrality: A centrality node is one who has more links, which is more important compare to other with relatively fewer links. Generally, a requester sends message to the node with higher centrality value, which gets the back nodes with shorter the distance for destination node.

From above definition, the focus of routing table updating is converted to calculate the value of centrality and prestige. To achieve this goal, the following measures will be implemented:

- Changes the routing table structure

To compute and store the value of centrality and prestige, the value of centrality and prestige of a node will be added respectively in each k bucket. A five-tuple of <ID, IP, Port, Auth, Cent> can be used to represent the node.

- Changes the RPC operations

There is no difference for a request node sending Ping or store command. When the node sends the find value or finds node command, it is essential for the requester to evaluate the centrality and prestige of those nodes that are extracted from a return message. A requester may update the routing table depend on the result of evaluation. The details of improved process are as follows: Node A selected some nodes to send the request. Each responder provides β nodes for requester. As a result, a responder has an in-link and β out-links. If the number of return nodes can not meet the requirements, the requester will start retransmission mechanism.

- How to select the nodes for a requester and responder depending on the adaptive algorithm.
- The requester does not update some nodes immediately and pick up some closer nodes by comparing the return nodes to repeat the previous step.
- When the target node or value is satisfied, the search process is termination.
- The value of centrality and prestige for each return node is calculated depend on the number of in-link and out-link.
- According to the value of centrality and prestige, the requester starts to update corresponding buckets.

The node adaptive selection must be referred to above process in order to explain more clearly. That is how to select some nodes in the beginning and how to update some nodes at the end.

Firstly, a requester picks up some evaluated nodes that are composed of nodes with higher prestige and nodes with higher centrality value for first requesting. Then, an equation to describe the relationship, say Pr =dp/dc, α=dp+dc.

The Pr is the proportion of prestige and centrality value. The higher the value of Pr means the node set owns more security. On the contrary, the set node owns more efficient.

For the first requesting, the requester selects α-1 central nodes and a prestige node. If it failed, the node will automatically pick up α-2 central nodes and 2 prestige nodes, and so on. To obtain a higher value of prestige and centrality, a responder was encouraged providing some better nodes. Actually, it is no obviously different for requester and responder to select the prestige and centrality nodes from the routing table. The adjustment process is automatic, so it is called adaptive. But, it is allowed for user to adjust parameters in order to get a reasonable node set with different security and efficiency in practices.

Secondly, after the requesting is finished, the requester begin to update some nodes depend on the two values. The Pu is the ratio of the amount of being updated prestige nodes (dp') to centrality nodes (dc') in a bucket, say Pu = dp'/dc'.

### *C. Algorithm Implementation*

The improved algorithm for ranking all nodes in a find process is based on the prestige and centrality algorithm [12]. The requester needs to calculate the values at the end of find instead of each step. After entered the KAD network, the node will initialize threshold values of prestige and centrality for each node to 1 separately. In this paper, the algorithm of degree centrality is adjusted to calculate the centrality value. The value of a node is defined as $Cc(i)$

$$Cc(i) = \frac{dx(i,j) * \Re}{(n-1)} = \frac{dx(i,j)}{(n-1) * dx(s,t)} \quad (1)$$

$$dx(i,j) = dx(i,t) - dx(t,j) \quad \text{if } dx(i,j) > dx(i,t)\ . \quad (2)$$

Where n is the number of nodes in a find process. Let t is the target node and s is the requester, the $dx(i,j)$ denotes XOR distance from i to t. The direct out-link need to be calculated for i, which is the centrality of those nodes who have been provided by i can be got. The value of the formula ranges from 0 to 1 because of the normalization with $(n-1) * dx(s,t)$.The $Cc(i)$ depends on two factors of the XOR distance and the number of nodes. According to XOR metric characteristic, $dx(i,j) > dx(i,t)$ indicates that j has exceeded the target node. So, $dx(i,j)$ must be adjusted with $dx(i,t) - dx(t,j)$ to reflect the extent of j close to t. From the DHT characterizes, it is easy to understand that a node more likely to know the nearer node. Hence, a node can be ranked with higher value because it provides a successor closer to target.

Similarly, the proximity prestige value is calculated as $Pp(i)$

$$Pp(i) = \frac{|Ii|/(n-1)}{\sum_{j \in Ii} dx(j,i) / |Ii|} \quad (3)$$

$$dx(j,i) = dx(j,t) - dx(t,i) \quad \text{if } dx(j,i) > dx(j,t)\ . \quad (4)$$

Where $|Ii|$ is the size of set that all nodes who can link directly or indirectly to I. The value of the above measure ranges form 0 to 1. The $Pp(i)$ depends on two factors of the XOR distance and the number of nodes related to i. Note the distance is inverse proportional to the prestige value. Reviewing the characters of k buckets, a node can know more with their neighboring nodes. For example, the two values can be calculated in the figure 3.

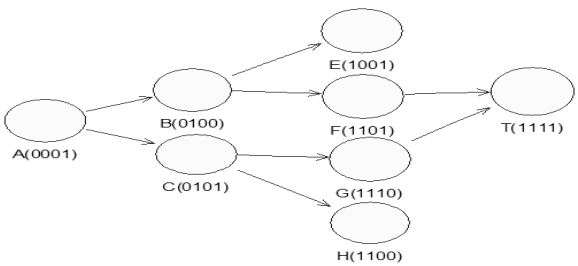



Figure 3. the simplified routing process from A to T

The 4 bits number in brackets instead of 16 is the node ID in order to simplify the routing process. After node A finished the find operation, those related nodes including A, B, C, F, G and T to node A can be selected out to

calculate, say n=6 and $dx(i, j)$ =14. Two values for all nodes are given in the following table.

TABLE I.
TWO VALUES FOR ALL RELATED NODES

| | A | B | C | F | G | T |
|---|---|---|---|---|---|---|
| Cc(i) | 0 | 0.129 | 0.157 | 0.029 | 0.014 | 0 |
| Pp(i) | 0 | 0.04 | 0.05 | 0.038 | 0.033 | 0.132 |

Note that the value of $dx(A,G)$ has been adjusted to 13 because of $dx(A,G) > dx(A,T)$.As can be seen from the table, the value of $Cc(B)$ is smaller than $Cc(c)$, which indicates that node linked by B is closer to T. the value of $Pp(B)$ is bigger than $Pp(C)$, which indicates that C is more reliable than B. Through the above algorithms, the requester can get the value of prestige and centrality for the all participating nodes. As a result, the requester will decide which of nodes will be replaced.

## V. EXPERIMENT AND RESULT

To test the validity of this way, an open source project named LibTorrent was selected for this work. LibTorrent is a feature complete BitTorrent implementation focusing on efficiency and scalability [13, 14]. There are two experiments are used to test the search success ratio and result for resisting the routing attack. Two different versions of the client program are created, in which a client program is created using original routing algorithm and another client program is created using above improvement routing algorithm. A total of 1,000 normal nodes involved in this testing in the DHT network. Half of the total number of the nodes uses the former client program, while the rest of the nodes use the second program where some parameters are set, for example α= β =3 in method improved-1, dp=1 and dc=2 in method of improved-2.There are about 300 nodes who insist in polluting other node's routing tables enter the network. Test process lasted for one month. Relevant data are collected after each nodes issue about 300 find value commands in the DHT network. It shows the difference for the search success rate in figure 4.

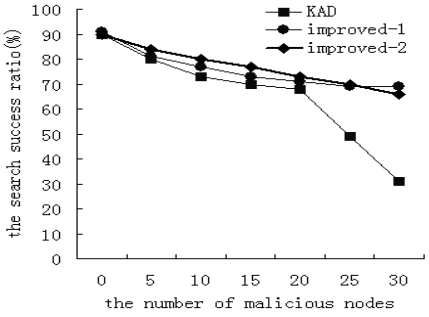



Figure 4. the search success ratio with different algorithm

In the same net circumstances, the result of malicious poisoning routing table is tested in another group of experiments. A total of 1000 normal nodes randomly request with 300 malicious nodes that use sending ping message and forwarding message with some unexpected nodes to poison others. The amount of 50 malicious nodes are entered the DHT network every other day. The half of 1000 nodes update routing table depending on the improved routing algorithm. The amount of malicious in every node's routing table is added up for comparing the effects of two algorithms against attacks. These statistical data are shown in figure 5.

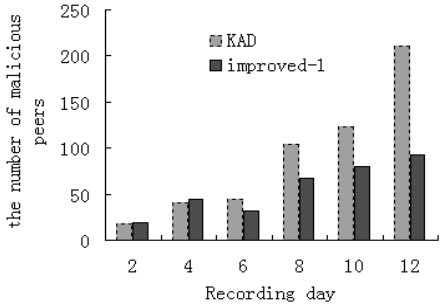



Figure 5. the comparative effects against malicious nodes attack

From the above result can be seen that the improvement algorithm obviously improved the search success ratio and resisted the routing table attacking by preventing those malicious nodes from inserting. If a malicious node can not contribute for other's request, the node will be removed from routing table because of lower the prestige or centrality value. Moreover, the two values not change even if the malicious initiative to send RPC message to a specified requester. In addition, the algorithm is a lightweight algorithm without increasing the client burden significantly.

## VI. CONCLUSION

In this paper, we analyzed some matters for using RPC command to update routing table and presented a lightweight routing algorithm based on link analysis to resist routing table attack, which can make more actively nodes remain in the routing table. Some parameters are automatically adjusted in this algorithm to adapt to different network environment without extra bandwidth. The result of the experiment showed that this algorithm can rank the nodes in the routing table. As a result, the requester can make a trade-off between efficiency and security. With IPV6 implementation, node can obtain a unique ID by Hashing IP address. This rank algorithm will play a more important role for create and update routing table in order to route with more efficiency and reliability.

## REFERENCES

[1] Zhiyi Chen, Minghe Huang, Qi Tan, "The Design of Kademlia System Base on Network Topology Matching",2010 Second International Workshop on Education Technology and Computer Science, pp. 146-147, March 2010.
[2] aMule network. http://www.amule.org.
[3] Azureus. http://azureus.sourceforge.net.
[4] Liang Guangmin. "An Improved Kadmelia Routing Algorithm for P2P Network", 2009 International

Conference on New Trends in Information and Service Science. pp. 63-64, June, 2009.
[5] Moritz Steiner, Taoufik En-Najjary and Ernst W. Biersack, "Exploiting KAD: Possible Uses and Misuses", Computer Communication Review. vol.37, pp. 65-67, October 2007.
[6] Peng Wang, James Tyra, Eric Chan-Tin, Tyson Malchow, Denis Foo Kune, Nicholas Hopper, Yongdae Kim , "Attacking the Kad Network", Proceeding SecureComm '08 Proceedings of the 4th international conference on Security and privacy in communication networks, pp. 22-25, Jul 2008.
[7] Daniel Stutzbach, Reza Rejaie. "Improving Lookup Performance over a Widely-Deployed DHT". The 25th IEEE International Conference on Computer Communications Proceedings, pp. 2-4, April 2006.
[8] Sergio Marti, Prasanna Ganesan and Hector Garcia-Molina, "DHT Routing Using Social Links", The 3rd International Workshop on Node-to-Node Systems (IPTPS 2004), pp. 100-111, February 2004.
[9] Kun Liu, Kanishka Bhaduri, Kamalika Das, Phuong Nguyen and Hillol Kargupta, "Client-side Web Mining for Community Formation in Node-to-Node Environments", The 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,vol.8, pp. 11-20, December 2006.
[10] George Danezis, Prateek Mittal. "SybilInfer: Detecting Sybil Nodes using Social Networks", IEEE/ACM Trans. on Networking, pp. 576-581, January 2009.
[11] Xiangtao Liu, Yang Li et al. Social Network Analysis on KAD and Its Application. Web Technologies and Applications. pp. 327-329, 2011.
[12] Bing Liu. "Web Data Mining", Beijing:Tsinghua University Press, pp. 175-190, Dec 2009.
[13] LibTorrent.http://www.rasterbar.com/products/libtorrent/index.html
[14] Zhang Wen, Zhao Ziming, Yang Tianlu and Wei Xiao kang, "Technology Principle and C++ Development", Beijing: Post &Telecom Press,pp.230-260, Aug 2008.

**Jian Wang** received his M.S. degree from College of Computer Science, Sichuan University, China, in 2009. He is currently working toward the Ph.D. degree at the same school. His research interests mainly focused on computer network, information security, and especially peer-to-peer technology.

**Xingshu Chen** received her Ph.D. degree from Institute of Information Security at the University of Sichuan of China, in 2004. She has been a professor at the Computer Science College of Sichuan University, China, since 2005. Her general research interests lie in peer-to-peer networks, information security and computer networks. She is currently the director of the Network and Trusted Computing Institute, Computer College of Sichuan University.

**Dengqi Yang** received his M.S. degree from College of Computer Science, Yunnan University, China, in 2006.He is currently working toward the Ph.D. degree in School of Computer Science, Sichuan University of China. His research interests mainly focused on computer network, information security include and peer-to-peer technology.

# Secure VPN Based on Combination of L2TP and IPSec

Ya-qin Fan
College of Communication. Engineering, Jilin University
Changchun 130012, china
Suntianhang27@163.com

Chi Li
College of Communication. Engineering, Jilin University
Changchun 130012, china
610516518 @qq.com

Chao Sun
College of Communication. Engineering, Jilin University
Changchun 130012, china
Sunchao.s.c@163.com

***Abstract*- This report is written to provide a method of building secure VPN by combination of L2TP and IPSec in order to meet the requirements of secure transmission of data and improve the VPN security technology. It remedies the secured short comes of L2TP Tunneling Protocol Tunneling Protocol and IPSec security. Simulation and analysis show that the construction method can improve the security of data transmission, and the simulation results of VPN is valuable for security professionals to refer.**

***Index Terms*- L2TP, IPSec, VPN, tunneling**

## I. INTRODUCTION

Since the last century, 80 years, computer and data network as the theme of modern technology has made great breakthrough, and its application has been deep into every corner of society. In particular the emergence of the late 80s Internet, in just a few years the world will be tens of thousands of local area networks and even thousands of computers has become a transnational, trans-regional between the information superhighway. The rise of the global Internet data communications network as a single entity, information sharing, communication convenient and efficient, leading to the traditional information technology resources with a breakthrough capacity[1]. VPN (Virtual Private Network) as an important way for enterprises to use the Internet.

VPN: virtual private net can achieve different network components and connections between resources. Virtual private network to use internet or other public infrastructure of the Internet for users to create tunnels, and provide the same private network security and security features[2]. Throughout the VPN technology, you can see, security and service quality is the VPN technical support, the needs of business users to promote wider use of IPSec VPN's, carriers are great efforts to build MPLS VPN, VPN technology allows the development of China's future with more diverse , flexibility, technical services and demand trends closely. VPN using a variety of security protocols, in the public network to establish a secure tunnel to provide private network capabilities. In ensuring quality of service at the same time, security is the primary VPN features.

In the current study VPN, L2TP and IPSec is the most widely used two kinds of tunneling protocol. L2TP supports multiple transfer protocols and remote access, but security is not strong. IPSec. IP layer can provide a strong security mechanism, but features such as multi-protocol support package deficient. Therefore, the integrated use of both, can both support each other to build a multi-protocol encapsulation, but also to provide authentication and encryption VPN[3].

L2TP and IPSec are on paper the two tunneling protocol to do a systematic exposition, and then each of the two tunneling protocol is proposed based on the principle of a combination of both to build a secure VPN approach, that is the first to use IPSec to encrypt data, and then data will be encrypted L2TP tunnel encapsulation. First introduced the model of this combined method to analyze structural model of the theory and obtained through the model structure to improve transport security, and then to write simulation software MATLAB source code of the program, the mode of data transmission speed the simulation, and finally by analyzing the simulation results show that the performance of this design

## II. VPN MEANING

Virtual Private Network, VPN is in the public network (such as the Internet) to establish a dedicated data communication network technology. In the virtual private network, the connection between any two nodes do not need to end the traditional private network of physical links required, but to use the resources of a dynamic group of public networks, private network out, the user is good with a special way to pass. More simply, VPN is the use of shared public network to establish a specific user data transmission channel, the user's remote


doi:10.4304/jnw.7.1.141-148

branch offices, business partners, such as linking mobile workers to provide end to end, a certain security and service quality-assured data communications services network technology. VPN can be connected using the network, it can be a single point device. With the traditional use of shared resources for voice company dedicated voice messaging service similar to the idea, VPN want to use the shared data protected network resources to provide dedicated data services.

"Virtual" means a better understanding of: the establishment of tunnels or virtual circuits to different physical networks or devices and is no longer using the physical establishment of a long line proprietary data network, but to build on widely distributed data network, such as ATM / FR / IP network. "Virtual" is relative in terms of DDN, although it seems to yourself from the user network, and in fact the user is superimposed on the public network in a virtual dimension. Between VPN and non-VPN, shared between multiple VPN network infrastructure, is different from the private network VPN main difference, it is "virtual" lies.

In order to facilitate the "special" definition of the concept, first define the Closed User Group, CUG concept. Closed for the group that many users of the site-specific formation of a closed user groups, user-based "switch" during the network's business to ensure communications and to prevent unauthorized access to these sites to other sites[4-5]. Within a specific CUG can use private addresses, the address space between different CUG can be reused. CUG This feature prevents unauthorized packets to a specific CUG spread within the network to prevent fraud and tampering of data packets in transit and other security attacks, and can be of different types of packets different treatment, statistics and billing, etc. .

The "private" at least two different ways of understanding:

1) CUG + QoS guarantee. The "special" as the circuit is connected to a closed user group or community, with QoS guarantee, but does not provide authentication and encryption and other security services. "Exclusive" means more emphasis on service performance in how not to consider security issues. Telecommunications experts and manufacturers generally are used to this understanding, such as ATM VPN, FR VPN, and the back will highlight the L2TP VPN.

2) CUG + security guarantees. The "special" as the connection is closed user group or community, but this time "dedicated" means more emphasis in the security services, in terms of performance only "best effort (best effort)" service, IP traffic between sites in the competing network resources. Computer experts and manufacturers generally prefer such an understanding, as will be described later IPSec VPN.

## III. VPN CATEGORIES

According to protocol type, VPN is generally divided into two categories: Based on the Layer Two Tunneling Protocol Layer-based VPN tunneling protocol VPN.

1) Based on Layer Two Tunneling Protocol: Layer Two Tunneling Protocol OSI model corresponds to the data link layer, using the frame as a data exchange unit. Layer Two Tunneling Protocol commonly used are PPTP, L2F, L2TP.

2) Based on the third layer tunneling protocols: Layer Two Tunneling Protocol OSI model corresponds to the network layer, use the package as a data exchange unit. The third layer tunneling protocol used has PISec, GRE. The second and third floor is the main difference between tunneling protocol user data in the first layers of network protocol stack is encapsulated.

Operators may participate in a variety of ways to a VPN to the management and implementation and, therefore, there are several types of VPN service, as shown in Figure 1, which is from the technical level (protocol stack) by the angle of the VPN type[6].

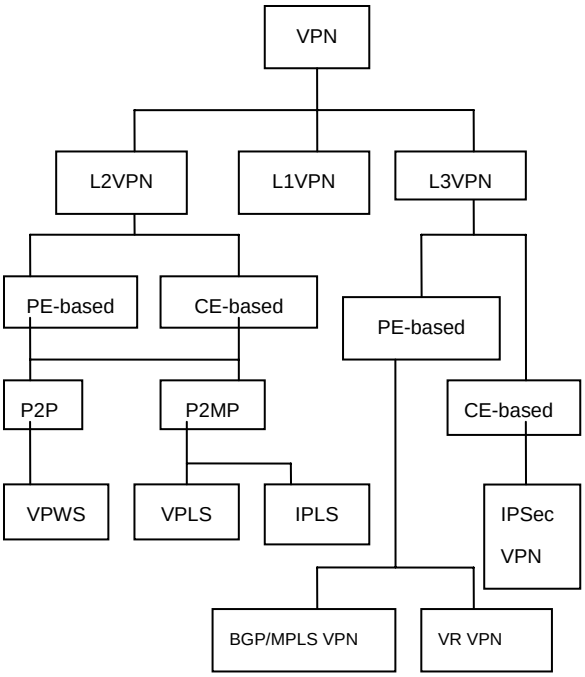



Figure 1 VPN service types

In the standard formulation, IETF, and more concerned about L3VPN L2VPN, research results are relatively more, IETF L1VPN although the research done, but the results much. ITU-T study in L1VPN put more effort to develop a large number of L1VPN standards.

## IV. VPN TUNNELING

In order to have transparency in IP networks transmit data packets, and provide some security and service quality assurance, then all the VPN must use one or more tunneling protocol. Tunnel technology uses a protocol to another protocol technology transfer, through the tunnel protocol. Tunneling Protocol provides for the establishment of the tunnel, maintenance, and deletion rules and how to package the enterprise network data

transmission in the tunnel, the following describes two commonly used tunneling protocol.

*A. Layer 2 Tunneling Protocol (L2TP)*

L2TP: Layer2 Tunneling Protocol PPTP and L2F protocols combines the advantages of a more comprehensive functional and technical[7]. It is also a PPP-based tunneling protocol, by IP, ATM, Frame Relay, and other public transport network to establish the tunnel to send the PPP packets.L2TP uses the client / server architecture, composed of two basic components, first, the client L2TP LAC (L2TP Access Concentrator), is used to initiate the call and the establishment of the tunnel; second server LNS (L2TP Network Server), provides a tunnel transmission services, but for all the end of the tunnel. In a traditional connection, the end user dial-up connection is the LAC, and the end of L2TP extends the PPP to the LNS.

*B. IP Security (IPSec)*

IPSec (IP Security) is the IETF IPSec working group in order to provide communication in IP layer security protocol developed by a family. It includes some security protocols, key agreement parts and security alliance. Section defines the security protocol for communications security protection mechanisms; key agreement part of the definition of how to protect the security protocol negotiation parameters, and how to communicate the identity of the entity identification; security alliance to store all the details of the consultations to be recorded.

Security protocol, including Encapsulation Security Payload, hereinafter referred to as ESP and the Authentication Header (AH) two. ESP protocol which provides for communication confidentiality, integrity protection; AH to provide integrity protection for the communications protocol[8]. The AH and ESP, has two operating modes: transport mode and tunnel mode.

IPSec protocol typically used have been designed with the algorithm-independent. Algorithm selection in the Security Policy Database (SPD) is specified.

## V. VPN SECURITY TECHNOLOGY MODEL

*A. IPSec VPN Connection Models*

In Figure 2, Site-to-site VPN configuration, each node is connected to separate networks, these networks are the other non-safety or public network isolation. As the security requirements of these networks partition, so unless otherwise deploy VPN client on the network node is not able to exchange data. This type of VPN configuration is "closed" Site-to-site network topology. Conversely, if connected to a network partition between the end nodes can freely exchange data and use other networks to forward and receive data. However, these non-secure data exchange. Therefore, IPSec can be used to guarantee some or all of the data exchange security. This type of VPN configuration is known as "open" Site-to-site network design. The key between the two is that, IPSec is used to implement data exchange gateway security. At the same time, more importantly, the realization of data exchange security safety net and connected to the security of end nodes is independent.

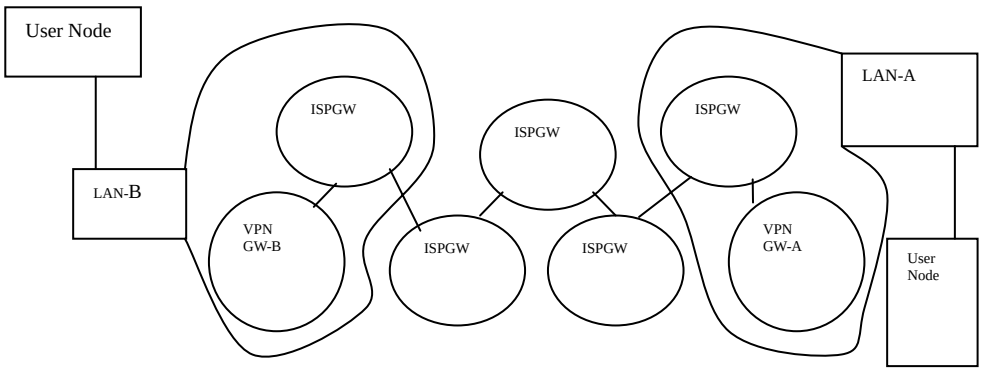



Figure2 Site-to-site VPN topology

In the Client-to-Site VPN network topology (see Figure 3), "open" and "closed" model is applicable. Isolated from the IPSec gateway (or its near) the connection between the nodes may be or not to be limited. In the "open" the Client-to-Site network topology, in the end nodes and the network path between the IPSec gateway is secure. In the "closed" network topology, in the end node and the gateway between the network access is secure. However, the client node and the IPSec gateway adjacent to (or after) the data exchange between the nodes only connected to the IPSec gateway to proceed.

Both network topology, the client node and the IPSec gateway architecture is similar ties in the traditional PSTN remote access dial-up networking. End nodes first establish a connection to the gateway, then the two nodes to communicate as IPSec. In addition, the gateway provides IP as a client node, this status allows the client to work with the other nodes IP and IPSec gateway directly connected end nodes adjacent to the network access. Client communication between nodes and the

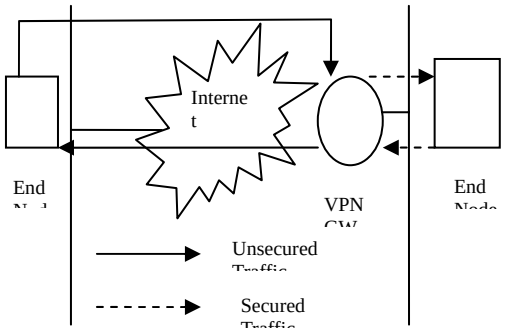



Figure 3 Client-to-site VPN topology

gateway is the same security is guaranteed by the IPSec. However, other IPSec client node and the gateway for communication between adjacent nodes are not safe.

*B. GRE tunnel model*

GRE tunnel is a traditional point to point connection. GRE as shown in Figure 4 applied to the enterprise network, you need the central node in the enterprise and between the various branches of the structure more than point to point GRE tunnels. When large enterprise branch offices, the allocation huge workload; and, if new branches, the central node on the need for additional configuration, an increase of the burden of network maintenance; In addition, branch offices with ADSL dial-up, etc., the branches of public Web address also increased the uncertainty of the center node configuration complexity. Although the dynamic VPN technologies, such as DVPN (Dynamic Virtual Private Network), can learn the public address branches and dynamically in the center tunnel between the nodes and branches, but the dynamic VPN technology is currently no uniform standard, various vendors private agreement with Dynamic VPN, can not communicate. Multipoint GRE tunnel solves the above problems, is ideal for branch offices of many corporate networks. Configuration in the central node multipoint GRE tunnels, the traditional branch-point GRE tunnel configuration, you can achieve a number of branches in the center between the node and dynamically created tunnel.

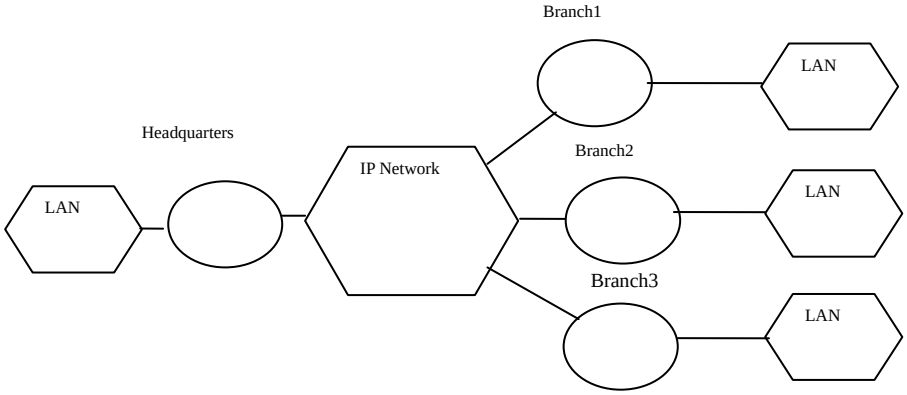



Figure 4 multipoint GRE tunnel topology

Multipoint GRE tunnels do not need to manually configure the tunnel destination address, but according to the received packet dynamic learning GRE tunnel destination address[9-11]. As shown below, the terminal device on the client device receives a GRE packet sent from the IP packet header to obtain the source address of the outer and inner payload (IP packet) source address, respectively, as the tunnel packet destination address and destination address (the branch network within the network address), the establishment of a tunnel entry. Among them, the packet destination address mask length can be manually configured. Forward through the multipoint GRE tunnel packet, the device according to the packet destination address, find the entry in the tunnel corresponds to the tunnel destination address, use this address as the external IP header encapsulation GRE destination address, as shown in Figure 5.

## VI. L2TP AND IPSec VPN INTEGRATION

L2TP and IPSec have the following deficiencies:
L2TP:

L2TP supports IP, IPX, Appletalk, and other network protocols, to support any of the wide area network technologies such as ATM, X.25 frame relay and any Ethernet technology. However, the L2TP protocol does not provide its own security mechanism, so the public network through the L2TP tunnel to transmit

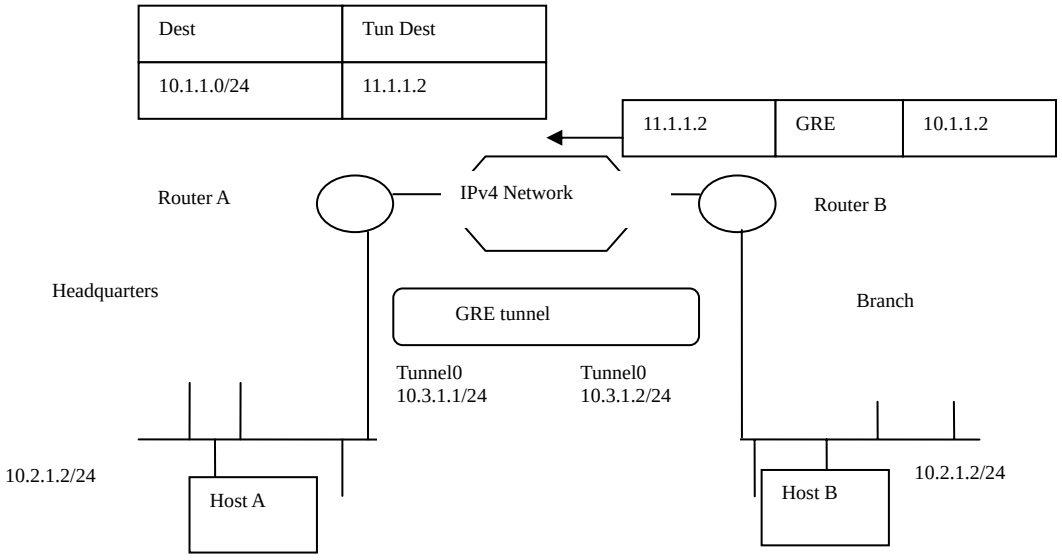



Figure 5 Dynamic Tunnel destination address

PPP business, L2TP control messages and data packets are vulnerable to attack. L2TP protocol has the following security issues:

1) L2TP is only defined on the end of the tunnel to authenticate an entity, rather than flowing through the tunnel authentication data of each packet, so that the tunnel could not resist inserting and address spoofing attacks.

2) Since there is no integrity check for each packet, there may have been DoS attacks, and send some bogus control information, resulting in L2TP tunnel or the PPP connection closes.

3) L2TP itself does not provide any encryption means, when the data require encryption, you need other technical support.

4) Although the PPP packets can provide confidential, but the PPP protocol does not support automatic key generation and automatic updates, so that eavesdropping attacker is likely to eventually break the key, resulting in the transmitted data.

Because the L2TP protocol has security problems, so when the specific arrangements for L2TP, and other agreements to be combined to maximize the reduction of L2TP security risks.

IPSec:

IPSec is the IETF IPSec working group to provide communications to IP layer security protocol developed by a family. It includes some security protocols and key agreement part of the IP data can flow integrity, confidentiality, anti-replay protection, etc., can also provide flexible connectivity level, identify the source of the data packet level. Currently, the Internet often use the Internet in building LAN IPSec between VPN, but not used for building remote access type independent VPN, main reasons are:

1) IPSec, while providing a strong host-level authentication, but it can only support a limited user-level authentication. Type in the remote access VPN in a remote end user to enter the internal network must be strict authentication. IPSec protocol currently can not easily and effectively implement this feature.

2) in the IPSec security protocol, always assume that the packet is encapsulated IP packet, it is not yet support multi-protocol encapsulation.

3) the current IPSec support only a fixed IP address to find the corresponding pre-shared keys, certificates and other identifying information, does not yet support dynamic allocation of IP addresses. The company staff are often away on business use of the telephone dial-up access to Internet network, when users are using dynamically assigned IP address, so can not authentication, access to the internal network.

After the above analysis, it is natural to have such thoughts: the L2TP protocol and IPSec protocols combine to make up L2TP using IPSec security deficiencies at the same time make use of L2TP IPSec in the user-level authentication, authorization and other deficiencies.

### *A. L2TP and IPSec integration of design*

The following building through the use of L2TP and IPSec-based VPN secure remote access example, to specific examples described. In this example, assume that the remote terminal and the internal network are using IP protocol, IPSec installed in the remote access terminal, the terminal used for remote access to internal network users to communicate with the internal IP packet before the IPSec security processing and then into PPP packet, L2TP encapsulation placed ISP access server-side, corporate intranets and the Internet connection protocol gateway that supports IPSec and L2TP protocol support.

First, the public network through a remote terminal access to dial-up ISP access server, PPP, PPP connection established between two points, ISP RADIUS server

using the tunnel completed by user identity authentication, if the user determine the user VPN, RADIUS server to the ISP access server provides the information needed to build the tunnel, the establishment of ISP and internal network access server gateway between the L2TP tunnel PPP connections through this tunnel extends from the remote terminal to the internal network gateway, the gateway receives a packet --- used for the control of user authentication information, the use of PPP protocol and improve user-level authentication function, the entrance to the internal network to the remote access user identity authentication, authentication is successful, the remote user and the gateway between the internal network --- transmission channel to set up, can be used for transparent transmission had already been dealt with IPSec security of user data, then the remote terminal and the gateway between the IPSec tunnel built in the IPSec security institutions under the protection of remote users and corporate intranets Users can secure data transmission. At this point the remote directly into the terminal just as the internal network within the same as the internal users to use the internal network resources[12-13].

L2TP tunnel work in two modes: the voluntary tunnel (Voluntary Mode) tunnel mode and active mode (Mandatory Mode). The LAC will be installed in different L2TP position, gave rise to two different modes of L2TP.

The first mode is integrated into the remote L2TP client, then client computer acts as a LAC. In this mode, the user independently of the L2TP configuration and management. Another model is installed in the L2TP NAS, usually ISP. In this mode, the client computer can not serve as the tunnel endpoint, but by the remote access server (such as NAS) as the tunnel endpoint, the user can transparently be L2TP service.

Agreement by the combination of IPSec and L2TP VPN security to improve the basic idea is: to integrate remote users or LAC IPSec client software, while the LNS is also integrated IPSec. Service software (integrated IPSec protocol LNS called Sceuer Remote Access Server, SRAS), which use IPSec. To improve the security of remote access communications[14-15]. IPSec protocol integrated in two ways, one is to IPSec remote access integrated into the host, known as the L2TP protocol secure voluntary model; one is integrated into the LAC on the IPSec, L2TP protocol as a mandatory security model.

1). Based on combination of L2TP and IPSec tunnel mode forced

Shown in Figure 6, when operating in forced mode, L2TP, L2TP tunnel is established between the LAC and LNS, but according to the security before LZTP analysis shows, LAC - LNS security between can not be guaranteed. L2TP tunnels in order to ensure better transmission of data security, we can use IPSec to provide security. In the LAC (SIP) and the LNS at the realization of IPSec, such as the LAC and LNS two security gateways, the public packet transmission network to provide security services.

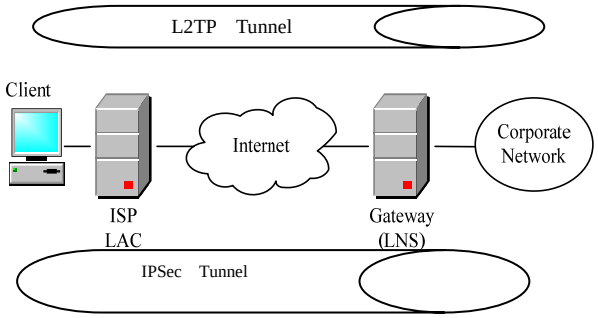



Figure 6 uses the IPSec tunnel mode force protection

2). Based on combination of L2TP and IPSec tunnel mode of the voluntary

Similarly, when the L2TP working in a voluntary mode, the client computer as LAC, the client computer at the realization of IPSec. Specific structure shown in Figure7.

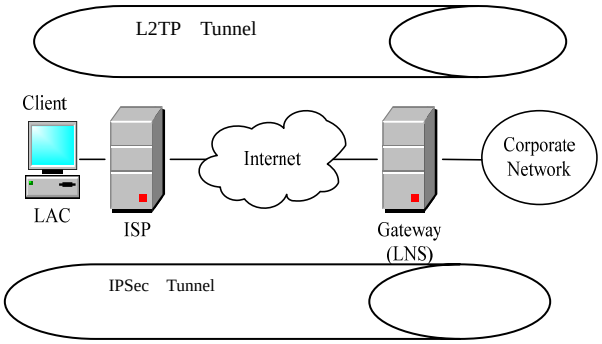



Figure7 the voluntary use IPSec tunnel mode protection

*B. L2TP and IPSec packet combining Encapsulation Format*

The logical structure of the L2TP, L2TP tunnels to ensure security, you can achieve in the LAC and LNS at IPSec, so the LAC and LNS into two security gateways to public data transmission network to provide security services reported. Assumes that all tunnels and link have been established.

Specific data flow is as follows:

1) IPSec package

IPSec-based security policies, IP packet by adding the IPSec Encapsulating Security Payload ESP header, trailer, and IPSec authentication trailer (Auth trailer), the IPSec encryption package.

2) PPP Package

IPSec packet processing by the PPP protocol encapsulation into PPP packets.

3) L2TP encapsulation and IP encapsulation

L2TP encapsulates PPP in groups of form L2TP packet,

and then add the formation of UDP packet UDP header, and then the formation of IP packets transmitted via the Internet in the PI.

Since IPSec has two modes: transport mode and tunnel mode, so the combination of IPSec and L2TP

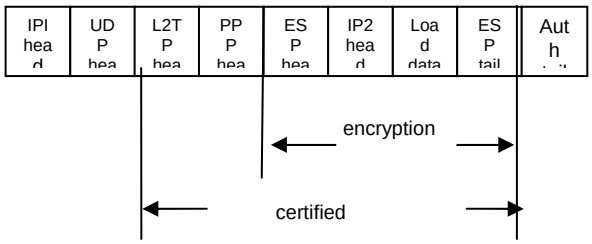



Figure 8 L2TP + IPSec transport mode encapsulation

package also has two modes, transport mode structures were shown in Figure 8 .

*C.L2TP a combination of VPN and IPSec works*

Remote users to communicate with the VPN as follows:
1) the remote user initiates a local call, then start a PPP connection to the LAC, LAC determine the type of business is the standard Internet services (such as www) or to the internal network. If it is required to access the intranet to find the corresponding LZTP tunnel and IPSee tunnel. And LAC on the PPP encapsulation and L2TP packets IPSee processing to generate L2TP + IPSec packets[16].
2) Once the tunnel is established successfully, the call to assign a Call ID. LAC LNS also send a link to the logo, the logo contains all the parameters have been negotiated.
3) LNS receives the packet, the first in the PI layer IPSec decryption and authentication processing, and then to the L2TP software. L2TP L2TP software will first remove and then sent to a virtual PPP interface, the interface to the PPP to the PPP head again after the call to the IP layer. Finally, according to the internal PI PI layer header sent to the VPN destination address within the network server.

The design, the remote user access to integrated IPSec protocol LAC, established a PPP connection, L2TP tunnels and IPSec tunnels, and their different scope, PPP role of the physical connection between the remote user and the LAC; PPP logical connection on remote between the user and the LNS; L2TP tunnel between the role of the LAC and LNS; IPSec tunnel between the role of the LAC and LNS.

*D. transmission time of three simulation packages*

Let the maximum transmission unit 500 bit, data transfer speed 20 bit / s, use slice transmission. Transmission time through the simulation of Figure 9, Figure 10, Figure 11, we can clearly see the load of the same length the circumstances under which the method of combining L2TP and IPSec transmission time the system overhead. This shows that the more complex security protection, the greater the overhead, improve security, system performance will be reduced. However, the security of data transmission has been greatly improved. Repeat this at the expense of visible transmission rate of packages to improve security to nature, this method is a relatively time-consuming, but in today's information age, especially in emerging network security risk situation, transmission time to increase the cost of some Data security is very necessary[17].

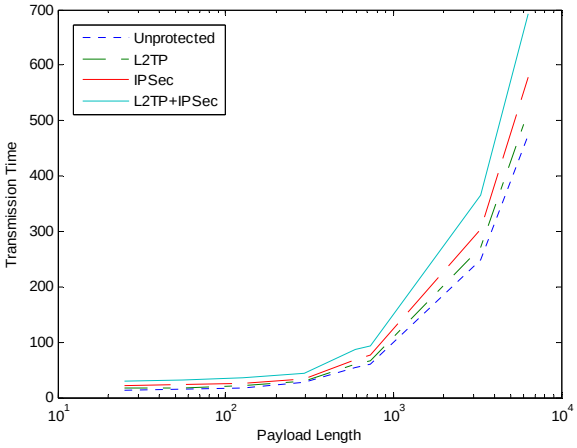



Figure 9 Transmission mode of transmission time simulation of Figure

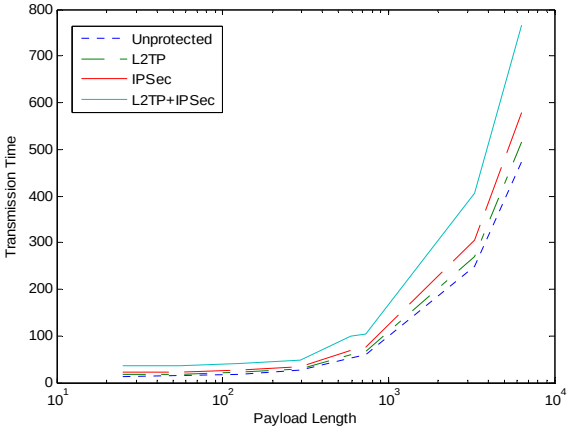



Figure 10 the tunnel mode of transmission time simulation of Figure

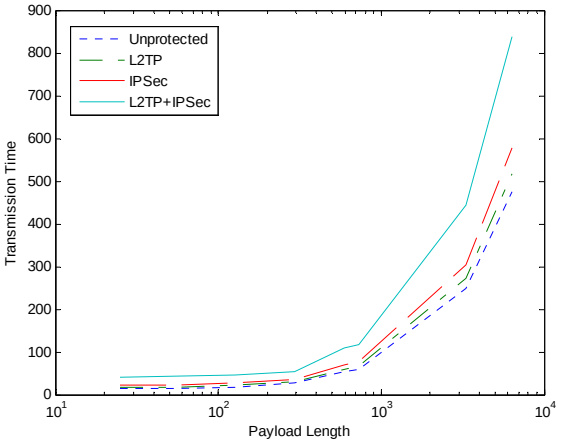



Figure 11 AH and ESP combined mode of transmission time simulation of Figure

Table 1 Different modes of data transmission time (s)

| Payload Length | Unprotected | L2TP | IPSec | L2TP+IPSec |
|---|---|---|---|---|
| 20 | 11.4 | 14.6 | 17 | 29 |
| 182 | 19.5 | 22.7 | 25.1 | 37.1 |
| 540 | 47.8 | 54.2 | 59 | 83 |
| 1180 | 90.2 | 99.8 | 107 | 142 |
| 4790 | 343.5 | 375.5 | 399.5 | 519.5 |
| 11610 | 830.1 | 906.9 | 964.5 | 1252.5 |

## VI. CONCLUSION

A secure VPN solution must be able to authenticate users and to strictly control that only authorized users can access VPN; be able to provide auditing and billing functions, show who and when to access what information; VPN solution must be able to assign a private network for users address and ensure the safety of the address; pass through the public Internet, data must be encrypted; VPN solution must be able to generate and update the client and the server's encryption key; On the integrated use of L2TP and IPSec, can only support each other to build a multi-protocol encapsulation, but also to provide authentication and encryption VPN. VPN solution must support the widespread use of the public on the Internet's basic protocols in ensuring quality of service at the same time, security is the primary VPN features.

## REFERENCES

[1] Dai Zongkun Tang Sanping. VPN and network security [M]. Beijing: Electronic Industry Press ,2006:4-6, 143
[2] the high sea and Britain. VPN technology [M]. Beijing: Mechanical Industry Press ,2005:8-12, 34
[3] [U.S.] Paul T. Ammann Xu six-way. Manage dynamic IP network (Chinese version) [M]. Beijing: Tsinghua University Press, 2004:563
[4] WANG Da. Virtual Private Network (VPN) fine solution. [M]. Beijing: Tsinghua University Press ,2007:12-43
[5] Jia-zhen Xu, Zi-meng Chen IPSec-based VPN with SSL-based comparison and analysis [J] Computer Engineering and Design, 2004,2 (2) :14-17.
[6] Qing-ming Jian SSL VPN and its secure remote access application [J] Sichuan Institute of Technology (Natural Science), 2005, 4 (1): 46-49.
[7] Carlton R. Davis Zhou Yongbin. IPSec: VPN security implementation. [M]. Tsinghua University Press ,2002:11-12
[8] Casey Wilson. Virtual Private Network creation and implementation. [M]. Machine Press ,2007:19-28
[9] Steven Brown. Build a virtual private network. [M]. People Post Press ,2007:22-34
[10] Sun-bo Xiang SSL VPN security analysis [J] Chengde Petroleum College, 2009,11 (1) :12-16
[11] Joshi JBD, Bertino E, Latif U. A Generalized Temporal Role-based Access Control Model [J]. IEEE Trans. On Knowledge
[12] Yuan Jue. King can. Cao Xiaomei. IPSec VPN application protocol. [M]. Computer Applications 2002-05
[13] Hongbo. VPN tunnel Technical Analysis Computer and Information Technology [J] 2002-03:33-36
[14] He Chaoxi. SINFOR Technology Senior Software Engineer DB (DB / OL) http://www.vpn.com.cn 2001-11 http://www.vpn.com.cn 2001-11
[15] Peterson, R., Interconectiong: Bridges and Routers, Ed.Addison-Wesley, 2000. Machinery Industry Press copy edition, 2004
[16] Deng Ming SSL VPN and security related introduced research [J] ISSN 1009-3044Computer Knowledge and Technology Vol.5, No.21, July 2009.
[17] You-xuan Chen, Guang-yu Wu of SSL VPN security issues and countermeasures [J] Yanbei Normal University, 2007, 23 (2) :22-26

**Ya-qin Fan** female, was born in 1963 in Shulan city, JiLin Province, China. She received her Bachelor Degree in Changchun Institute of Posts and Telecommunications in 1985. Her directions of research include the broadband communications technology and wireless communication theory.Yaqin Fan is an Associate Professor and also a Postgraduate Tutor, who is now working in Institute of Communications Engineering, Jilin University in China. Her lectures include the following specialized courses as "Modern Switching Technology" and "Computer Communication Networks", etc.Associate Professor Yaqin Fan had been in charge of many research projects and participated in a number of scientific research items in her research field, and published a number of valuable research papers in domestic and international research fields.

**Chi Li** male, was born in 1991 in Jilin City, Jilin Province, China. he is an undergraduate of Communication Engineering in Communication Engineering College Jilin University. Being interested in the research of broadband communications and wireless communication theory, Chi Li has taken effort on studying professional knowledge and participated in various research projects with the help of teachers.

**Chao Sun** female, was born in 1987 in Baicheng city, JiLin Province, China.she is an undergraduate of Communication Engineering in Communication Engineering College Jilin University. Being interested in the research of broadband communications .

**Project Title：Technology development plan project of jilin province china**
**Project number: 20090513**

# A New RFID Tag Code Transformation Approach in Internet of Things

Yulong Huang
School of computer science and engineering, South China University of Technology, Guangzhou, China
E-mail: h.yulong@mail.scut.edu.cn

Zhihao Chen
Applied Research Center, Motorola Inc. Hanover Park, IL USA
E-mail:chen@motorola.com

Jianqing Xi
School of Software Engineering, South China University of Technology, Guangzhou, china
E-mail: csjqxi@scut.edu.cn

***Abstract*: Along with the development of RFID technology, RFID tag is used for identifying object in more and more areas. RFID becomes one of the most important technologies in application and development of Internet of Things. Currently, in the applications of Internet-Internet of things, one of the most important issues is that the information carried by tag code can't be shared in different tag code standard systems and the tag code in different systems is incompatbile. In order to solve this issue, a new RFID tag code transformation approach for Internet of Things is proposed by this paper. With this approach, information sharing and exchanging between different tag codes standard systems becomes feasible and the global Internet-Internet of Things applications will be accelerated.**

***Index Terms*: Internet of Things, RFID technology, Tag code transformation, indirect transformation, incompatible tag codes, information sharing**

## I . INTRODUCTION

Along with the development of RFID technology [1], RFID tag is used to identify objects in more and more areas. With the advantages in the fields of logistics management [8] [9] and product packaging [10], RFID technology will be used instead of bar code technology in these fields and that will make it everywhere in future. At the same time, along with the development of the Internet, the trend of Internet development is to connect all the things in the world. Obviously, one of the most important technical foundations for the development of Internet-Internet of Things is RFID technology. From the Internet reports 2005[2], ITU indicated that one network which connects all the things in the world can be built by using RFID technology and Internet technology, and the information sharing and exchanging between different objects will be realized. Currently, the common definition of Internet of Things is that the Internet of Things is a ubiquitous network which combines various information sensing devices with the Internet. In order to exchange information between things and things (or between people and things), the RFID tags, sensors and bar codes, which attached to various things connected to the Internet by using code reader. Currently, one of the most important issues in the applications of Internet of Things is that different tag code standard systems are incompatible. To accelerate applications of global Internet of Things, we should transform the tag code between different code standard systems and solve the issues about the information sharing which between different systems and networks.

This paper is organized as follows: First, the existing basic problem of Internet of Things applications is introduced. Then the major tag code standards in the applications of internet of things and the related research works are described. After that, a new tag code transformation approach for Internet of Things is proposed and the performance of the approach is analyzed. Finally, our conclusion and future work is presented.

## II. EXISTING TAG CODE STANDARDS

There are several different tag code standards in the applications of Internet of Things, which will bring more difficulties to locate items and share information. For example, under the pervasive computing environment [11], the information carried by EPC tag in things is not understood by things in uCode standard system, and vice versa. Currently, the major international coding standards in the worldwide are EPC coding standard [4] proposed by EPCglobal[3] and uCode coding standard[5] proposed by uID-center. The national coding standard used within one country is mobile RFID coding standard proposed by NIDA, which is used for mobile RFID service [ISO/IEC CD 29174].As the code standards are playing important role in the fields of logistics management [8] [9] and product packaging in different countries and different industries, it can be imagined that there will still co-exist multiple coding standards in the future. In the next section, we will introduce these three main coding standards respectively.

### *A. EPCglobal's EPC standard*


doi:10.4304/jnw.7.1.149-156

EPCglobal is a non-profit organization co-founded by EAN and UCC in 2003.It dedicated to the development and promotion of relevant standards based on EPC network. EPCglobal's electronic product code standard [4] system is composed by different types of code standards. According to the code length, EPC code is divided into three types: EPC-64, EPC-96 and EPC-256. In order to ensure that the EPC code can uniquely identify all the items in the world while the prices of EPC tag are kept as low as possible, EPC-96 type code is recommended by EPCglobal. With this code type, it provides unique identity for each of 268 million companies in the world. Today, EPCglobal has launched several type of code scheme, such as: EPC-96 I Type, EPC-64 I type, EPC-64 II type, EPC-64 III type, EPC-256 I type, EPC-256 II type and EPC-256 III type. Different code schemes are listed separately in tableI.

TABLE I. EPC CODE SCHEME

| Coding System | Type | Header field | EPC Manager | Object Classification | Serial Number |
|---|---|---|---|---|---|
| EPC-64 | I | 2bits | 21bits | 17bits | 24bits |
| | II | 2bits | 15bits | 13bits | 34bits |
| | III | 2bits | 26bits | 13bits | 23bits |
| EPC-96 | I | 8bits | 28bits | 24bits | 36bits |
| EPC-256 | I | 8bits | 32bits | 56bits | 160bits |
| | II | 8bits | 64bits | 56bits | 128bits |
| | III | 8bits | 128bits | 56bits | 64bits |

In EPC code, the EPC managers are responsible to maintain and allocate the value of Object Classification field and Serial number within its scope. From table I, we know that the length of EPC manager field is 28 bits in EPC-96 I type coding scheme. This allows the EPCglobal to assign numbers to about 268 million manufacturers in the world. The length of Object Classification field is 24 bits and the capacity of this field can meet the requirement of assigning numbers to current inventory of the entire UPC units. Serial number field (whose length is 36 bits) provides unique identifier to all objects in the same type. The capacity of serial number field is 68719476736, which can be provided for each manufacturer about $1.1x10^{18}$ unique product IDs by the combination with the object classification field ---this greatly exceeds the current total of all the identified products in the world.

In EPCglobal tag data specification, EPC-96 type code includes: GID-96, SGTIN-96, SSCC-96, GSRN-96, SGL N-96, GRAI-96 and GIAI-96.These codes are derived from the codes which are defined in GS1 Specification (the detailed information is shown in [7]). For the detailed information, please refer to the document about EPC-global tag data standards [4];

### *B. uID-center's uCode standard*

uCode standard[5] is proposed by Japanese uID-center. T-Engine forum initiated the establishment of uID-center. The goal of uID-center is to establish and promote automatic identification technology in worldwide and build a ubiquitous computing environment. The basic length of uCode is 128 bits which can be expanded to 256 bits, 384 bits or 512 bits based on the different requirements. Since the uCode's length is 128 bits, which can meet the current and future requirements of growing of the number of objects, the uCode-128 scheme is recommended by uID-center. In this paper, the code transformation model is applied on this code scheme. Commonly, uCode is composed of five portions: version field, TLDC (Top Level Domain Code) field, CC (class code) field, DC (domain Code) field and IC (identification Code) field. The detailed code structure shown in table II:

TABLE II. uCode-128 CODE SCHEME

| Code scheme | Version | Top Level Domain code | Class Code | Domain Code | Identification Code |
|---|---|---|---|---|---|
| uCode-128 | 4 bits | 16 bits | 4 bits | Defined by CC | Defined by CC |

From table II, we know that the length of version field is 4 bits. The default value of version field in current uCode scheme is "$0000_2$".the length of TLDC is 16 bits. It is used to marking the top level domain which the uCode belongs to. And the maximum capacity of TLDC field is 65536.The UID-center is responsible for the management and value allocation of top-level domain field. UID-center can allocate the value of TLDC field to different countries or international organizations. In the current specification, several TLDC numbers are reserved by UID-center. The detailed information is shown in table III

TABLE III. RESERVED VALUES OF TLDC FIELD

| TLDC | Application |
|---|---|
| 0xe000 | Certificate standard encoding space |
| 0xfffe | Identity for space and time |
| 0xffff | eTRON identity |

The length of CC field is 4 bits. The first bit is used to indicate the total length of uCode(the number "1" indicates that the total length of uCode is 128 bits and the number "0" indicates that the total length of uCode is 256 bits or more). The last three bits is used to indicate the partition of DC field and IC field. The detailed information is shown in table IV:

TABLE IV. CORRELATION BETWEEN THE PREDIFINED VALUE OF CC FIELD AND THE PARTITION OF DC AND IC FIELD

| type | Class code | Length of Domain Code | Length of identification code |
|---|---|---|---|
| N/A | $1000_2$ | Reserved | |
| A | $1001_2$ | 8 | 96 |
| B | $1010_2$ | 24 | 80 |
| C | $1011_2$ | 40 | 64 |
| D | $1100_2$ | 56 | 48 |
| E | $1101_2$ | 72 | 32 |
| F | $1110_2$ | 88 | 16 |
| N/A | $1111_2$ | Reserved | |

From table IV, we know that partition of DC field and IC field are indicated by the value of CC field. Generally speaking, the value of DC field is allocated by the manager of TLDC field. Then the value of IC field is allocated by the manager of DC field. uCode is usually expressed by binary numbers and compatible with the codes which are defined in GS1 specification[7].

### *C. NIDA's mRFID standard*

mRFID code standard[6] is proposed by NIDA which aims at providing mobile RFID service in South Korea. The standard is now adopted by ISO certification organization [ISO / IEC CD29174]. Generally speaking, there are three different types of mRFID code: mCode, micro-mCode and mini-mCode. mCode is recommended by mRFID specification. The application scenario of Micro-mCode is similar with two dimension bar code. Mini-mCode is used in the area of small-capacity RFID tag. In this paper, the code transformation is applied on mCode. At the same time, mCode is also compatible with the code defined in GS1 specification [7].

The maximum length of mCode is 128 bits. Commonly, mCode is expressed in hexadecimal value string. mCode is composed of six fields: TLC(Top level Code)field, Class field, CC(company Code) field, ICC(item Category Code)field, IC(Item Code)field and SC(serial Code)field. The detail code structure is shown in table V:

TABLE V. THE STRUCTURE OF MCODE

| **mCode** | | | | | | | | | Description | |
|---|---|---|---|---|---|---|---|---|---|---|
| TLC (12 bits) | Class (4 bits) | CC+ICC+IC+SC | | | | | | | Length (bits) | Class name |
| | | 16 bits | 16 bits | 16 bits | 16 bits | 16 bits | 16 bits | 16 bits | | |
| $000_H$ | Reserved | | | | | | | | N/A | |
| $001_H$ ~ $EFF_H$ | $0_H$ | IC | | | | | | | 48 | A |
| | $1_H$ | CC | IC | | | | | | 64 | B |
| | $2_H$~$3_H$ | Reserved | | | | | | | 64 | C,D |
| | $4_H$ | CC | | IC | SC | | | | 96 | E |
| | $5_H$ | CC | IC | SC | | | | | 96 | F |
| | $6_H$ | CC | ICC | IC | | | | | 96 | G |
| | $7_H$~$E_H$ | Reserved | | | | | | | N/A | |
| | $F_H$ | Reserved for class extension | | | | | | | N/A | |
| $F00_H$ ~ $FFF_H$ | Reserved for other code structure | | | | | | | | N/A | |

From table V , we know that the length of TLC field is 12 bits. The value of TLC field is allocated to different countries or organizations by KISA which is authorized by NIDA. When the value of TLC field is $000_H$ or from $F00_H$ to $FFF_H$, mCode is reserved. The length of Class field is 4 bits and it indicates the partition of CC, ICC, IC and SC field in mCode. When the value of Class field is $0_H$, mCode only has a 32 bits IC field. When the value of Class field is $1_H$, mCode has a 16 bits CC field and a 32 bits IC field. When the value of Class field is 4H, mCode has a 32 bits CC field, a 16 bits IC field and a 48 bits SC field. When the value of Class is 6H, mCode has a 16 bits CC field, a 16 bits ICC field and a 48 bits IC field.

## III. TAG CODE TRANSFORMATION

### *A. Current Research for Tag Code Transformation*

Currently, researchers focus on the code transformation in Internet of Things to transform GS1 codes, and codes based on ISO14443 [13] and ISO15693 [13] into EPC code [4] as they are compatible [12]. That helps more and more deployments of applications in EPCglobal network. However, after development of years, uCode and mCode are becoming popular and important as EPC. uCode, mCode and EPC will become three major standard systems in Internet of Things. As we all know, the RFID

tag code standards are playing important roles in the fields of logistics management and product packaging in different countries and different industries. It can be imagined that there will still co-exist multiple coding standards in the future. As we mentioned before, the incompatibilities of those code standards hinders the development and applications of Internet of Things. The transformation between any two of those code standards become extreme important so that the information they carry is able to be shared and exchanged. The method [12] that to transform GS1 codes or ISO14443, ISO15693's code into EPC code directly is not able to be applied for the code transformation among EPC, uCode and mCode. One of the major reasons is that the method [12] is only applied between compatible systems. However, for incompatible tag code systems, the method in article [12] is not applicable. For example, the domain code field in uCode includes manufacturer information and product classification information. But, the EPCmanager field in EPC only contains manufacturer information. Moreover, the length of fields in different code systems is very different. To solve the problem of code transformation among incompatible tag codes and achieve the goal of sharing information among different tag code standard systems, we proposed an new tag code transformation method based on GS1 code in our transformation process: first, we transform tag code into GS1 code from any of EPC, uCode and mCode, then the new code will be transformed to the required code.

### *B. Tag Code Transformation Approach based on GS1code*

In this paragraph, we proposed a new tag code transformation approach for Internet of Things, which provides the information sharing and exchanging between incompatible tag codes standard systems and greatly accelerate the building and applying the global Internet of Things .Since the applications in EPC standard and uCode standard are used in more areas than those in mCode standard, we will transform the above three codes into EPC and uCode respectively. In this approach, there are two processes, one is to transform mCode and uCode into EPC, the other one is to transform mCode and EPC into uCode. In the next section, we will describe these two processes respectively.

#### *Process.1 Transform uCode, mCode into EPC*

In EPC network, information sharing and exchanging are mostly based on EPC code. The true global Internet of Things includes various code standards, in which the EPC system is more popular. To share and exchange the information in the tag code, we transform uCode and mCode into EPC. In the next section, we will demonstrate this process.

(1) Transform uCode-128 into EPC-96

As we mentioned before, the code structure between uCode-128 and EPC-96 are very different. We can't transform uCode-128 into EPC-96 directly. However both uCode-128 and EPC-96 are compatible with GS1 Code [7], we can first transform uCode-128 into a GS1 code with some extension components. Then, we transform GS1 Code into relative EPC-96.The whole transformation process is shown as follows:

**Assume:** the intermediate code in transformation process is a GS1 code G (for the detail code structure, please refer to document [GS1 GS] [7]) which extends the AI21 and AI250 component. At the same time, we assume mCode and uCode follow the GS1 code standard.

**Input:** a 128 bits uCode U

**Output:** a 96 bits uCode E

Transformation process:

**Begin**

*With the definition of uCode code standard, we can extract each field in code U. such as: U.Version, U.TLDC, U.CC, U.DC and U.IC;*

*G.AI250:=Transformation (U.Version)*$_{10}$*+Transformation (U.CC)*$_{10}$*;*//transform string U.Version and string U.CC into decimal string, then, combined the decimal string;

*G.CountryCode:=Transformation (U.TLDC)* $_{10}$*;*

//transform string U.TLDC into a decimal string

*If (G.CountryCode.length>the length which defined in GS1 specification) stop transformation process;*

*Temp: =Transformation (U.DC)*$_{10}$*;*//transform string U.DC into a decimal string;

*If (temp.length>the length which defined in GS1 specification) stop transformation process;*

*G.factoryCode:=substring (Temp);* //with the GS1 specification exacted the value of factoryCode field from string temp;

*G.ProductCode:=substring (Temp);*

//with the GS1 specification exacted the value of productCode field from string temp;

*G.AI21:=Transformation (U.IC)* $_{10}$*;* // transform string U.IC into a decimal string;

*E.EPCmanager:=Transformation(G.countryCode+G.factoryCode)* $_{2}$*;*//concatenated G.countryCode and String G.factoryCode, considering the result to be a decimal integer and transform the result into a binary string;

*If (E.EPCmanager.length<28) filled zero in front of string;*

*E.objectClassification:=Transformation (G.productCode)*$_2$*;*//transform string G.productCode into a binary string;

*If (E.objectClassification.length<24) filled zero in front of string;*

*E.serialNumber:=Transformation (G.AI21)*$_2$*;*//Transform string G.AI21 into a binary string;

*If (E.serialNumber<36) filled zero in front of string;*

*E.header:=Transformation (G.AI250)*$_2$*; //* transform string G.AI250 into a binary string;

*In order to make the E.header value not conflict with the predefined values in EPC specification, we can add a certain numeric string in front of the string E.header. Of course, we need delete this string in the transformation process which transforms EPC into uCode.*

**End;**

Since the check digit in GS1 code is not used in transformation process, we do not need to compute the check digit in our transformation process. The whole transformation process is shown in Figure1:

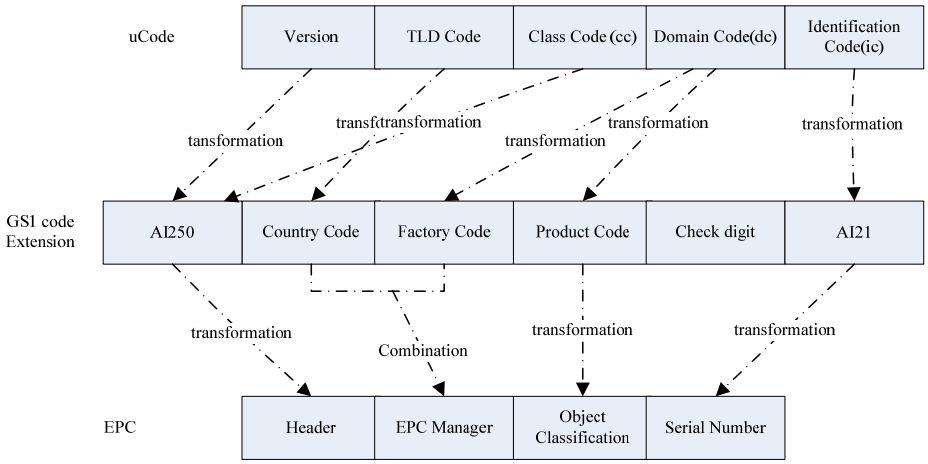



Figure1. The process of transforming uCode into EPC

(2) Transform mCode into EPC-96

From table V, we know that the value of Class field in mCode indicates the mCode structure. For example, when the value of Class field is $0_H$, the mCode only contains a 32 bits IC field. When the value of Class field is $5_H$, mCode contains a 16 bits CC field, a 16 bits IC field and a 48 bits SC field. In the transformation process, if some fields in mCode do not exist, we do not need to consider these fields in transformation process. In the following section, we will describe our transformation process using mCode in which the value of Class field is $5_H$. First, we transform a 96-bits mCode into a GS1 Code with some extension components. Then, we transform the GS1 code into a relative EPC. The whole transformation process is shown as follows:

**Assume:** the intermediate code in transformation process is a GS1 code G (detailed code structure refers to document [GS1 GS] [7]) which extends the AI21 and AI250 component. At the same time, we assumed that mCode and uCode follow the GS1 code standard.

**Input:** a 96 bits mCode M (in which the value of Class field is $5_H$)

**Output:** a 96 bits uCode E

**Transformation process:**

**Begin**

*With the definition of mCode code standard, we can extract each field in code M.such as: M.TLC, M.Class, M.CC, M.IC, M.SC; As mCode is expressed in hexadecimal value string, the value of these field are hexadecimal value string.*

*G.AI250:= Transformation (M.class)$_{10}$;*//transform string M.class into a decimal string

*G.countryCode:=Transformation (M.TLC)$_{10}$;*//transform string M.TLC into a decimal string

*If (G.countryCode.length>the length which defined in GS1 specification) stop transformation process;*

*G.factoryCode:=Transformation (M.CC) $_{10}$;*

//transformation string M.CC into a decimal string

*G.productCode:=Transformation(M.IC)$_{10}$;*//transform string M.IC into a decimal string;

*If (G.productCode.length>the length which defined in GS1 specification) stop transformation process;*

*G.AI21=Transformation (M.SC)$_{10}$;*//transform string M.SC into a decimal string;

*E.EPCmanager:=Transformation(G.countryCode+G.factoryCode)$_2$;*//concatenated G.countryCode and string G.factoryCode, considering the result to be a decimal integer and transform it into a binary string;

*If (E.EPCmanager.length<28) filled zero in front of string;*

*E.objectClassification:=Transformation (G.productCode)$_2$;*//transform string G.productCode into a binary string;

*If (E.objectClassification.length<24) filled zero in front of string;*

*E.serialNumber:=Transformation (G.AI21)$_2$;*//transform string G.AI21 into a binary string;

*If (E.serialNumber<36) filled zero in front of string;*

*If (E.serialNumber>the length which defined in EPC specification) stop transformation process;*

*E.header:=Transformation (G.AI250)$_2$;*//transform string G.AI250 into a binary string;

*If (E.header.length>the length which defined in EPC specification) stop transformation process;*

*In order to make the E.header value not conflict with the predefined values in EPC specification, we can add a certain numeric string in front of the string E.header.*

**End;**

Since the check digit in GS1 code is not used in transformation process, we do not need to compute the check digit in our transformation process. The whole transformation process is shown in Figure 2:

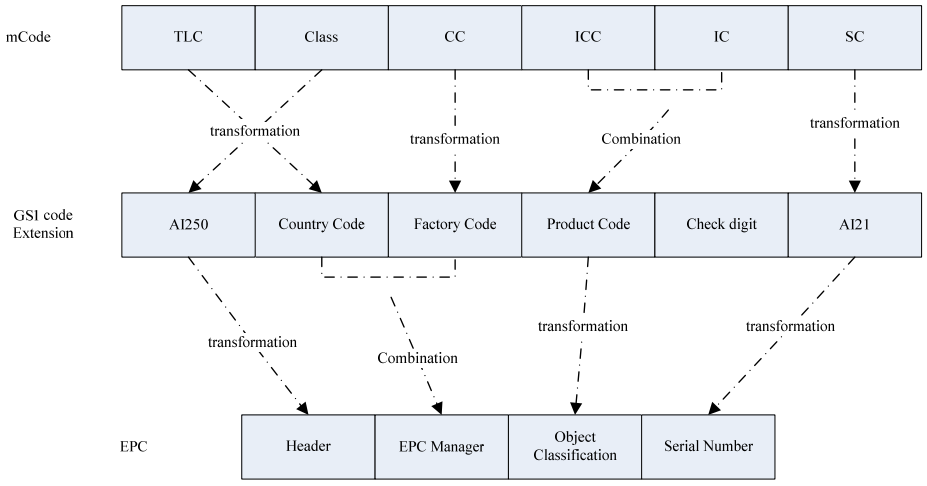



Figure2. The process of transforming mCode into EPC

*Process.2 Transform EPC, mCode into uCode*

To achieve the information exchanging between uCode code standard system and other code standard systems and help the interaction of uID network with other networks, we can transform EPC and mCode into uCode. The transformation process is exactly the opposite process described in *process.1* Section *A*. So in here, we may only describe the transformation process how to transforms the mCode into uCode. Because of the big difference between the structures of mCode and uCode, we can't transform mCode into uCode directly. Fortunately, we can use GS1 code with some extension components as an intermediate code in transformation process. As we mentioned before, the value of Class field indicates the code structure of mCode. In the following section, we will describe the whole transformation process using mCode in which the value of Class field is $5_H$. The process of transforming mCode into uCode is shown as follows:

**Assume**: the intermediate code in transformation process is a GS1 code G (for detailed code structure, please refer to document [GS1 GS] [7]) which extends the AI21 and AI250 components. We also assume that mCode and uCode follow the GS1 code standard.

**Input**: a 96 bits mCode M (the value of Class field is $5_H$).

**Output**: a 128 bits uCode U

**Transformation process:**

**Begin**

*With the definition of mCode standard, we can extract each field in code M. such as: M.TLC, M.Class, M.CC, M.IC, M.SC; mCode is expressed in hexadecimal value string;*

*G.AI250:=Transformation (M.class)*$_{10}$;//transform string M.class into a decimal string;

*G.countryCode:=Transformation (M.TLC)*$_{10}$;//transform string M.TLC into a decimal string;

*If (G.countryCode.length>the length which defined in GS1 specification) stop transformation process;*

*G.factoryCode:=Transformation (M.CC)* $_{10}$;

//transform string M.CC into a decimal string

*G.productCode:=Transformation (M.IC)* $_{10}$;

//transform string M.IC into a decimal string

*If (G.productCode.length>the length which defined in GS1 specification) stop transformation process;*

*G.AI21=Transformation (M.SC)* $_{10}$;

//transform string M.SC into a decimal string

*U.classCode:=Transformation (G.AI250)*$_2$;

//transform string G.AI250 into a binary string

*If (U.classCode.length>the length which defined in uCode specification) stop transformation;*

*U.classCode:="1"+U.classCode;*//added the character '1' to the front of string.

*U.version:="XXXX";*//among which, at least has one bit character is '1';

*U.domainCode:=Transformation(G.factoryCode+G.productCode)*$_2$;

//concatenated G.factoryCode and string G.productCode, considering the result to be a decimal integer, then transform it into a binary string;

*If (U.domainCode.length<the length which defined in uCode specification) filled the zero in front of string;*

*Else if (U.domainCode.length>the length which defined in uCode specification) stop transformation process.*

*U.TLDC:=Transformation (G.countryCode)* $_2$;

//transform string G.countryCode into a binary string

*If (U.TLDC<the length which defined in uCode specification) filled the zero in front of string;*

*Else if (U.TLDC>the length which defined in uCode specification) stop transformation process;*

*U.IC:= Transformation (G.AI21)*$_2$;

//transformation string G.AI21 into a binary string

*If (U.IC<the length which defined in uCode specification) filled the zero in front of string;*

*Else if (U.IC>the length which defined in uCode specification) stop transformation process;*

**End;**

Since the check digit in GS1 code is not used in transformation process, we do not need to consider the check digit in our transformation process. The whole transformation process is shown in Figure 3:

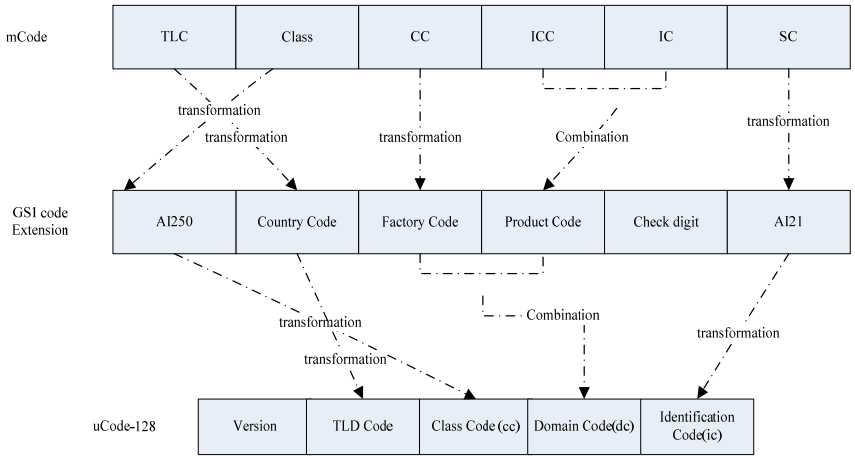



Figure.3 The process of transforming mCode into uCode

## IV. PERFORMANCE ANALYSIS

To prove the effectiveness of our tag code transformation approach, we conducted simulation experiments with our approach. Because the existing code transformation approach [12] can not apply to the code transformation between incompatible tag codes. So in here, we only verify the effectiveness of our code transformation approach. In our experiment, the hardware platform specifications are: CPU is Intel Pentium Dual Core 2.16GHz; memory capacity is 2GHz. Operating system is Windows XP Professional. With Myeclipse7.1 development tool, we built a prototype system of our tag code transformation approach. With the prototype system, we verified the effectiveness of our approach. Here is our experiment process: First, with the different tag code standards, we generate numbers of tag codes. Then, we transform these tag codes into different tag code standards with our prototype system. In this case, the performance analysis result using our approach is obtained, shown in Table VI.

TABLE VI. THE PERFORMANCE ANALYSIS OF TAG CODE TRANSFORMATION APPROACH

| Transform velocity scheme / Code numbers | Transform uCode into EPC | Transform EPC into uCode | Transform mCode into EPC | Transform mCode into uCode |
|---|---|---|---|---|
| 100 | 31ms | 16ms | 15ms | 16ms |
| 500 | 47ms | 31ms | 28ms | 47ms |
| 1000 | 63ms | 47ms | 47ms | 62ms |
| 5000 | 157ms | 125ms | 78ms | 181ms |
| 10000 | 297ms | 219ms | 141ms | 234ms |
| 50000 | 1375ms | 1016ms | 609ms | 1109ms |
| 100000 | 2719ms | 1969ms | 1198ms | 2172ms |

In the experiments of transforming uCode to EPC, it spends 31ms on transforming of 100 codes, 47ms on 500 codes, 63ms on 1000 codes, 157ms on 5000 codes, 297ms on 10000 codes, 1375ms on 50000 codes and 2719ms on 100000 codes. The data above show the effectiveness of our code transformation approach. Though, these data are obtained under experiment environment, there would not be much different in the real application.

## V. CONCLUSION AND FUTURE RESEARCH

Along with the development of RFID technology, the applications of Internet of Things gradually obtain more and more attentions in academic and industrial research. There exist many different tag code standards in the Internet of Things. Moreover, the information carried by different tag code standard systems can't be exchanged. Our tag code transformation approach in Internet of Things provides a solution for this issue. With this approach, the information exchanging between incompatible codes standard systems become possible. And that accelerates the development and applications of global Internet of Things. In the future, we would like to cover all the tag code standard systems, for example, including ISO RFID standards. So that things in Internet can share and exchange information with each other.

## ACKNOWLEDGMENTS

We are grateful to NIDA's Yi, Seung-Jai for giving us some explanations about mCode structure. This research supported by Science and Technology Planning Project of Guangdong Province, China (No. 2009B050700008) and the Fundamental Research Funds for Central Universities, SCUT (No. x2rjD210435);

## REFERENCES


[1] Landt, J. The history of RFID.IEEE Potentials, 2005, 24(4), pp: 8−11;

[2] ITU. Internet Reports 2005: The Internet of Things—executive summary, 2005:2;

[3] EPCGlobal Inc., EPCglobal Architecture Framework v1.2; http://www.epcglobalinc.org/standards/architecture/architecture_1_2-framework-20070910.pdf

[4] EPCglobal Inc., Tag Data Standards v1.0.3; http://www.epcglobalinc.org/standards/tds/tds_1_3-standard-20060308.pdf

[5] Sakamura K. Ubiquitous ID technologies, 2008. http://www.uidcenter.org/pdf/UID910-W001-080226_en.pdf

[6] NIDA, mobile RFID Code Architecuture; https://www.mcode.kr:444/rfid_doc/english/sub01/mcode.jsp?page_pos=1

[7] GS1: GS1 General Specification v10. http://www.gs1india.org.in/gs1barcodes/WORD_files/GS1_General_Specifications_v10.pdf

[8] R.Agrawal,etc.,TowardsTraceability across Sovereign, Distributed RFID Databases. Proceedings of the 10th International Database Engineering and Applications Symposium, 2006, pp: 174-184.

[9] Zaheeruddin Asif, etc. integrating the supply chain with RFID: a technical and business analysis. Communications of AIS, 2005, 15(2);

[10] Bo Yan, Guangwen Huang. Supply Chain Information Transmission based on RFID and Internet of Things. International Colloquium on Computing, Communication, Control, and Management, 2009, pp: 166-169;

[11] Robert Grimm, Tom Aderson.etc, A System Architecture for Pervasive Computing. Proceedings of the 9th workshop on ACM SIGOPS European workshop, 2000, pp: 177-182;

[12] Lo¨ıc Schmidt, Nathalie Mitton, etc.Towards Unified Tag Data Translation for the Internet of Things. Wireless VITAE' 09:2009.

[13] DIN: A Survey of Card Standards. http://www.din.de/sixcms_upload/media/2896/Survey%20of%20Card%20Standards.pdf



**Yulong Huang** is a Ph.D candidate of South China University of Technology. He has received B.S degree in Computer Science and Technology College from Wuhan University in 2002 and received M.S degree in computer Science and engineering college from Guizhou University in 2008.His current research interests include Internet of Things. Database Technology and Distributed System.

**Zhihao Chen** is a senior research scientist at Motorola ARC (Applied Research Center). He has been working with software and system engineering issues since 2000. He has a PhD from the University of Southern California. His recent research concerns software engineering about policy/rule system, event-driven service oriented architecture, and modeling and learning with a particular focus on light weight modeling methods. His doctoral research aimed at improving the validation of, possibly inconsistent, knowledge-based systems, and empirical methods and model integration. His research areas extend from software and system engineering to data mining and machine learning fields.

**Jianqing Xi** is a professor of software engineering. He has received his MSc in software engineering and PhD in computer architecture from national university of defense technology in 1988 and 1992 respectively. His main research interest is in Database and Data Warehouse, Distributed system based on P2P technology, Chinese information process, Data Management and Software Development technology

# A Distributed Trust Evaluation Model for Mobile P2P Systems

Xu Wu
Department of Computer Science, Xi'an University of Posts and Telecommunications, Xi'an, China
Email: xrdz2006@163.com

*Abstract*—**Security is one of key factors which influence the development of mobile P2P systems. However, traditional security techniques cannot be applied directly to a mobile P2P network due to some of its characteristics such as heterogeneous nature of the peers, limited-range as well as unreliability of wireless links. In the paper we propose a distributed trust evaluation model, which helps the systems to operate normally with high probability. The model uses a polling protocol and seven metrics to real-time evaluate the reputation of mobile peers. The model exhibits three interesting features not seen in previous works. Firstly, it considers voting for peers from the perspectives of both trust and distrust. This appears to be the first attempt to incorporate distrust in the polling algorithm. Secondly, it credits/penalizes a peer according to its interaction behaviors, the size of interaction and the vote accuracy. This mechanism of credit and penalization is expected to deter dishonesty or misbehavior by the entities involved. Thirdly, it effectively solves the trust problem when no prior interaction history exists, an issue that has not been addressed in many models. In the end, the model is shown to be efficient and robust in the presence of attackers through simulation.**

*Index Terms*—**trust, P2P, distributed, mobile**

## I. INTRODUCTION

With the deployment of high bandwidth 3G (and expected deployment of 3.5G and 4G) cellular networks and wireless LANs, there is an increasing interest in wireless P2P networks. A wireless mobile network is a cooperative network where each node requires collaborating with each other to forward packets from a source to a destination. It is often termed an infrastructure-less, self-organized, or spontaneous network. In mobile P2P systems, each peer acts as both client and server to share its resources with other peers, and communicates with each other via unregulated, short-range wireless technologies such as IEEE 802.11, Bluetooth, or Ultra Wide Band (UWB) [9]. It is obvious that mobile P2P systems are different from the wired ones, since each object is able to move around and each has a limited radio range. Compared to a fixed peer-to-peer system, the mobile network environment is more distributed, with wider participants and more autonomic than the fixed P2P system.

However, traditional security techniques cannot be applied directly to the mobile P2P systems due to the limitations of the wireless medium, expensive bandwidth, and the limitations of the mobile devices due to small memory and limited computation power [1]. Therefore, computation-intensive techniques like public-key cryptography are not expected to be used in mobile P2P systems. Such a distinction is also beyond the ability of the conventional key management scheme because we cannot guarantee the secrecy of each peer's private key. In addition, mobile devices are susceptible to a variety of attacks for example, eavesdropping, denial of services, wormhole, and Sybil attack. Even a few malicious peers can easily spread deceitful data and make the systems be in confusion without great efforts. Therefore, some smart trust management schemes are needed to identify trustworthiness of mobile peers in order to distinguish between malicious peers and innocuous peers, and to strengthen reliable peers and weaken suspicious peers.

The idea of constructing a trust-based scheme is motivated from existing human societies in the world. Embedded in every social network is a web of trust; with a link representing the trustworthiness between two individuals. When faced with uncertainty, individuals seek the opinions of those they trust. The intent is to develop a similar trust management mechanism for P2P Systems, where peers maintain trust value for other peers. This trust value is used to evaluate the trustworthiness of other peers. This establishes a web of trust in the network, which is then used as an inherent aspect in predicting the future behavior of peers in the network. Because mobile P2P systems pose some unique challenges, not every current trust evaluation models [2-6] are applicable to mobile P2P systems properly. In the paper we propose a distributed trust evaluation model, which helps the systems to operate normally with high probability. The model uses a polling protocol and seven metrics to real-time evaluate the reputation of mobile peers. The model exhibits three interesting features not seen in previous works. Firstly, it considers voting for peers from the perspectives of both trust and distrust. This appears to be the first attempt to incorporate distrust in the polling algorithm. Secondly, it credits/penalizes a peer according to its interaction behaviors, the size of interaction and the vote accuracy. This mechanism of

Manuscript received January 1, 2011; revised June 1, 2011; accepted July 1, 2011.
This work has been supported by Scientific Research Program Funded by Natural Science Basis Research Plan in Shaanxi Province of China (Program No.2011JQ8006) and Shanxi Provincial Education Department (Program No.11JK1060).


doi:10.4304/jnw.7.1.157-164

credit and penalization is expected to deter dishonesty or misbehavior by the entities involved. Thirdly, it effectively solves the trust problem when no prior interaction history exists, an issue that has not been addressed in many models.

The rest of the paper is organized as follows. Section 2 describes related work. Section 3 presents the proposed trust model. Section 4 contains experimental study. Finally, we conclude this paper in Section 5.

## II. Related Work

Trust-management approach for distributed systems security was first introduced in the context of Internet as an answer to the inadequacy of traditional cryptographic mechanisms. Some of the notable earlier works in this domain have been trust-management engines such as RT framework [10]. Since then many trust models based on reputation have been proposed for P2P networks, such as EigenTrust [2], PowerTrust [3], Bayesian network-based trust model [4] and so on. Yet, little has been done to show how trust and distrust can be incorporated into a trust model to yield an outcome that is beneficial to trust computation. In fact, even a small amount of information about distrust can tangibly help judge about a peer's trust. The proposed mechanism does borrow some design features from several existing works in literature but as a complete system differs from all the existing reputation-based systems.

EigenTrust [2] model is designed for the reputation management of P2P systems. The global reputation of peer *i* is marked by the local trust values assigned to peer *i* by other peers, which reflects the experience of other peers with it. The core of the model is that a special normalization process where the trust rating held by a peer is normalized to have their sum equal to 1. The shortcoming is that the normalization could cause the loss of important trust information. Runfang Zhou and Kai Hwang [3] proposed a power-law distribution in user feedbacks and a computational model, i.e., PowerTrust, to leverage the power-law feedback characteristics. The paper used a trust overlay network (TON) to model the trust relationships among peers. PowerTrust can greatly improve global reputation accuracy and aggregation speed, but it can not avoid the communication overhead in global trust computation.

A Bayesian network-based trust model [4] proposed by Wang and Vassileva uses reputation built on recommendations in P2P networks. The work differentiates between two types of trust, trust in the host's capability to provide the service and trust in the host's reliability in providing recommendations. A new trust model based on recommendation evidence is proposed for P2P Networks by Tian Chun Qi et al [5]. The proposed model has advantages in modeling dynamic trust relationship and aggregating recommendation information. It filters out noisy recommendation information. Though the trust model filters out noisy recommendation information, the algorithm is complex algorithm considering the complexity of the algorithm design and the workload in the system running.

Thomas Repantis and Vana Kalogeraki [6] propose a decentralized trust management middleware for ad-hoc, peer-to-peer systems, based on reputation. In the work, the middleware's protocols take advantage of the unstructured nature of the network to render malicious behavior, and the reputation information of each peer is stored in its neighbors and piggy-backed on its replies. The proposed approach allows peers to calculate a local trust for other peers with the reputation information which is collected by flooding reference trust requests to peers' friends. However, in large scale mobile P2P networks, flooding mechanism is not scalable.

Recently, there are many approaches studying trust management of wireless systems. The significant efforts done so far are to manage trust with the help of Certificate Authority (CA) or Key Distribution Center (KDC). A CA/KDC is responsible for setting up the foremost trust relationships among all the nodes by distributing keys or certificates [7]. Then, CA's functionality is substituted by t sub-CAs using threshold cryptography, and these t sub-CAs will issue partial certificates afterwards. In the solution, a group of n servers together with a master public/private key pair are firstly deployed by CA. Each server has a share of the master private key and stores the key pairs of all nodes. The shares of master private key are generated using threshold cryptography. Thus, only n servers together can form a whole signature. For any node wanting to join the network, it must first collect all of the n partial signatures. Then the node can compute the whole signature locally and thus get the certificate. However, this strategy suffers from difficulty on collecting t certificates efficiently.

In the distributed CA scheme [8], Kong et al. mentioned that the trust between a to-be-member node and t member nodes in its neighborhood can be established by out-of-bound physical proofs, such as human perception or biometrics. However, we can find that this method is far from practical. It is obviously impossible for a node acquiring t nodes to trust it in its local communication range, because the trust evidence should be evaluated and authenticated, or there should exist off-line trust relationships between this node and the t member nodes. In an infrastructureless mobile network environment, the evaluation of trust evidence will be very hard.

A novel trust evaluation model to be proposed in this paper has addressed these weaknesses. Namely, the model has defined clear criteria for trust calculation. A mobile peer can select one or more peers to interact based upon the criteria such that the aggregated trust value of these selected *peers* satisfies the trust level specified by the mobile peer owner. The trust level is, in turn, determined by the value of the interaction to be performed. In addition, the mobile peer owner calculates and updates trust values associated with each of the peers involved in an interaction by Trust Updating algorithm.

## III. The Trust Metrics

In the section we firstly present the definition and type of trust, then introduce seven trust factors which influence the trust in such a mobile environment.

### A. The definition and type of trust

Trust plays a role across many disciplines, including sociology, psychology, economics, political science, history, philosophy, and computer science. As such, work in each discipline has attempted to define the concept. The problem with defining trust is that there are many different types of trust and it means something different to each person, and potentially in each context where it is applied. In order to facilitate making computations with trust in mobile P2P systems, we propose the following definition: trust in a peer is a commitment to an action based on a belief that the future actions of that peer will lead to a good outcome. This definition forms the foundation for identifying main factors that influence the trust in mobile P2P systems, and how it can be used in computation. Our trust model has two types of trust: direct trust and recommendation trust. Direct trust is the trust of a peer on another based on the direct interacting experience and is used to evaluate trustworthiness when a peer has enough interacting experience with another peer. On the other hand, recommendation trust is used when a peer has little interacting experience with another one. Recommendation trust is the trust of a peer on another one based on direct trust and other peers' recommendation.

To figure axis labels, use words rather than symbols. Do not label axes only with units. Do not label axes with a ratio of quantities and units. Figure labels should be legible, about 9-point type.

Color figures will be appearing only in online publication. All figures will be black and white graphs in print publication.

### B. The trust factors

The mobile network environment is more distributed, with wider participants and more autonomic than the fixed P2P network. Since there is no centralized node to serve as an authority to monitor and punish the peers that behave badly, malicious peers have an incentive to provide poor quality services for their benefit because they can get away. An important question raised is how can a mobile peer owner decide one or more peers to interact with it? In other words, what should be the selection criteria on which the mobile peer owner makes the interaction decision. Obviously, the mobile peer owner should select the *peers* that have an acceptable level of trust. In our distributed trust model, a *peer*'s trust is measured by a *trust level*. The value is the result of the *peer*'s aggregated interactional behaviors (reflected by its responses) over a specific past period. The trust level reflects the truthfulness of the *peer* in performing the interactions.

The value is the function of the following parameters.

**Satisfaction or dissatisfaction degree in interactions:** When a mobile peer finishes an interaction with another peer, the mobile peer will evaluate its behavior in the interaction. The result of evaluation is described using satisfaction or dissatisfaction degree which is in the range (-1, 1). Satisfaction and dissatisfaction degrees express how well and how poor this peer has performed in the interaction, respectively. Satisfaction or dissatisfaction degree can encourage interacting sides to behave well during interactions. However, it is sufficient to measure a peer's trustworthiness without taking into account the number of interactions.

**Number of interactions performed:** Some peers have a higher interaction frequency than some other peers due to a skewed interaction distribution. A peer will be more familiar with other peers by increasing the number of interactions. A simple aggregation of feedbacks may fail to capture the true record of a *peer*'s interactional behavior. For example, a *peer* that has performed dozens of interactions but cheated on 1 out of every 4 occasions will have a higher aggregated trust value in comparison with a *peer* that has only performed 10 interactions and has been faithful in all of these occasions. In other words, the total number of interactions that the *peer* has performed over the specific past period is also an important indicator of its trust and should be taken into account for the calculation of its trust value. In our model, the average feedback value, measured as the ratio of the sum of the feedbacks the *peer* has received over time period $T_h$ to the total number of interactions the *peer* has taken part over the same period, is used instead of a simple sum.

**Size of interactions**: Size has different meanings in different P2P environments. In a P2P file sharing network, the size of interaction expresses the file size shared in each interaction, while in a P2P business community, it shows the sums of money involved in each interaction. We know that helping with an interaction with a value of $1000 certainly worth more credits than that with a value of $10. Similarly, failure to perform an interaction of £1000 should be penalized more than failing a $10 one. Size of interactions is an important factor that should be considered in the trust model. For peers without any interacting history, most previous trust models often define a default level of trust. But if it is set too low, it would make it more difficult for a peer to show trustworthiness through its actions. If it is set very high, there may be a need to limit the possibility for peers to "start over" by re-registration after misbehaving. In our trust model, the introduction of the size of interactions effectively solves the trust problem of peers without any interacting history. An example of Size of interactions is given in Table 2.The details will be described in the next part.

**Time**: The influence of an interacting history record to trust always decays with time. The more recent interactions have more influence on trust evaluation of a peer. For instance, if peer X has interacted with peer Y for a long time, the change of trust degree influenced by the interaction three years earlier is weaker than that of today. In our trust model, we introduce time factor to reflect this decay, that is, the most recent interaction usually has the biggest time factor.

**Vote accuracy**: We use a distributed polling algorithm [1] to collect peer's reputation information in our model. Vote accuracy factor reflects the accuracy that a peer votes for other peers. For example, if a peer correctly votes for other peers, it will have a high vote accuracy factor. The purpose of introducing this factor is to encourage peers to vote actively and correctly in our model. As if their suggestions are more worthy of belief, people are always honest in the real society.

**Punishment function**: The measurement of trust is the accumulation of the effects of interactions, both positive and negative. We not only consider the decay of influence of the interaction experiences with time, but also punish malicious actions. Punishment should be involved by decreasing its trust degree according to the amount of malicious behaviors. Therefore we introduce the punishment factor in our model to be used to fight against subtle malicious attacks. For instance, if a peer increases its trustworthiness through well-behaving in small-size interactions and tries to make a profit by misbehaving in large-size interactions, the peer would need more successful small-size interactions to offset the loss of its trust degree.

**Risk**: Every peer has its own security defense ability which is reflected by risk factor, such as the ability to detect vulnerabilities, the ability to address any viruses and to defend against intrusions.

## IV. A Distributed Trust Evaluation Model

To dynamically select a subset of trusted peers among a set of $N$ peers, $\{trusted\ peer_i,\ i \in \{1, ..., N\}\}$, based upon their real-time interactional behavior to assist a mobile peer to perform security sensitive tasks, a distributed trust evaluation model is required. It includes two steps: trust computing and updating. A mobile peer can select one or more peer to interact if the aggregated trust value of *these selected peers* satisfies the trust level specified by the mobile peer owner. The mobile peer owner updates trust values associated with each of the peers involved in an interaction based upon the feedback received.

### A. Assumptions

The Distributed Trust Evaluation Model is designed based upon the following assumptions:

TABLE I.
A Example of Table TV

| $P_i$-ID | $Trust_i$ | $Sat_i$ | $TotalInter_i$ | $Fail_i$ | $Size_i$ |
|---|---|---|---|---|---|
| $P_2$ | 0.6 | 0.6 | 20 | 1 | \$1000 |
| $P_3$ | 0.5 | 0.3 | 15 | 2 | \$100 |
| $P_4$ | 0.8 | 0.7 | 35 | 1 | \$100 |

Every mobile peer owner maintains a table TV (Trust Valuation) containing trust value associated with each of the *peers* that the agent owner has dealt with in the past period $t_h$. An example TV is given in Table 1. In the table, each row corresponds to one *peer* containing six attributes: {$P_i$-ID, $Trust_i$, $Sat_i$, $TotalInter_i$, $Fail_i$, Size}.

The $P_i$-ID is the unique identifier of $peer_i$. $Trust_i$ is its aggregated trust value, respectively. The $Trust_i$ attribute indicates the level of $peer_i$'s trustworthiness (or honesty) in performing its job. It is assumed that the initial value of $Trust_i$ is set to zeros indicating that the mobile peer has not yet had any experience in dealing with the *peer*. The initial value is greater than the values indicating a malicious *peer* ($\leq$ -1). In this way, a newly deployed *peer* will not be treated unfairly. $Sat_i$ refers to the average value of $Trust_i$, i.e. $Sat_i = \lambda \times Trust_i$, ( $-1 \leq \lambda \leq 1$). The parameter, $\lambda$, represents the impacts of $Trust_i$ in calculating the value of $Sat_i$, respectively. The choice of value given to the parameter is of the mobile peer owner's preferences. For example, if a mobile peer owner feels that $Trust_i$ should weigh more, then he may assign 0.7 for $\lambda$ (the value is used in the example given in table 1). The higher the value of $Sat_i$, the more confidence the mobile peer owner has in the $peer_i$. $TotalInter_i$ refers to the total number of interactions taken part by the $peer_i$ with the mobile peer owner during the past period $t_h$. $Fail_i$ is the total number of occasions when $peer_i$ fails to respond during the period.

Size has different meanings in different P2P environments. In a P2P file sharing network, the size of interaction expresses the file size shared in each interaction, while in a P2P business community, it shows the sums of money involved in each interaction. We also assume that table TV is sorted in a descending order according to the satisfaction (Sat) values. Thus, the *peer* with the highest satisfaction value shall be in the first row of the table. The mobile peer may decide an upper-limit for the trust value according to his/her preferences. Tables TV is controlled by the validity periods $t_h$, to maintain the freshness of the relevant data and to reduce memory and computational expenses. Trust evaluation model

### B. Trust evaluation model

Consider the situation where mobile peer X wants to interact with peer Y in order to accomplish a certain task. Peer X won't interact unless it is sure that peer Y is trustworthy. In order to find out whether peer Y is trustworthy, peer X calculates a trust value for peer Y. There are two ways in which to calculate trust value: direct and recommendation. When peer X has enough interaction experience with peer Y, peer X uses direct trust to calculate the trust value for peer Y. On the other hand, when peer X doesn't have enough interaction experience with peer Y, peer X uses recommendation trust to calculate the trust value for peer Y. In our paper, an interaction experience threshold is predefines based on the size of interactions in a P2P network. For example, the threshold is lower when downloading a 1M confidential file than a 200M confidential file. If peer X's interaction experience with peer Y exceeds this predefined threshold, peer X chooses direct trust to calculate the trust value for peer Y, otherwise, it chooses recommendation trust.

**Direct Trust Value:** Direct trust is denoted as $D(T_x(y), S)$. Where $T_x(y)$ is the direct trust value

that peer X calculates for peer Y. $S$ expresses peer Y's level of size of interaction which is granted by peer X. The level of size of interaction in a P2P business community is shown in Table 2.

TABLE II.
THE LEVEL OF SIZE OF INTERACTION

| Level | Bottom limit | Top limit |
|---|---|---|
| 1 | $m_0$ | $m_1$ |
| 2 | $m_1$ | $m_2$ |
| .... | .... | .... |
| n | $m_{n-1}$ | $m_n$ |

In Table 2, $m_0 \le m_1 \le m_2 \le \cdots \le m_n$, where $m$ denotes the size of interaction in a P2P business community, e.g., \$100, \$1000, \$10000, etc. The level of size of interaction satisfies the following rules.

(1) The lowest level is given to a new peer that doesn't have any interaction history.
(2) A certain level is updated if the number of successful interactions reaches the predefined number in the level. The predefined number is decided by the peer itself. The lower the current level is, the more the number of successful interactions it needs.
(3) The predefined successful interaction number in a certain level is increased if interactions fail due to malicious activities.

By introducing the notion of level, new peers are given the chances for interactions, which solves the trust problem when no prior interaction history exists, an issue that has not been addressed in many models. At the same time, it prevents new peers from cheating big time by having interaction chances. The direct trust value $T_x(y)$ is defined as:

$$T_x(y) = \alpha * \sum_{i=0}^{N(y)} \left(\frac{S(x,y) * M(x,y) * Z}{N(y)} + pen(i)\frac{1}{1+e^{-n}}\right) + \beta Risk(y) \quad (1)$$

where $\alpha$ and $\beta$ and are weighting factors that satisfies the condition $\alpha + \beta = 1$. $N(y)$ denotes the total number of interactions that peer X has performed with peer Y and $S(x,y)$ denotes the peer X's satisfaction degree of interaction in its ith interaction with peer Y which is in the range of (-1, 1). $M(x,y)$ is the ratio between the size of the ith interaction and the average size of interactions which reflects the importance of the ith interaction among all the interactions that peer X has performed with peer Y. Therefore, $M(x,y) = \frac{m_i}{m_v}$ where $m_i$ is the size of the ith interaction and $m_v$ is the average size of all interactions. We use $Z$ to denote the time factor.

$$Z = u(t_i, t_{now}) = \frac{1}{t_{now} - t_i}, Z \in (0,1) \quad (2)$$

where $t_i$ is the time when the ith interaction occurs and $t_{now}$ is the current time. $\frac{1}{1+e^{-n}}$ is the acceleration factor where $n$ denotes the number of failures. It can make trust value drop fast when an interaction fails. As this factor increases with $n$, it helps avoid heavy penalty simply because of a few unintentional cheats. Finally, $Risk(y)$ is used to express the risk factor. $pen(i)$ denotes the punishment function and

$$pen(i) = \begin{cases} 1, \text{ if the ith interaction fails} \\ 0, \text{ if the ith interaction succeeds} \end{cases} \quad (3)$$

**Recommendation trust value:** When two peers have litter interaction experience, other peers' recommendation is needed for trust establishment. Recommendation trust is the trust of a peer on another one based on direct trust and other peers' recommendation. Recommendation trust is calculated based on an enhanced polling protocol to be described below. Let we assume that peer Y requests an interaction with peer X and the size of the interaction is $Q$. First, peer X computes peer Y's direct trust denoted as $D(T_x(y), S)$.

*1)*

(1) If $Q \le S$ and $T_x(y)$ reaches a certain value (which is set by peer X), peer X considers peer Y to be trustworthy. It will then decide to interact with peer Y.
(2) If $Q \le S$ but $T_x(y)$ fails to reach a certain value, peer X chooses to join a group based on its interest. Then it checks its own group and location with GPS and floods a HELLO message, to announce itself to other peers by using Echo protocol [11], then requests all other members of the group to cast a vote for peer Y from the perspective of trust and distrust in the level of $Q$. For any new peer without any interaction history, its trust value would be 0 and would be granted the lowest level of the size of interaction. Without voting, it will be permitted to interact at the lowest level.
(3) If $Q \ge S$ but $T_x(y)$ fails to reach a certain value, peer X immediately refuses to interact with peer Y.
(4) If $Q \ge S$ and $T_x(y)$ reaches a very high value, peer X chooses to join a group based on its interest and then requests all other members of the group to cast a vote for peer Y from the perspective of trust and distrust at the level of $Q$.

Second, after the other peers receive the poll request message, they will decide whether to cast the vote based on the following formula. Let $E$ denotes a voting peer, then

$$DT_e(y) = \sum_{i=1}^{N(y)} \left(\frac{S(e,y) * M(e,y) * Z}{N_e(y)} + pen(i)\frac{1}{1+e^{-n}}\right) \quad (4)$$

where $DT_e(y)$ is the poll value of $e$ in Y. $N(y)$ denotes the total number of interactions $E$ has conducted with Y at level $Q$. $S(e,y)$ denotes peer $E$'s satisfaction degree in the ith interaction with y from the perspective of trust or the dissatisfaction degree from

the perspective of distrust. $M(e,y)$ is the ratio between the size of the ith interaction and the average size of interactions. $pen(i)$ is the punishment function.

Lastly, X calculates the recommendation trust which is given as:

$$RT_x(y) = (T - T') \tag{5}$$

$T$ is the vote result from the perspective of trust and $T'$ the vote result from the perspective of distrust. $T$ is given as

$$T = \frac{\sum_{i=1}^{N(w)} R(w) \times p}{N(w)} \tag{6}$$

where $N(w)$ denotes the total number of votes and $R(w)$ denotes peer $w$'s vote accuracy factor which is in the range of (0, 1). $p$ is related to $DT_w(y)$ such that if $DT_w(y) > 0$, $p = 1$, else $p = 0$.

## V. Experimental Study

Experiments have been carried out to study the effectiveness and the benefits of our proposed model. In a real environment, there may exist some vicious attacks including malicious recommendations or cheating in the accumulation of trust in small-size interactions. In addition, it should solve the trust problem when there is no interaction history or little trust value.

TABLE III.
Default Parameters in Simulation Experiments

| | |
|---|---|
| Number of Peers | 300 |
| Communicating range (m) | 70 |
| Simulation area ($m^2$) | $500 \times 500$ |
| Number of malicious Peers | 0% - 70% of all peers |
| Risk attitude | averse, neutral, seeking |
| Communication protocol | 802.11 |
| Life time (s) | [50,100] |
| Maximum speed (m/s) | 20 |

The simulation environment is set up as follows: we create 300 peers that will perform interacting in a mobile p2p resource sharing system. 300 mobile peers are uniformly distributed at the area whose size is $500m \times 500m$. Communicating range of a mobile device is *70m.* The simulated experiments were run on a dual-processor Dell server and the operation system installed on this machine is Linux with kernel 2.6.9. To make our simulation as close to the real mobile p2p systems where peers often go offline, we simulate the offline peers by assigning every peer a random lifetime (or Time-To-Live) within the step range [50, 100]. After reaching the lifetime, the peer will not respond to any service request, and won't be counted in the statistics either. After one more step, the peer comes alive again with a new life time randomly chosen from the range [50, 100]. In this analysis, we assume that all mobile peers have a same amount of battery power and participate in communication positively regardless of their roles. Each peer acts as both client and server to share its resources with other peers, and communicates with each other via IEEE 802.11. The default parameters in simulation experiments are showed in the table 3.

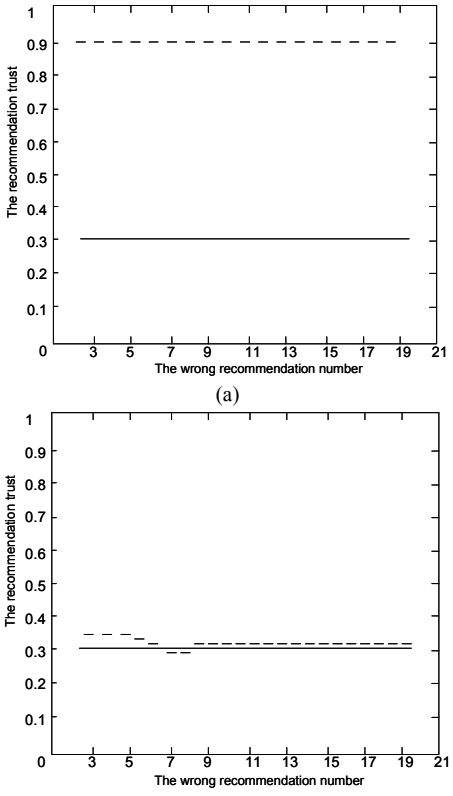



Figure 1. The relationship between the wrong recommendation number and the recommendation trust

### A. Malicious recommendation

In the first experiment we evaluate the trust evaluation model in terms of its efficiency of excluding malicious recommendations in the network. We implement and simulate a file sharing system. In this analysis, we assume that all mobile peers have a same amount of battery power and participate in communication positively regardless of their roles. So, we consider only a consistency evaluation factor. Figure 1 shows the simulation result in which the broken line denotes the recommendation trust value Tm that includes malicious peers' recommendations and the solid line denotes the real recommendation trust value Tr that doesn't include any malicious recommendations. In this simulation, a same malicious recommendation event occurs every 10 seconds.

As we can see Figure 1 (a), normal recommendation trust value is 0.3, but a malicious recommendation result is 0.9 by few malicious peer which broadcasts three times as high as a normal recommendation result. This indicates the vulnerability of a system without a trust evaluation scheme. Figure 1 (b) shows the process of filtering inconsistent data of a malicious node which acts

inconsistently after certain seconds with a proposed trust evaluation scheme. We can see that Tm fluctuates around Tr but the scale of the fluctuation is very small. The earlier the system detects a malicious node, the lower the malicious recommendations of it can affect the aggregated result.

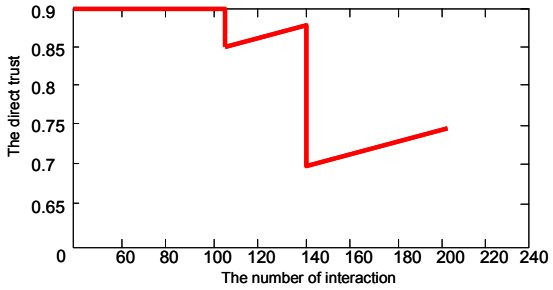



Figure 2. The relationship between the direct trust value and the number of interactions

### B. Cheating in the accumulation of trust

In our trust model direct trust has the property of rising slowly and dropping fast. The introduction of level of size of interaction limits interactions to certain levels. Consequently, attack is prevented when malicious peers cheat in large interactions through improving trust value using many small interactions. Our second experiment verifies the effectiveness of the proposed model by simulating the relationship between the direct trust and the interaction number of malicious peers. In the experiment, we firstly conducted 100 interactions. If there is no fraud, the direct trust value is 0.9 as shown in Figure 2. We can see when the peer has a behavior of fraud in the 105th interaction, its direct trust value would drop to 0.85. The peer has 35 small interactions from the 105th interaction to 140th interaction, but there is a difference between the current direct trust value and the previous 105th one. In the 140th interaction, the peer has another malicious behavior resulting in its direct trust value dropping faster. When it has one more cheating behavior, its trust value continues rapid dropping. All together, though the peer has successfully conducted 60 small-size interactions, the direct trust value could not recover the original trust value. The main reason is that after one malicious behavior, a peer needs to successfully conduct many more honest interactions to make up for the loss of trust value. If it conducts small-size interactions, it needs to have more interactions to offset the loss of its trust degree.

### C. Group Cheat

In the third experiment, we assess the performance of our mechanism under two attack models: independent cheat and group cheat. Our experiment also points out that the trust model is also sensitive to the group cheat. In the experiment, we add a number of malicious peers to the network such that malicious peers make up between 0% and 70% of all peers in the network. Figure 3 shows what is happening. In this figure, we compare the independent cheap and group cheat. Under independent cheat, the malicious peers firstly accumulate trust values through small interactions, gaining a relatively high trust. After trusted by most adjacent peers, the peer takes advantage of its high trust value to attack another peer, which means to always provide an inauthentic file to another peer when selected as download source.

Group cheat is that there is a group in which the peer of the group provides an authentic file to each other and provides an inauthentic file to the peer outside the group. The rate of inauthentic downloads under independent cheat or group cheat increases at the beginning, then starts to drop when the number of malicious peers reaches to 30%-40% of all peers in the network. The reason is that the trust computing mechanism used in our experiments punishes this behavior by lower the trust values quickly. Since malicious peers found by the mechanism will lose choice selected as download sources. As a result, the rate of inauthentic downloads will drop. However, due to the good rating coming from the cheating group, the rate of inauthentic downloads under group cheat drops more slowly than the one under independent peer. Yet one thing remains assured: the rate under group cheat is still dropping and will drop to 5%. Even if no malicious peers are present in the system, downloads are evaluated as inauthentic in 3%-5% of all cases – this accounts for mistakes users make when creating and sharing a file, e.g., by providing the wrong meta-data or creating and sharing an unreadable file.

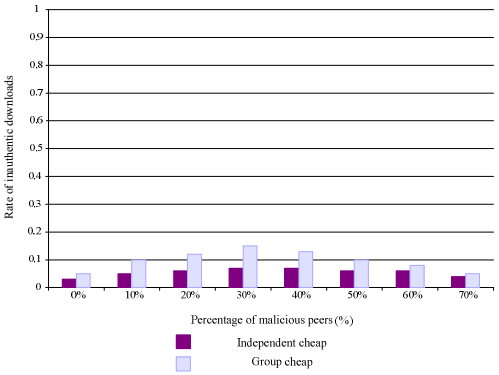



Figure 3. Simulation results of peers under independent cheat and group cheat

### D. The trust problem for a new peer

For peers without any interacting history, most previous trust models often define a default level of trust. The problem is that if this level is set too low, it may be difficult for the new peer to prove its trustworthiness through actions. If this level is set too high, there may be a need to limit the possibility that peers would have to "start over" by re-registration after misbehaving. In our trust, the introduction of the size of interaction effectively solves the trust problem of peers without any interacting history.

We assume that peer X is a new peer which has no interaction experience with other peers. If it wants to interact with peer Y, in our trust model, its trust value is 0 and it would be granted the lowest level of size of interaction. Without any voting, peer X would be immediately permitted to interact at the lowest level. Therefore our trust effectively solves the trust problem

for a new peer. As the new peer is granted the lowest level of size of interaction, it is prohibited from cheating in larger size interactions by giving interaction chances.

### *E. The influence of the time factor*

The experiment shows that our proposed model can truly reflect the influence of interaction history to trust which always decays with time, as shown in Figure 4. Many previous trust models failed to consider the decay influence, so the calculation of trust values is not very accurate. As the result, trust values are usually computed higher than the real ones in those trust models. Our proposed model can well solve this problem.

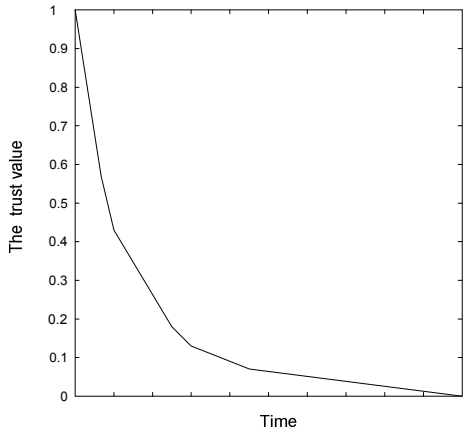



Figure 4. The relationship between time and trust value

## VI. CONCLUSION AND FUTURE WORK

The realization of trust mechanism in mobile p2p systems is quite different due to some characteristics of mobile environment, which indicates the trust between participants can not be set up simply on the traditional trust mechanism. In the paper we proposed a novel trust evaluation model for mobile P2P systems. The main factors that influence the trust in mobile P2P systems are identified. Our model does not employ cryptographic approaches or certification mechanisms, so it is light enough to fit well with mobile P2P systems without great overheads. To the best of our knowledge, our approach is one of the incipient researches on trust evaluation model for mobile P2P systems that can detect malicious and compromised mobile peers. In addition, the proposed model effectively solves the trust problem of peers without any interacting history. We expect that our trust evaluation model can help to make resilient mobile P2P systems. In the near future, we would like to test our trust into more real mobile p2p systems and analyze the system performances.

## ACKNOWLEDGMENT

The work in this paper has been supported by Scientific Research Program Funded by Natural Science Basis Research Plan in Shaanxi Province of China (Program No.2011JQ8006) and Shanxi Provincial Education Department (Program No.11JK1060).

## REFERENCES

[1] K. Takeshita, M. Sasabe, H. Nakano, "Mobile P2P Networks for Highly Dynamic Environments", in *Proc. of the 6th IEEE International Conference on Pervasive Computing and Communications*, Hong Kong, 2008, pp. 453-457.

[2] S.D. Kamvar, M.T. Schlosser and H.G. Molina, "The EigenTrust Algorithm for Reputation Management in P2P Networks", in *Proc. 12th International Conference on Word Wide Web,* Budapest, Bulgaria, May 2003, pp. 640-651.

[3] R. Zhou and K. Hwang, "PowerTrust: A Robust and Scalable Reputation System for Trusted P2P Computing," *IEEE Transactions on Parallel and Distributed Systems,* Vol.18, No.5, May 2007.

[4] Y. Wang and J. Vassileva, "Trust and Reputation Model in Peer-to-Peer Networks," in *Proc. 3th International Conference on Peer-to-Peer Computing*, Sweden, September 2003, pp. 150-157.

[5] C.Q. Tian, S.H. Zou, eta, "A New Trust Model Based on Recommendation Evidence for P2P Networks", *Chinese Journal of Computers,* 2008, Vol.31, No2:271-281.

[6] Thomas Repantis and Vana Kalogeraki, "Decentralized trust management for ad-hoc peer-to-peer networks," in *Proc. of the 4th international workshop on Middleware for Pervasive and Ad-Hoc Computing,* Melbourne, Australia, 2006.

[7] L. Zhou, Z.J. Haas, "Securing ad hoc networks", *IEEE Special Issue on Network Security,* vol.13.No.6, 1999, pp. 24–30.

[8] J. Kong, P. Zerfos, H. Luo, S. Lu, L. Zhang, "Providing robust and ubiquitous security support for mobile ad-hoc networks", in *Proc. 9th International Conference on Network Protocol*, November 2001, pp. 25-260.

[9] O. Wolfson, B. Xu, H. Yin, and H. Cao, "Search-and-Discover in Mobile P2P Network Databases", in *Proc. of the 26th IEEE International Conference on Distributed Computing Systems,* July 2006.

[10] N. Li, J. Mitchell, and W. Winsborough., "Design of a role-based trust management framework", in *Proc. of the IEEE Symposium on Security and Privacy*, Oakland, 2002, pp. 114-130.

[11] N. Sastry, U. Shankar, D. Wagner, "Secure verification of Locotion Cloims", in *Proc. of the 2nd ACM worhhop on Wireless security*, New York, 2003, pp. 1-10.

Xu Wu received his M.Sc. degree in Computer Science, and Ph.D. degree in Computer Science, from the Beijing University of Technology, in 2005, 2010, respectively. Her research interests include trusted computing, pervasive computing, mobile computing, and software engineering.

She has published more than 30 technical papers and books/chapters in the above areas. She is currently a lecturer of Xi'an Universtiy of Posts and Telecommunications. Her research is supported by Scientific Research Program Funded by Natural Science Basis Research Plan in Shaanxi Province of China and Shanxi Provincial Education Department.

# The Design and Implementation of Single Sign-on Based on Hybrid Architecture

Zhigang Liang
College of Computer Science &Engineering
South China University of Technology
Guangzhou, China
Email: bumengxin@126.com

Yuhai Chen
Asset Management IT Department
HSBC Software Development (Guangdong) Limited
Guangzhou, China
Email: yuhai_chan@163.com

***Abstract*—For the purpose of solving the problems of user repeated logon from various kinds of Application which based on hybrid architecture and in different domains, single sign-on architecture is proposed. On the basis of analyzing the advantages and disadvantages of existing single sign-on models, combined with the key technology like Web Service, Applet and reverse proxy, two core problems such as single sign-on architecture mix B/S and C/S structure applications and cross-domain single sign-on are resolved. Meanwhile, the security and performance of this architecture are well protected since the reverse proxy and related encryption technology are adopted. The results show that this architecture is high performance and it is widely applicable, and it will be applied to practical application soon.**

***Index Terms*—single sign-on, web service, cross domain, reverse proxy, B/S, C/S**

## I. INTRODUCTION

With the information society, people enjoy the progress in the huge interests, but at the same time also faced the test of information security. With all system users need to log in the system increased, users need to set a lot of user names and passwords, which are confused easily, so it will increase the possibility of error. But most users use the same user name and password, this makes the authentication information is illegally intercepted and destroyed the possibility of increased, and security will be reduced accordingly. For managers, the more systems need more corresponding user databases and database privileges, these will increase management complexity. Single sign-on system is proposed a solution to solve the problem. Using single sign-on, we can establish a unified identity authentication system and a unified rights management system. It not only improve system efficiency and safety, but also can use user-friendly and to reduce the burden on administrators.

TABLE 1 The comparison of a variety of single sign-on to achieve models

| SSO Achieve-Model | Action ability | Manageability |
|---|---|---|
| Broker Model | The large transformation of the old system | Enable centralized management |
| Agent Model | Need to add a new agent for each of the old system, transplantation is relatively simple | Management more difficult to control |
| Agent and Broker Model | Transplantation simple, transformation of the old system with limited capacity | Enable centralized management |
| Gateway Model | Need to use a dedicated gateway to access various applications | Easy to manage, but databases between the different gateways need to be synchronized |
| Token Model | Implementation of relatively simple | Need to add new components and increase the management burden |

Single sign-on refers to when the user needs to access a distributed environment which has different applications to provide the service, only sign on once in the environment,

[1] Manuscript received December 29, 2010; revised October 26, 2010; accepted October 18, 2010.
[2] corresponding author: Zhigang Liang, South China University of Technology; Yuhai Chen HSBC Software Development


doi:10.4304/jnw.7.1.165-172

no need for the user to re-sign on the various application systems[1]. Now there are many products and solutions to implement SSO, such as Passport of Microsoft, IBM Web Sphere Portal Server although these SSO products could do well in the function of single sign-on, but most of them are complex and inflexible. Currently, the typical models to achieve SSO include broker model, agent model, agent and broker model, gateway model and token model [2]. In table 1, it analyses these models can be implemented and manageability. Based on the above comparison, agent and broker model has the advantages both centralized management and revised less original application service procedure. So I decide to adopt agent and broker model as the basis for this model. In order to integrate information and applications well and with the B/S mode in-depth application software, there has been the concept of enterprise portal, offer a best way to solve this problem. Enterprise portal provides business users access information and applications, and complete or assist in a variety of interactive behavior of a single integrated access point. The appropriate system software portal provides a development, deployment and management of portal applications services. Enterprise information portal concerns portal, content management, data integration, single sign-on, and much other content.

## II. System Construction which Registers based on the Web Service mix Construction Single Sign-on

The system consists of multiple trust domains. Each trust domain has much B/S architecture of the application servers; in addition to B/S architecture of the application servers also included C/S architecture application servers. All the applications are bound together through a unified portal to achieve functionality of single sign-on. You can see that this architecture is based on the agent and the broker model. A unified agent portal is playing a broker role, and various applications are playing an agent role. The B/S architecture applications are installed on the Client side of SSO Agent, and the unified portal is installed on the Server side of SSO Agent. Between them is through these two Agents to interact. In addition, in Fig 1, the external provision of authentication server is LDAP authentication interface. Token authentication Web Service server provides the interfaces of single sign-on token of the additions, deletions, editions and queries. But the permission Web Service server provides the appropriate authority information system, to achieve unified management authority for accessing unified portal application system.

The system supports cross-domain access, that is, the domain D1 users can access the application domain D2, and the domain D2 users can access the application domain D1. At the same time, the system also supports the application of different structures between the single sign-on, that is, user after accessing the application A of the B/S structure access the application E of C/S structure without having to repeatedly enter user name and password, or user access the application A after the application E without re-enter login information.

The whole structure of Single Sign-on is as Fig 1 shown.

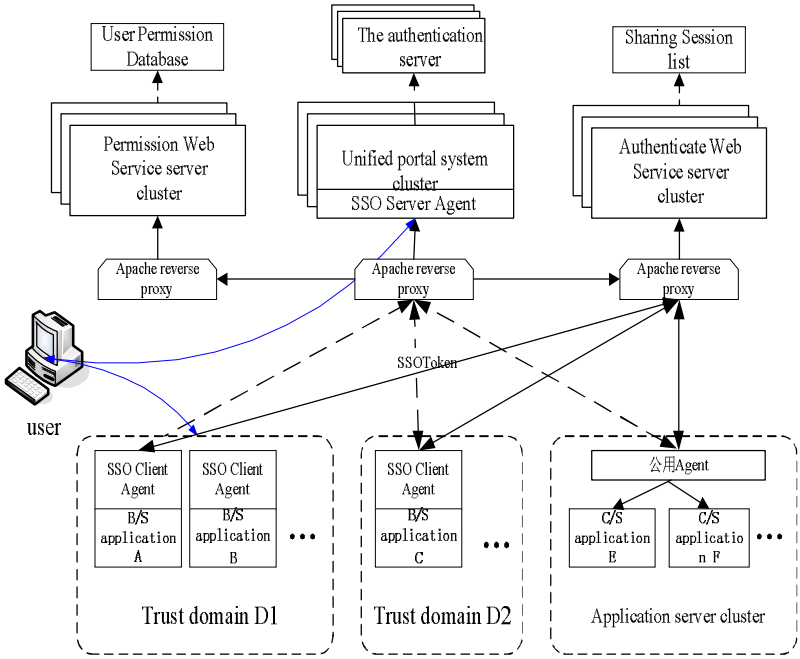



Figure 1: The Structure of Single Sign-on

### A. *The login process*

The whole single sign-on process is as Fig 2 shown:

Below is the process specific steps description:

*1)* User login in the client browser to access A application, SSO Client of A system intercept and redirect the URL to the landing page of Unified Portal System

*2)* Enter the user name and password, Unified Portal System submits to the authentication server for authentication. If the information is correct, Unified Portal System automatically generates, saves notes and the role of the user ID to a local, and calls the increate-note interface of Web Service to insert the information.

*3)* Unified Portal System returns a list of application resources pages to the user. The user clicks any one application system (e.g. A system). The SSO Client-side of A application system read the notes information and call the query-notes interface of Web Service. If it is consistent and within the time limit, it will get the role information of the user in A application system and log in A application system. At the same time, it will call the update-note interface of Note Certification Web Service to update the log-in time of this current note. Then call the interface of user rights Web Service to get this user's permission information with corresponding application system.

*4)* If user end to access A application system, exit and click on the link of B application system, system implementations will be are as the same as steps (3).

*5)* If user complete all the required access-applications and need to do the log-off operation, it will mainly call the deletion-note interface to destroy the corresponding note information.

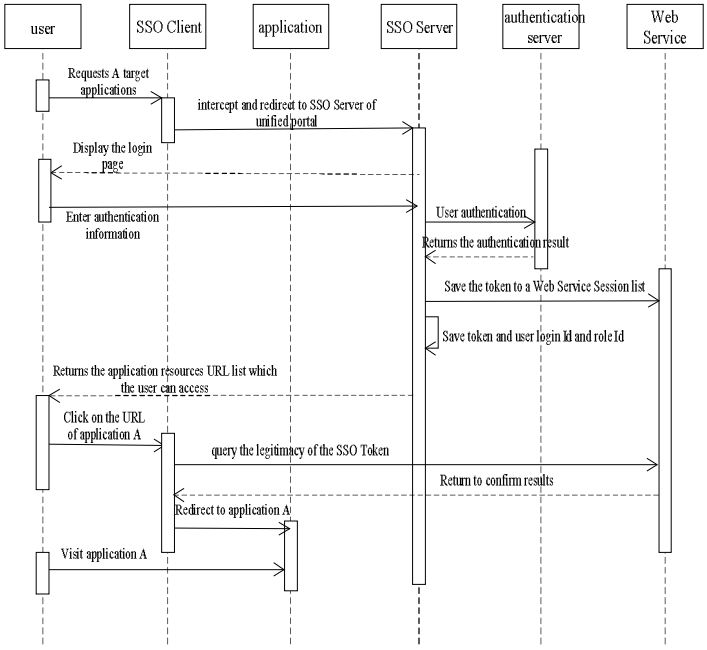



Figure 2: The whole process of Single Sign-on

### B. *The solution of Cross-domain problems*

In the traditional implementation of single sign-on system will be generally used cookie as storage of client-side notes, but because of restrictions on cookie itself properties make it only on the host under the same domain effective, and distributed application system always can not guarantee that all hosts under the same domain. The current system does not store the note information in the client-side but placed various application parameters of the link directly. The note-verification is through the application of the SSO Client-side call to the corresponding interface of Web Service to complete.

Through the Simple Object Access Protocol (SOAP) to provide software service in the Web, use WSDL file to illuminate and register by UDDI [3]. Shown in Fig 3, after the user through the application of UDDI to find a WSDL description of the document, he can call the application which through SOAP to provide by one or more operations of Web services. The biggest characteristic of Web Service is its cross-platform, whether it is the application of B/S structure or C/S structure, whether it is the application using J2EE or .NET to implement, it can access Web Service as long as to give Web Service server's I:P and interface name. The following is this system process of achieving cross-domain access:

*1)* User log in Unified Portal system successfully.

*2)* User accesses A application system within the trusted domain D1, complete the access and then exit this application.

*3)* User clicks the URL of B application system within trusted domain D2 of the resources list of Unified Portal.

*4)* SSO Client of B application intercepts the request, gets the note behind URL, and calls the query-note interface of Web Service.

*5)* Query interface of Web Service gets back the legal information of this note to the SSO Client.

*6)* SSO Client redirect to B application system, the user access B application.

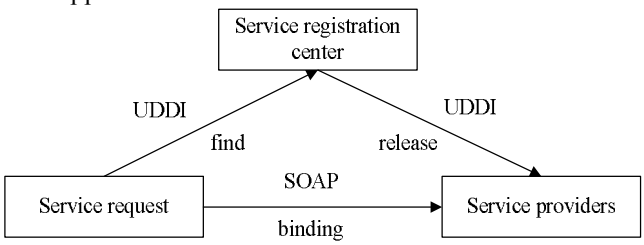



Figure 3: Web Service Structure

### C. *The Solution of Single Sign-on between B/S and C/S Structures*

As we know, the implementation principles of applications are quite different between B/S and C/S structures. In this system, the applications of B/S structure can be accessed through by clicking URL of the application-resources-list page of Unified Portal. Since the browser security restrictions, the page does not allow users to directly call the local exe files, so need to adopt an indirect way to call C / S architecture applications. This article uses the way of Applet to call local exe files, the implementations as below:

For all C/S structures, create a common Agent. This Agent's role is an interceptor, which means it need browsers to access after the C/S structure joined up Unified Portal system. (Please note that: Since the original B/S architecture and C/S structure is not using the same authentication method. For the C/S application access to the unified portal framework to achieve single sign-on system, the need for a unified authentication management, and in order to change the amount of compression to a minimum. Implementation of this system is to create a needless user name and password authentication code for all applications which are accessed a unified portal, and land on the unified portal system certified landing page. When a user uses browser to log into the unified portal system successfully and then can access any application, including the B/S architecture and C/S structure of the application. To be ensure the security of C/S application framework, when the user clicks directly to the desktop shortcut to open applications still using the original authentication.)

Applications of C/S architecture are all using the same Applet of URL. The received parameters of this common Applet include bills, application name, unified login-name and password. When a user does not do the login operation before, the first visit a C/S application will be intercepted to the login-page of Unified Portal system for sign-on. If a user logged in before, when visiting a C/S application, this Agent will call the interface of Web Service note-validation to validate the note which was transferred. If the validation is successful, Applet object will be downloaded to the user's local to implement. In order to transform the original applications as little as possible, the method of this article is to open the login window of the corresponding application through by Applet. Below are the codes:

```
public void OpenExe(String appName){
  Runtime rn=Runtime.getRuntime();
  Process p=null;
  p=rn.exec("c:\." + appName + ".exe");
}
```

After opening the log-in window of the application, the operation steps of this Applet as follows:

*1)* Applet needs to call the bottom API of windows to get the user-name of login window, password-input box and the handle of login button through by JNI.

*2)* Locate the user-name-input box to send unified login name. Locate password-input box to send the password. (Password information is arbitrary and in order to distinguish it from the user clicks on a shortcut directly landing system, also need to send a code that uses a unified portal access without a password authentication system.) Locate the login button to send the click event.

*3)* At last, Applet will minimize the IE window, the related windows of applications will be placed to the forefront.

These are the implementation process of C/S architecture application single sign-on. The application codes which have not been changed at all before will join up the Unified Portal system using a loosely coupled way. Need to explain that, due to the Applet JVM security restrictions, cause Applet can not directly call the user's System32 directory of local native windows dll. Now the method is first to start to use C or C + + to write the class which got the corresponding input box and button of the login window, and generate a JNIWindowUtil.dll file (JNIWindowUtil is a user-defined dll's name). And it is to place the dll in the same directory with the Applet. When the Applet is downloaded to the client side, dll is also downloaded to the user's System32 directory of local at the same time. Applet process also needs to execute statement: System.loadLibrary("JNIWindowUtil"). After completing these above steps, it can really use JNI in Applet internal to achieve the corresponding functions.

### *D. Authentication server*

The old system user authentication information is usually stored in a database, but this architecture used LDAP to store user information. LDAP, short for Lightweight Directory Access Protocol, is the standard directory access protocol based on a simplified form. It also defines the way data organization; it is based on TCP/IP protocol of the de facto standard directory service, and has distributed information access and data manipulation functions. LDAP uses distributed directory information tree structure. It can organize and manage various users' information effectively and provide safe and efficient directory access. Compared with the database, LDAP is the application for reading operation more than writing operation, and database is known to support a large number of writing operations. LDAP supports a relatively simple transaction, but the database is designed to handle a large number of various transactions. When the query in Cross-domain data is mainly read data, modify the frequency is very low. When Cross-domain access to the transaction, it does not require a large load, so in comparison with the database, LDAP is the ideal choice. It is more effective and simple. This framework is applied to a large bank, the bank's systems can belong to different regions, and use of personnel may come from different geographies. In order to achieve distributed management, the use of three-level management, respectively named the Bank headquarter, Provincial and City branches of the three levels of branches, as shown in Fig 4:

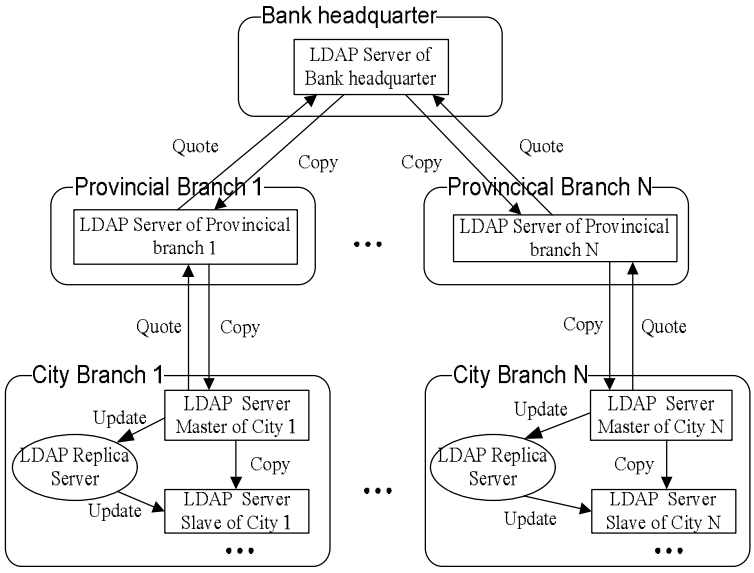



Figure 4: LDAP Authentication Structure

Directory replication and directory reference is the most important technology in LDAP protocol. It can be seen from the figure, Provincial and City branches of the LDAP server branch data are copied from the floor, but not a simple copy of all information, just copy the relevant data with their own information. Because for a particular application system, its users are mostly belong to the same

region, so that implementation can greatly simplify the management of directory services and to improve the efficiency of information retrieval. When a user outside the region to use this system, because of its user information in the region can not retrieve LDAP server, you need to other regions of the LDAP server to query, and therefore requires a way to use up the reference queries, first Provincial branches of the server search, without further reference to Bank headquarter of the server up until the search to the appropriate user information.

The management of the regional City branch, using the LDAP directory replication model of Single Master/Multi Slave. When a directory user queries the directory information, Master LDAP Server and Slave LDAP Server (Slave server can have more than one) can provide services to the directory, depending on the directory user makes a request to which the directory server. When the user requests the directory update directory information, in order to ensure the Master LDAP Server and Slave LDAP Server in the same directory information content, the need for replication of directory information, this is achieved through the LDAP Replica server data synchronization. Using directory replication, when the directory number of users increases or the need to improve system performance, only simply add Slave LDAP server to the system and then can immediately effective in improving system performance, and the whole directory service system can have a good load balancing.

### *E. Permissions Web Server*

Access Control technology began in the computer age of providing shared data. Previously, the way people use computers is mainly to submit the run-code written by user or run the user profile data. Users do not have much data sharing, and do not exist to control access to data. When computer comes into user's shared data, the subject of access control is nature to put on the desktop. Currently, the widely used access control models is using or reference to the early nineties of last century the rise of role-based access control model (Role-Based Access Control - RBAC). RBAC model's success is that it is inserted the "role" concept between the subject and object, decouples effectively between subject and the corresponding object (permission), and well adapts to the subject and object associated with the instability. RBAC model includes four basic elements, namely the user (User - U), roles (Roles - R), session (Session - S) and permission (Permission - P), also in the derived model also includes constraints (Constrains - C). The basic idea is to assign access rights to roles, and then the roles are assigned to users. In one session, users can gain the access rights through roles. The relationship between the elements: a user can have multiple roles, a role can be granted to multiple users; a role can have multiple permissions, a permission can be granted multiple roles; user can have multiple conversations, but a conversation is only to bind a user; a conversation can have multiple roles, a role can share to multiple conversations at the same time; Constraints are that act on specific constraints on these relationships. As shown in Fig 5:

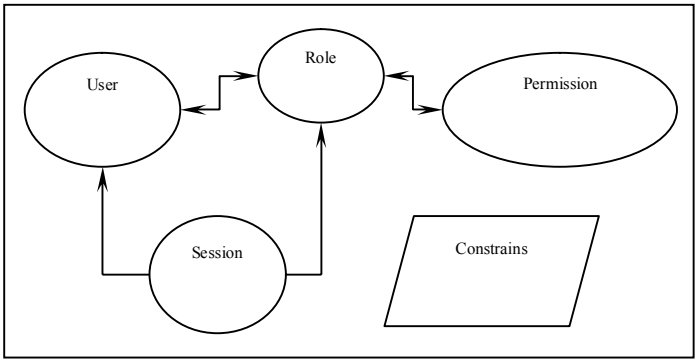



Figure 5: RBAC Basic Model

This system is to use this very sophisticated permission access control model.

Rights management, not only protects the safety of system, but also facilitates management. Currently most using the manner of code reuse and database structure reuse, rights management module is integrated into business systems. Such a framework has the following shortcomings.

*1)* Once the permissions system has been modified, the maintenance costs will be very high. This is the general shortcoming of using code reuse and database structure reuse. Once revised, we will have to update the code in all business system and database structure, and also to ensure that existing data can smooth the transition. Some processes may require manual intervention, which is a "painful" thing for the developers and maintenance personnel.

*2)* Did not facilitate management of Permission data. Need to enter permission management module of various business systems to manage the corresponding rights. It is complex operation, and not intuitive.

*3)* For different architectures, different software operating environment, we must develop and maintain different permissions system. For example, B/S and C/S architecture system must each develop their own rights management system.

This paper argues that most common function of the permission system can abstracted from business systems to form an independent system - "unified rights system". Business system only retains the rights inquiries, read common data system and the control rights function of this system specific fine degree (such as menus, buttons, links and so on). As shown Fig 1. How to achieve a unified rights management? This paper argues that there are two implementations, one way is to use Web services to provide rights data; the other is using Mobile Agent to provided permissions data. However, the second one run, maintenance costs are higher, and implement is more difficulty than Web services. So this architecture using Web services to provide authority data of the various systems in

a unified way. Business system using Web services client interface to query data and obtain system privileges to share data. The client is just a port, and specific implementation code is placed in "unified rights system". These client interfaces introduced to the business system by package. If we keep the client interfaces unchanged, modify and upgrade of the unified authority system will not affect the business system. Users and permissions through Web pages of "unified rights system" to unify management and to achieve the user's single sign-on. The biggest advantage of Web services is the integration of data between heterogeneous systems. This breaks the restrictions of B/S, C/S structure; there is no difference between Windows and Linux platform.

## III. System Security Analysis

*1) The interception of user name and password.* The system for authentication of the user login and send the user name and password to Applet objects are used SSL protocol. And make sure that information during transmission confidentiality and integrity. Meanwhile, due to the key which is hard to get and time limited, so it can effectively prevent that intermediary attack to the transmission of information.

*2) Replay attack.* Many systems will use the ways of time stamp to avoid duplication attacks. However, this approach requires the computer clocks of communication parties to be synchronization. But it is difficult to achieve, while also appears the following situation: the two sides' clocks which are connecting with each other, if they are out of synchronization occasionally, the correct information may be mistaken to discard for replay information, but the incorrect replay information may be as the latest one to receive. Base on the above, this system needs a simple method F of an appointment between query interfaces of Web Service provided and SSO Client of each application system or Agent. This system's parameter value is a random string X. The whole process of bill validation as shown in Fig 6:

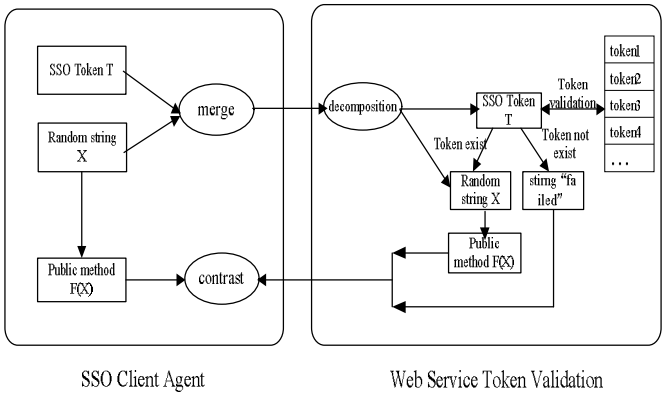



Figure 6: The process of Based on Web Service bill validation

*a)* When the user accesses to application system A, the SSO Client of system A intercept and call the query interface of Web Service provided, and the input parameters are a random string X and the corresponding note.

*b)* Web Service server receives system A's call, intercepts note to compare with the note's information of Session queue. If the queue contains the note, it will return the value of F(X) for showing validation is successful. If not, it will return 'failed' for showing validation is failed.

*c)* SSO Client of the application A receives the return information of Web Service server, and then compares the return value with F(X) of this system. If the two are the same, it will redirect to system A, otherwise it will not be allowed to visit.

The random string is different, which each interact with Web Service server. So you can limit replay attacks very well.

*3) Use reverse proxy technology.* Reverse proxy technology is a substitute, which is a reverse proxy server as to N identical application servers. When external access to this application, it just knows the reverse proxy server and can not see the back multiple application servers. This improves the security of this application system.

Through the above analysis, this system can provide users with a good safety Web environment.

## IV. System Performance Analyzes

First, this system in addition to use SSL encryption in the transmission of user name and password, the interactions of between other servers and between user and servers are based on HTTP protocol to transmit. SSL encryption and decryption process requires a lot of system cost, severely reduces the performance of the machine, so we should not be use this protocol to transmit data too much. Since the data which need to encrypt is small, only a user ID value (note), so the performance of using MD5 to encrypt is quite satisfactory.

Second, when user accesses any application system of each domain, they will be redirected to Unified Portal system for identity authentication, or directed to Web Service server for note validation. User need to sign on the system only when he is certification first time. When the visitor volume is larger, the user switch to the new application system will easily handle an interruption, which is single sign-failure phenomenon [7]. This phenomenon has two reasons, one is the server load is too large, the other one is network bandwidth is not enough [8]. Among them, the method which is resolved the server load is too large is to use server cluster. Cluster is made up of multiple servers. As a unified resource, it provides a single system service to external. In this system, except for using reverse proxy technology to improve the security of accessing the applications, the more important is capability which can

help to implement cluster technology of load balancing. The whole structure of reverse proxy is shown in Fig 7:

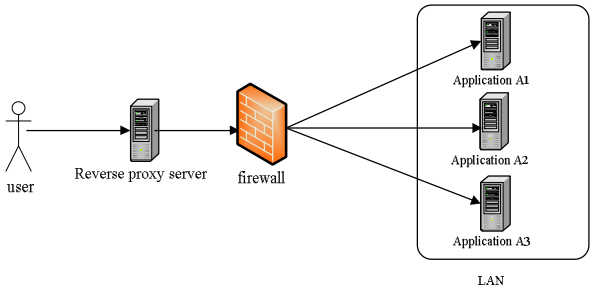



Figure 7: Reverse Proxy Structure

Fig 7, reverse proxy server R provides the corresponding interface to implement the algorithm of load balancing except for providing cache for the behind A1, A2 and A3 application. That is, it can consider the arrival request to distribute to the server which has the best performance through by scanning the conditions of CPU, memory and I/O of A1, A2, A3 server. By LoadRunner8.1, the use of reverse proxy system before and after was related to stress testing. The test results are shown in Fig 8:

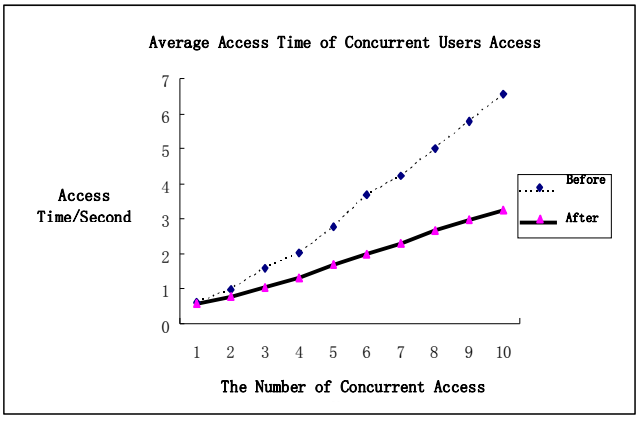



Figure 8: The system stress testing result

It can be seen from Figure 8, at the beginning, when the number of concurrent users is not large, use the reverse proxy and out of use proxy is similar. But with the gradual increase of concurrent users, the performance difference between the two is more and more evident. To 100 concurrent users to access, the system response time of using the reverse proxy is almost twice as fast as the one out of use proxy.

System Web Service server needs to store the information of note, so using Web Service server cluster to pay attention to this problem: the different Servers of cluster use different JVM, so an object of JVM can not be accessed by other JVM directly. For this problem, there are two methods to resolve:

*1)* Put the object in Session, and then configure cluster to the copy model of Session.

*2)* Use Memcache, put the object in Memcache, and then all Server get this object from Memcache. To be equivalent to open a public memory area, which everyone can access.

Any more, business system requires get rights information data through the Web services frequently. This performance of the system put forward higher requirements. The system has been taken two measures to improve performance:

*1)* It receives a request by using time-sharing patterns of authority data server. After that, if always be calculated in real-time data, it will not certainly respond in time as the server limited resources. This will cause the system to slow down. A "time-sharing patterns of authority data" can solve this problem. When the system data changes (such as a new operation is authorized to the role, etc), the system automatically calculates the affected user, and then re-calculate the relevant authority data, save to the specified field of database. When the business system requests data, only run "to read the database designated field corresponding to the specified data" such a simple action, you can greatly speed up the system response speed.

*2)* Designed the cache structure to rely solely on time-sharing model is not enough to improve response time. After all, access to the database is relatively resource intensive. This paper designes cache struture which is based on the memory to further improve the speed, to achieve the following functions: when the system starts, it will reside the public data which is often uesd to cache (such as organizational structure, role information, etc.), if the change in the operation of this part of the data, then also update the data in the cache. It is to adjust the priority of the data according to the frequency of access the cache object, and regularly to remove expired objects. It can improve the cache hit rate through the trade-off algorithm of cache object which is optimized.

## V. CONCLUSION

This text develops a single sign-on. Compare with the traditional sign-on model, this text presents a new solution which is single sign-on of hybrid framework of being based on Web Service. This program can be applied to distributed environment of being based on multi-trust-domain, and also applied to the case which has both B/S structure and C/S structure applications. It optimizes the unified management of the administrator faces to user information, facilitate the user to access to each application resource, improve the efficiency of user use application resource, enhance the security of user access each site, and provide the basis for integrating more sub-application systems.

## VI. ACKNOWLEDGMENT

This research project has been supported by Pansky Science and Technology Co., LTD

## REFERENCES


[1] Paker T A .Single Sign-on systems—the technologies and the products [J]. European convention Security and Detection,1995.

[2] ZHANG Ting,GENG Jixiu. Research and Design ofW eb-based SSO System[J] Computer Simulation 2005,22(8): 128-131

[3] IBM,Microsoft.Security in a Web services world: A proposed architecture and roadmap [EB/OL]. http://www.ibm.com /developerworks/library/ws-secmap/, 2002 -04-07.

[4] Liu Runda,Zhu Yunqiang,,Song Jia,,Feng Min. Implementation of a simple cross domain single sign on system [J] computerapplications 2007, 27(2): 288-291.

[5] DENG Yun,CHENG Xiao-hui System design of mobile cross-domain single sign-on[J] Computer Engineering and Design 2010, 31(8): 1667-1672..

[6] LI Xin,ZHANG Jun Design and realization of XKMS model towards web service[J] Computer Engineering and Design 2010, 31 (8) : 1738-1742

[7] JIN Wei-zu,LI Ping-xin Solution Schema for Single Location Invalidation Based on CAS Cluster Computer Engineering 2010,36 (1):51-54

[8] JIN WANG Qi Websites Single Sign-on Based on Reverse Proxy[J] Computer Engineering 2008, 34(14): 138-140.

[9] D.F.Ferraiolo, R Sandhu, S Gavrila, et al. Proposed NIST standard for role-based access control[J]. ACM Transaction on Information and System Security, 2001, 4(3): 224-274.

[10] E.Bertino, P.A.Bonatti. TRBAC: a temporal role-based access control model[J]. ACM Transaction on Information and System Security, 2001, 4(3): 191-223.


Zhigang Liang, 1985-08, was born in Maoming City, Guangdong Province. Currently a graduate student at Computer Science and Engineering of South China University of Technology, major field of study is computer system structure and computer application.

He ever had an internship in HSBC Software Development (Guangdong) Limited and Pansky Science and Technology Co., LTD, job title is Java software engineer.

Yuhai Chen, was born in Maoming City, Guangdong Province on 30th October 1984. He graduated from Guangdong Polytechnic Normal University in 2008, major in Software engineering, has a Bachelor of Engineering. Major Research Interests include Software Development and Technology, Systems Analysis and Database System.

He ever worked at Complete Solution (Guangdong) Limited in Guangzhou City from Jun. 2008 to Nov. 2009, job title is JAVA SOFTWARE ENGINEER, and primary responsibility are System Development and Java programmer. From Dec. 2009, he joins HSBC Software Development (Guangdong) Limited, job title is SOFTWARE ENGINEER, and primary responsibility are System Development and AS400 programmer. He is familiar with AS400 (RPG, CL, SQL), C++, Java, Struts, Spring, Hibernate, HTML, JavaScript, CSS, XML, MySQL and Oracle Database.

# An Access Control Model based on Multi-factors Trust

Shunan Ma[1]
[1]College of Computer Science and Technology, Beijing University of Technology, Beijing, China
Email: mashunan@emails.bjut.edu.cn

Jingsha He[2] and Feng Gao[1]
[2]School of Software Engineering, Beijing University of Technology, Beijing, China
Email: jhe@bjut.edu.cn, maple0371@emails.bjut.edu.cn

*Abstract*—**Conventional access control models are suitable for centralized and static environment, but they seldom meet the requirements of open and dynamic environments. The design of effective access control models to meet the challenge is a current research focus. As an important factor in security, trust is increasingly applied to security management, especially in access control. In this paper, by considering multi-factors feature of trust, we proposed an access control model based on multi-factors trust. The model includes multi-factors trust computation, permission mapping and feedback module. Simulation results show that the model is suitable for access control in dynamic environments.**

***Index Terms*—trust; access control; permission**

## I. INTRODUCTION

Access control, which is used to restrict the use of resources, is an important safeguard in network security. Nowadays, most of access control models have been studied extensively in centralized and static environment, and they seldom meet the requirements of some open and dynamic environments, such as Grid and P2P. Traditional access control models do not work in these environments. The entities that a system well will interact with or the resources that will be accessed are not always known in advance. Thus, it is almost impossible to predefine permissions to an entity. Since almost all traditional access control models rely on successful authentication of predefined users, they become unsuitable for open and dynamic environments. Access control in these environments must dynamically adapt to dynamic addition and deletion of entities.

Trust is a part of our daily life and thus can be used as a tool to reduce the complexity of making access decisions, which can be accomplished by using trust to provide security [1]. In recent years, many researchers have applied trust to the dynamic environments. In [2], trust models are proposed to control anonymity, unpredictability and uncertainty. However, there is several factors affect trust. For example, entities' attribute, time, network condition and history behavior.

In this paper, we propose an access control model based on multi-factors trust. We compute trust using several factors, and we present a trust based access control model.

The rest of the paper is organized as follows. In Section 2, we review some related works. In Section 3, we compute trust using several factors. In Section 4, we describe an access control model based on multi-factors trust. In Section 5, we present and analyze some simulation results. Finally, in Section 6, we conclude this paper with a discussion on our future work.

## II. RELATED WORK

Access control is the mechanism that allows owners of resources to define manage and enforce access conditions applicable to each resource [3]. Most access control models were developed for special environment. There are mainly three kinds of access control methods: Discretionary Access Control (DAC)[4], Mandatory Access Control (MAC)[1][5]and Role-based Access Control (RBAC)[6][7][8]. However, none of them is suitable for the dynamic and open environments due to the lack of flexibility and efficiency.

Semantic access control (SAC) [9] is a new kind of access control model, which uses machine reasoning at a semantic level to determine whether let the requests pass according to the semantic descriptions of the policies, requests, resources and other entities. Compared with traditional access control, SAC is more scalable, more applicable to dynamic environments with heterogeneous and complex access criteria. But the foundation of SAC is the semantic web technologies. Thus it can not be applied in all access control fields.

A management of access control for distributed environment which supports dynamic collaboration is discussed in [10]. With the development of ontology and semantic web, a new access control, context-based access control has been designed. Different from the traditional access control, context-based access control looks context as the key of access control [11]. It uses the context of requestor to decide whether to give the requestor the warranty or not.


doi:10.4304/jnw.7.1.173-178

With the research of trust, some researchers have thought about the application of trust. Researchers in [12] have proposed a way to incorporate the concept of trust to RBAC to address this particular problem. The concept of access control based on a trust level was discussed in [13], who described a role-based access control model that assigns roles to users based on their trust level. TCAC [14] proposed an extended RBAC model based on trust and context. In TCAC, only the user's trust value is not less than the trust threshold in access control policy and the context information satisfy the limit, the user will be given some roles. Some researchers proposed trust-based access control [15][16]. Trust-based access control is to use trust to the requestor as the key of access control. The owner of resource gives the warranty to the requestors who he trusts. In these work proposed above, although trust was introduced into access control models, but they didn't take into account factors which affect trust.

## III. MULTI-FACTORS TRUST COMPUTATION

### A *Multi-factors of Trust*

Trust is very subjective that reflects one body's subjective expectation on another body's future actions based on their previous exchanges [17]. In general, trust changes dynamically along with the behavior of users. Every user has a particular trust value towards others at a certain point of time or during a certain period of time. The trust value changes as a result of interactions with others.

When computing trust value of an entity, we will take into account its attribute, history behavior, time and network condition.

IP attribute can affect trust value of an entity. We use $T_1$ to denote the influence of IP attribute for trust computation.

We can predict an entity's future behavior based on its history behavior. We use $T_2$ to denote the influence of history behavior for trust computation. After an interaction between a subject and an object, there is feedback information from the owner of object. The information will be stored by the object, which can impact the subject's trust computation next time.

Time can affect trust value of an entity, for example, an entity's trust value is different in the daytime or at night. We use $T_3$ to denote influence of time for trust computation.

Network condition can also affect trust computation. We use $T_4$ to denote the influence of network condition to trust computation.

### B *Multi-factors Trust Computation*

Given IP attribute affects trust computation, we suppose that IP attribute has four possible cases. Trust interval is then divided into four disjoint subintervals in which every case of the IP attribute corresponds to a subinterval. For example, an entity's trust value which corresponding to IP is shown in Table I in which the trust interval is A= [0, 1].

TABLE I.

TRUST INTERVAL CORRESPONDING TO IP ATTRIBUTE

| IP attribute | Very safe | Safe | Moderate | Dangerous |
|---|---|---|---|---|
| Trust interval | [0.9,1] | [0.6,0.9) | [0.3,0.6) | [0,0.3) |

TABLE II.

TRUST INTERVAL CORRESPONDING TO HISTORY BEHAVIOR

| History behavior | Trusted | Moderate | Attempted hostile | Hostile |
|---|---|---|---|---|
| Trust interval | [0.8,1] | [0.5,0.8) | [0.2,0.5) | [0,0.2) |

According to interaction results between entities, the history behavior can be divided into four possible cases, such as trusted behavior, hostile behavior, attempted hostile behavior and hostile behavior. Every case corresponds to a subinterval. For example, an entity's trust value corresponding to history behavior is shown in Table II in which the trust interval is A= [0, 1].

There are different methods to evaluate history behavior factor according to different application. In this paper, we use the following method to evaluate history behavior factor in trust computation. Assume that a subject's trust value last time is T. The subject accesses an object, and the object's owner gives satisfaction degree on the subject. The satisfaction degree is denoted as S, and $S \in [0,1]$. The actual trust value of the object on the subject is $T_a = T \times S$. According to Table II, we can see $T_a \in K_i$, in which $K_i = [a,b]$ is an interval in Table II. The trust value corresponding to history behavior factor $T_2$ is a random value between $a$ and $T_a$. Namely, $T_2 \in [a, T_a]$. If an entity's history behavior is good and its actual trust value $T_a$ is high, then the trust value corresponding to history behavior factor is also high.

The computation of trust value which corresponding to history behavior factor can be described as follows:

Step1: We divide trust into $i$ intervals, and every interval $K_i = [a_i, b_i]$.

Step2: If it is the first time a subject interacts with the object, then the trust value which corresponding to history behavior factor $T_2 = 0.5$. Otherwise, we read the subject's trust value T at last time from history information which stored by the object.

Step3: According to feedback satisfaction degree S, we can compute the actual trust value $T_a = T \times S$.

Step4: We select the trust interval $K_j = [a_j, b_j]$ to make $T_a \in K_j$.

Step5: The trust value corresponding to history behavior factor $T_2$ is randomly generated, and $a_j \le T_2 \le T_a$.

Generally speaking, access behavior has time characteristics. According to history access behavior, we establish the relationship of time factor and trust. We divide access time into four periods: frequent period, moderate period, rare period and impossible period. Trust value which corresponding to access time is shown in Table III.

We consider network condition as a factor which affects trust computation. Network condition mainly includes bandwidth and channel security. We divide network condition into four cases: very safe, safe, moderate and dangerous. For example, the network condition corresponding trust value is shown in Table IV.

We calculate trust values which corresponding to IP attribute, history behavior, time and network condition, then use these values to compute a subject's trust value. There are different methods to compute $T_1, T_3, T_4$. The method in this paper is determining trust intervals which these factors correspond to, and randomly selecting trust values in these trust intervals. Trust value corresponding to each factor has more accurate selection methods. Therefore, the multi-factors trust computation method has good scalability. In this paper, we mainly focus on feasibility of the multi-factors trust computation method.

To reflect dynamicity of trust in an open environment, we compute trust with four factors introduced above as follows:

$$T = \alpha_1 T_1 + \alpha_2 T_2 + \alpha_3 T_3 + \alpha_4 T_4 \quad (1)$$

$$\text{in which } \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 = 1 \quad (2)$$

In formula (1), $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ is the respectively weight of every factor, and $\alpha_1, \alpha_2, \alpha_3, \alpha_4 \in [0,1]$. These weight values are selected according to specific case.

### *C Multi-factors weight distribution method*

Weight allocation of each factor is a multiple attribute decision problem. There are several weight allocation methods, for example, information entropy method, multi-objective optimization method and fuzzy aggregation method.

In this paper, the weights of these four factors are initialized to 1/4. The information and subjects' trust and access feedback results are considered as sample information and stored in files. When the sample reaches a certain number, then we use information entropy weight allocation method [18].

TABLE III.

TRUST INTERVAL CORRESPONDING TO TIME

| Access time | Frequent | Moderate | Rare | Impossible |
|---|---|---|---|---|
| Trust interval | [0.8,1] | [0.5,0.8) | [0.2,0.5) | [0,0.2) |

TABLE IV.

TRUST INTERVAL CORRESPONDING TO NETWORK CONDITION

| Network condition | Very safe | Safe | Moderate | Dangerous |
|---|---|---|---|---|
| Trust interval | [0.9,1] | [0.6,0.9) | [0.3,0.6) | [0,0.3) |

Following is the procedure of information entropy method:

Step1: Determine sample information which need to deal with, and extract every factor's data.

Step2: Establish fuzzy similarity relationship.

Step3: Use fuzzy equivalence closure method to achieve fuzzy equivalence matrix, and then determine the classification number of factors.

Step4: Determine the mutual information content of each factor at every confidence level.

Step5: According to information content of factors, determine their weight value.

Every entity built its history access record table which stores entity's history behavior which provides decision support for future behavior trust computation. The history access record table includes subject, time, behavior and trust value. Every factor's weight and the calculated trust value will be stored as files.

The formula we have proposed describes factors that can affect trust computation including entities' IP attribute, time, history behavior and network condition. It also depicts dynamicity and uncertainty of trust in open and dynamic systems.

## IV. ACCESS CONTROL MODEL BASED ON MULTI-FACTORS TRUST

### *A Mapping between Trust and Permission*

In this section, we describe how a trust value is mapped to access permissions for providing fine-grained access control over sensitive resources. Meanwhile, access permissions can be dynamically adjusted based on the change of the trust values.

Assume that an object's permission set is P. We divide the trust of object's owner on a subject into k intervals, namely, $T=(T_1,T_2,...,T_k)$. Then, access permission set AP can be represented as

$AP = \{\phi, \{read\}, \{read, write\}, ..., \{read, write, app, exe\}\}$ in which the number of elements of AP is k, which is equal to the number of trust intervals and each trust interval corresponds to an access permission set. For instance, if trust interval [0, 0.2] corresponds to $\phi$, subjects whose trust quantification value belong to [0, 0.2] would have no access permission to the object. If trust interval [0.4, 0.6] corresponds to {read, write}, subjects whose trust quantification value belong to [0.4, 0.6] would have the read and write permissions to the object.

In an open environment, a subject's permission is dynamic with the change of its trust value. When there is deception, the owner of the resource can modify mapping relationship between the access permission set and the trust intervals.

### *B Multi-factors Trust based Access Control Model*

The mapping method mentioned above is used for access control and an access control model based on multi-factors trust is thus proposed. The model includes three layers: request management, access control management and access feedback management. The model also includes four modules: subject request, multi-factors trust computation, access permission mapping and access feedback. The model is shown in Figure 1.

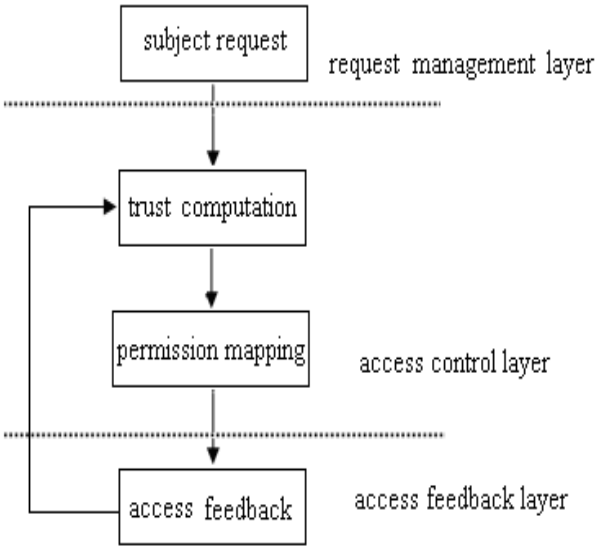



Figure 1. Multi-factors trust based access control model

The request management layer is mainly responsible for access request scheduling.

The main function of the trust computation module is to divide trust interval and compute trust value. Based on several related factors and access feedback information, it would compute the new trust value of a subject.

The permission mapping module reads the mapping table between a subject's trust value and access permissions and assigns corresponding permissions to the subject.

In the access feedback module, the feedback information from access behavior is sent to the trust computation module, which can be referenced for the next access control.

To provide global network environment information for every entity, we apply a network performance matrix M which is an N×N matrix. M [i, j] of the matrix indicates network performance between i and j which mainly includes network speed and channel security. When trust value is computed, entities get network performance information from the matrix which can be computed according to trust interval.

## V. SIMULATION RESULTS

To verify the validity of the multi-factors trust based access control model, a simulation is developed using Java program. Assume that there are 2 subjects and 8 objects. Every factor's trust interval is shown in TABLE I, II, III, IV. Permission mapping relationship is shown in TABLE V.

In the experiment, we assume that every honest subject's trust value is randomly initialized to a value between 0.5 and 1, and hostile subject's trust value is randomly initialized to a value between 0.9 and 1. After each access, the satisfaction degree of the object to the honest subject is $S \in [0.9,1]$ and that of the object to the hostile subject is $S' \in [0,0.2]$. Access feedback result denotes behavior evaluation of an object on a subject. History access record table is created for every object. Network matrix is created to denote network condition.

Experiment 1: Subject A and subject B are both honest nodes. Subject A and objects are in very safe network segment. Subject B and objects are in safe network segment. Two honest subjects randomly access objects 10 times respectively. The experiment executes 50 times and we randomly extract one of them. The permission change of subject A and subject B is shown in Figure 2.

TABLE V.

PERMISSION MAPPING

| Trust interval | [0,0.2) | [0.2,0.4) | [0.4,0.6) | [0.6,0.8) | [0.8,1) |
|---|---|---|---|---|---|
| Permission ID | 0 | 1 | 2 | 3 | 4 |
| Permission set | ∅ | { read } | { read, write } | { read, write , app } | {read, write, app, exe } |

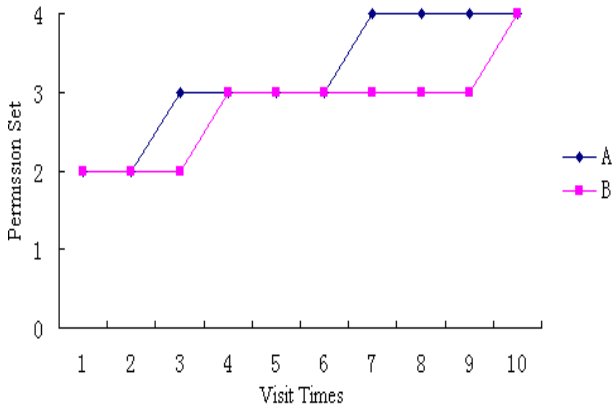



Figure 2. Subjects' permissions corresponding to different IP

We can see from Figure 2, since subject A and B are both honest nodes and they have good behavior, their permission levels are gradually increasing. Because the IP of subject A is in the network segment whose security is higher than the IP of subject B, subject B needs more visit times than A to achieve the same permission.

Experiment 2: Subject A and subject B are both hostile nodes. Subject A and objects are in safe network, but subject B and objects are in dangerous network. Two honest subjects randomly access objects 10 times. The experiment executes 50 times and we randomly extract one of them. The permission change of subject A and subject B is shown in Figure 3.

We can see from Figure 3, since subject A and B are both hostile nodes and they have bad behavior, their permission level drops fast. Because subject B is in dangerous network, permission of subject B drops faster than subject A.

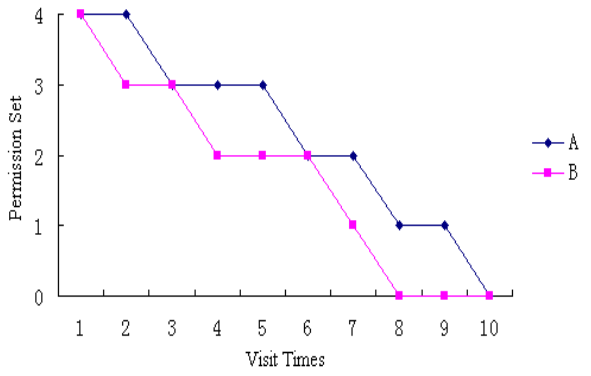



Figure 3. Subjects' permissions corresponding to different network condition

Experiment 3: Subject A and subject B are both hostile nodes. Access time of subject A is in the frequent or moderate time period, and subject B access time is in the rare or impossible time period. Two honest subjects randomly access objects 10 times. The experiment executes 50 times and we randomly extract one of them. The permission change of subject A and subject B is shown in Figure 4.

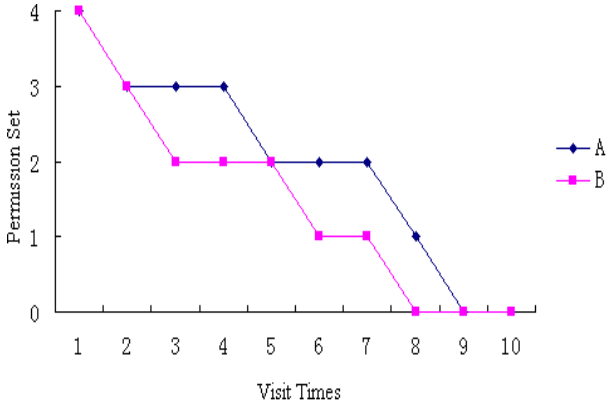



Figure 4. Subjects' permissions corresponding to different time

We can see from Figure 4, since subject A and B are both hostile nodes and they have bad behavior, their permission level drops fast. Because subject B access time is in rare or impossible time period, the permission of subject B drops faster than subject A.

Experiment 4: Subject A is an honest node and subject B is a hostile node. Subject A and B randomly access objects 10 times respectively. From the fourth access, subject A becomes a hostile node and has bad access behavior. From the fourth access, subject B becomes an honest node and has good access behavior. The experiment executes 50 times and we randomly extract one of them. The permission change of subject A and subject B is shown in Figure 5.

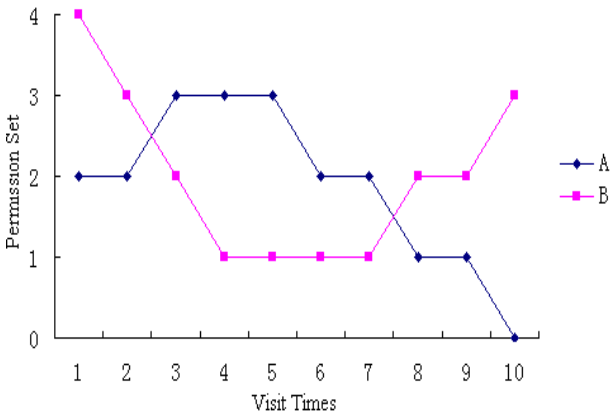



Figure 5. Subjects' permissions corresponding to history behavior

We can see from Figure 5, at beginning, trust value of subject A is randomly initialized to 0.58, and it has good behavior. The feedback satisfaction degree is high and the permission level is gradually increasing. From the fourth access, subject A has hostile behavior, and the permission level drops fast. The trust value of subject B is randomly initialized to 0.96, and it has hostile behavior. The feedback satisfaction degree is low and the permission level drops fast. From the fourth access, subject B has honest behavior, and the permission level is gradually increasing.

We can conclude that the access control model based on multi-factors is feasible. It can reflect trust's dynamic nature and manage access control permission.

## VI. Conclusion and Future Work

In open and dynamic network applications, not every resource requestor can be known beforehand by the resource owners because of the open property. Trust can be used as a tool to reduce the complexity of making access decisions, which can be accomplished by using trust to provide security. By considering several factors which affect trust, we computed trust using several factors, and we proposed an access control model based on multi-factors trust. In the model, we described how trust values can be mapped to access permissions. In the future, we will design the mapping algorithm in the model to make it more suitable for fine-grained access control.

## Acknowledgement

This work has been supported by Beijing Education Commission Science and Technology Development Fund #KM201010005027.

## References

[1] X. Ni and J. Luo, “A Trust Aware Access Control in Service Oriented Grid Environment”. Proc. 6th International Conference on Grid and Cooperative Computing, 2007.

[2] A. Alfarez and H. Stephen, “Supporting Trust in Virtual Communities”. Proc. 33rd Annual International Conference on System Sciences, Vol. 6, Maui, Hawaii, 2000.

[3] P. Samarati etal, “Access control: Policies, models, and mechanisms”. In Foundations of Security Analysis and Design, LNCS, Vol. 2171, Springer-Verlag, pp.137–196, 2001.

[4] L. Snyder, “Formal models of capability-based protection systems”. IEEE Trans. Computers, Vol. 30, pp. 172-181,1981.

[5] X. Qian, L. T. F, “A MAC policy framework for multilevel relational database”. IEEE Transactions on Knowledge and Data Engineering, 8(1), pp. 1-14, 1996.

[6] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, “Role-based access control models”. IEEE Trans.Computers, Vol. 29, pp.38- 47, 1996.

[7] G. Ahn and R. Sandhu, “Role-based authorization constraints specification”. ACM Trans on Information and System Security, pp. 21-30, 2000.

[8] K. Taylor and J. Murty, “Implementing role based access control for federated information systems on the web”. Pro. Australasian Information Security Workshop, Adelaide, Australia, 2003.

[9] X. Wang, J. Luo, A. Song and T. Ma, “Semantic Access Control in Grid Computing”. Proc. 11th International Conference on Parallel and Distributed Systems, 2005.

[10] D. Chadwick, A. Otenko, and E. Ball, “Role-Based Access Control with X.509 Attribute Certificates”. IEEE Internet Computing, 7(2), pp. 62–69, 2003.

[11] T. Alessandra et al, “A semantic context-aware access control framework for secure collaborations in pervasive computing environments”. Proc. ISWC2006, Athens, GA, USA, pp. 473-486, 2006.

[12] Y. Guo, H. Fan, Q. Zhang and R. Li, “An Access Control Model for Ubiquitous Computing Application”. Proc. 2nd International Conference on Mobile Technology, Applications and Systems, Guangzhou, China, 2005.

[13] S. Chakraborty and L. Ray, “TrustBAC: integrating trust relationships into the RBAC model for access control in open systems”. Proc. 11th ACM symposium on Access control models and technologies, pp. 49–58, 2006.

[14] F. Feng, C. Lin, D. Peng and J. Li, “A Trust and Context Based Access Control Model for Distributed Systems”. Proc. 10th IEEE International Conference on High Performance Computing and Communications, pp. 629-634, Washington, 2008.

[15] A. Z. Lin, E. Vullings, and J. Dalziel, “A Trust-based Access Control Model for Virtual Organizations”. Proc. 5th International Conference on Grid and Cooperative Computing Workshops, 2006.

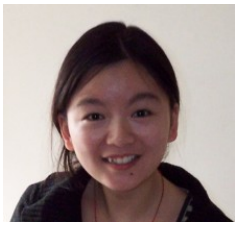
[16] R. Bhatti, E. Bertino and A. Ghafoor, “A Trust-based Context-Aware Access Control Model for Web-Services”. Proc. IEEE International Conference on Web Services, 2004.

[17] J. Jiang, H. Bai and W. Wang, “Trust and cooperation in peer-to-peer systems”. Springer-Verlag, pp. 371-378, 2004.

[18] D. Huang. Means of Weights Allocation with Multi-Factors Based on Impersonal Message Entropy [J]. Systems Engineering-Theory Methodology Applications, 12(4), pp. 321-324, 2003.

**Shunan Ma** is a Ph.D candidate in the College of Computer Science and Technology at Beijing University of Technology, Beijing, China. She received her B.S. degree in Qufu Normal University in 2004 and M.S. degree in Jiangnan University in 2007, respectively. Her research interests include network security and distributed network technology.

**Jingsha He** is a professor of the School of Software Engineering at Beijing University of Technology (BJUT) in Beijing, China. He received his doctorate from the University of Maryland at College Park in 1990. Prior to joining BJUT in 2003, he worked for IBM, MCI Communications and Fujitsu Laboratories engaging in R&D of advanced networking and computer security. His interests include methods and techniques that can improve the security and performance of the Internet.



**Feng Gao** received her B.S. degree in e-business engineering from PLA Information Engineering University in 2003. She is currently working towards the Ph.D. degree with the School of Computer Science and Technology, Beijing University of Technology. Her research interests include network security, privacy protection and trust etc.

# A Private Data Transfer Protocol Based On A New High Secure Computer Architecture

Gengxin Sun
International College of Qingdao University, Qingdao, China
Email: sungxmail@gmail.com

Fengjing Shao and Sheng Bin
Information Engineering College of Qingdao University, Qingdao, China
Email: {sfj, binsheng}@qdu.edu.cn

***Abstract*—Focusing on the characteristics of the new high secure architecture of network computer, an operating system with internal network structure is designed. The operating system contains two subkernels: local kernel and network kernel, the two subkernels run individually in two subsystems. In order to communicate between two subsystems securely, an inter-subsystem private data transfer protocol is proposed and implemented in this paper. The private protocol is a connection-oriented protocol, it can provide reliable end-to-end connectivity Protocol format and protocol connection management based on signature verification are elaborated. Combining with shared transit cache which mounted on the shared bus, the private data transfer protocol can ensure data to be transferred safely and inerrably between subsystems. The effectiveness of the private data transfer protocol is verified by the results of final experiments.**

***Index Terms*—private protocol, data transfer, operating system, computer architecture, bus bridge, shared transit cache, protocol performance evaluation**

## I. INTRODUCTION

With the development of computer network, the network security is becoming more and more important. A series of network security technologies have been used to protect the computer security, such as computer virus scan technology, intrusion detection technology [1,2], software or hardware encryption technology [3,4], secure computer architecture [5], etc.

Based on the traditional architectures of computer such as Von Neumann [6], network and local storage are connected to the same bus, which causes potential security issue because data in local storage might be theft if network intruder has the control of the system bus. Aiming at this security problem, many methods have been proposed, such as TPM (Trusted Platform Module) [7,8], information exchange and encryption [9]. Although these methods enhanced the security of computer, they could not address the fundamental problem due to the vulnerability of Von Neumann architecture.

A new high secure architecture of network computer is proposed by Fengjing Shao [10]. The new computer architecture has a single CPU and two physically isolated high-speed system buses (local bus and network bus), ensure only one bus can be connected to CPU at the same time, a Bus Bridge [11] is designed.

In this new architecture, computer system is divided into tow subsystems. All the network devices and other devices mounted on the network bus form into a network subsystem which connects with the Internet. All the storage devices and other devices mounted on the local bus form into a local subsystem which is isolated from the Internet. So even if the network intruder suddenly gets the whole control of network bus in network subsystem, only the temporary information of the network subsystem is exposed, while the local subsystem is left intact. Thus, hardware-level isolation can effectively ensure the security of sensitive data in local subsystem.

As the new secure computer architecture has one CPU and two subsystems, there should be a befitting operating system to support this architecture. In order to enhance the security of the computer architecture, an operating system with internal network structure is designed, the operating system contains two subkernels: the local kernel and the network kernel. The two subkernels run individually in two subsystems, and they are coordinating relationship rather than subordinate relationship. The relationship between the operating system and the new secure computer architecture is shown in Fig. 1.

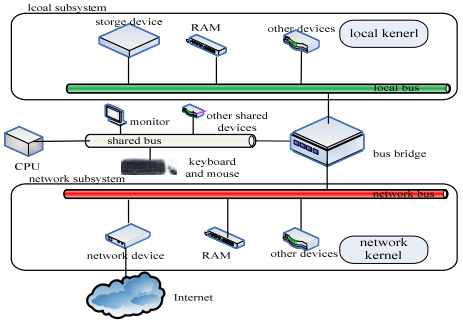



Figure 1. The operating system based on new high secure computer architecture

* Supported by the High-Tech Research and Development Program of China (863 Program) (2006AA01Z110)
** Corresponding author: Fengjing Shao (sfj@qdu.edu.cn).


doi:10.4304/jnw.7.1.179-186

The new secure computer architecture combined with the operating system can effectively prevent network intrusions from invading local subsystem. But at the same time, it also prevents the data that user gets from Internet in the network subsystem from entering the local subsystem, and the data that user wants to send to Internet in local subsystem from entering the network subsystem. In order to implement communication between two subsystems, there should be a secure communication mechanism. Currently there are many secure communication technologies in bus systems [12], but none of them can be used in the new secure computer architecture. In order to communicate between two subsystems securely, an inter-subsystem private data transfer protocol is designed and implemented in this paper. With the private data transfer protocol, data can be transferred between the two subsystems and the network intrusion can still be isolated from the local subsystem.

## II. BUS BRIDGE

The function of the bus bridge is to connect the shared bus with one of the peripheral buses. The goal of the bus bridge is:

- The bus bridge should make sure that the shared bus can connect one of the peripheral buses in anytime;
- The bus bridge should make sure that only one of the peripheral buses is connected to the shared bus at the same time;
- When CPU needs to connect the other peripheral bus, the bus bridge can cuts current connection and connects shared bus with the other peripheral bus.

### *A. Structure of Bus Bridge*

Based on the functionality, the bus bridge is designed as Fig. 2.

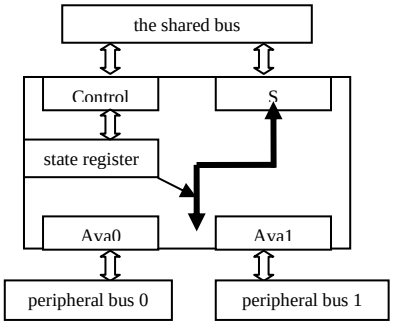



Figure 2. Structure of bus bridge

The bus bridge includes two slave ports (Control and S), two master ports (Ava0 and Ava1) and one state register.

The function of the Control port is to receive the instructions for the bus bridge from CPU, or send the data of the state register to CPU. The function of the S port is to receive/forward all Avalon signals from/to the shared bus. All these signals of the S port are bridged with their corresponding signals of the Ava0 or Ava1 port, which connects the peripheral bus with all Avalon bus signals, depending on the instruction type. Therefore, a path between the shared bus and the peripheral buses is built. A state register with five bits exists in the bus bridge. The lower four bits of the register store work modes; the higher one bit, called flag bit, stores the flag that helps describe which peripheral bus is connected to the shared bus through the bus bridge currently. If the flag bit is set as 0, all the signals of the S port connect with their corresponding signals of the Ava0 port. Therefore the shared bus connects peripheral bus 0, and the system works in area 0; otherwise the flag bit is set as 1 and the system works in area 1.

The data of state register shows as table 1.

TABLE I. DATA OF STATE REGISTER

| The lower 4 bit of state register | |
|---|---|
| ***value*** | ***Work mode*** |
| 0000 | Requirement for initialization |
| 0001 | Finish of initialization |
| 0010 | Maintain in current area |
| 0011 | Maintain in area 0 |
| 0100 | Maintain in area 1 |
| 1111 | System reset |
| others | reserved |
| The flag bit | |
| ***value*** | ***Work area*** |
| 0 | area 0 |
| 1 | area 1 |
| others | area 0 |

The data of state register is written by CPU, and read by CPU for detecting the state of the system.

### *B. Implement of Bus Bridge*

The implement of the bus bridge includes four modules:

- Write transfer

Avalon fundamental slave write transfer [13] describes how CPU writes state register through the Control port. Write transfer is completed in a single bus cycle. CPU writes the work mode and work area according to table 1.

In the write transfer of the bus bridge, the CPU passes signals *writedata*, *write_n* and *address* to the bus bridge through the shared bus. The signal *writedata* carry command code from CPU; the signal *write_n* can enable the write operation; the shared bus set the signal *chipselect*, which selects the bus bridge, according to the signal *address*. Once the signals *write_n* and *chipselect* are both available, the bus bridge reads the lower five bits of the signal *writedata*, and writes in the state register. The pseudo codes of write transfer show as following:

```
always @ (posedge clk)
begin
        if (!write_n && chipselect = = 1)
           the state register <= writedata[4:0] ;
end
```

- Read transfer

Avalon fundamental slave read transfer describes how CPU reads state register through the Control port, Read transfer is completed in a single bus cycle. CPU reads the work mode and work area according to table 1.

In the read transfer of the bus bridge, the CPU passes signals *read_n* and *address* to the bus bridge through the shared bus. The signal *read_n* can enable the read

operation; the shared bus set the signal *chipselect*, which selects the bus bridge, according to the signal *address*. Once the signals *write_n* and *chipselect* are both available, the bus bridge sends data of the state register to the lower five bits of the signal *readdata*, which carry data to CPU. The pseudo codes of read transfer show as following:

```
always @ (posedge clk)
begin
    if (!read_n && chipselect == 1)
        readdata[31:0] <= {27'b0,the state register} ;
end
```

- Connection between S port and master ports

All the signals of the S port connect with their corresponding signals of the master ports; therefore the shared bus connects peripheral bus. For example, the S port gets a signal *writedata*, which has 32 bits, from the shared bus. Its corresponding signal in the Ava0 port is named *ava0_writedata*. The connection in verilog program is set as follow:

```
ava0_writedata[31:0] <=  writedata[31:0]
```

The flag bit hold the flag of the work area, if it's 0, the system works in area 0; otherwise the system works in area 1. In the Verilog program of the bus bridge, the change of flag bit is one of the trigger condition of always block for connecting the S port signals and master port signals. When the state of flag bit changes, the bus bridge automatically reconnects the S port and the master port.

- Reset

When the system is reset, the state register is set as 01111, which means the system work in area 0, and the work mode is "system reset"; the bus bridge bridges the S port signals and the Ava0 port signals.

## III. Internal Structure of Operating System

The operating system with internal network structure consists of two subkernels: the local kernel and the network kernel. The two subsystems of the new secure computer architecture are managed respectively by two kernels, that is, network kernel manages network subsystem and local kernel manages local subsystem.

Inter-process communication of the operating system is similar to Inter-computer communication of network. A kernel is like a subnetwork, there is only a Router process, which acts the role of Router in network, in each kernel, and other processes except Router process are called as user process. Router process is the data transfer interfaces between the subkernels, a user process can only communicate with a user process of another kernel through Router process of the same kernel. Topological structure of the operating system is shown in Fig. 3.

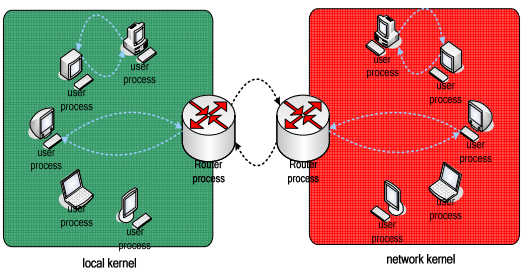



Figure 3. Topological structure of the operating system

The Router process communicates with user processes in the same subkernel via signal mechanism. The two Router process in different subkernels communicate with each other via message mechanism. Because only one subkernel can use the CPU at any time, i.e., only one kernel is active at the same time, the two Router process can't directly communicate with each other via message. Therefore, they must use the shared transit cache mounted on the shared bus to transfer messages indirectly. The process of inter-subkernel data transfer is shown in Fig. 4.

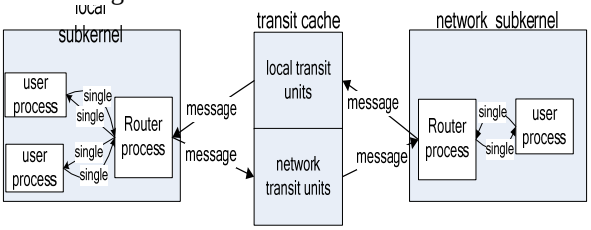



Figure 4. The process of inter-subkernel data transfer

## IV. Shared Transit Cache

The shared transit cache mounted on the shared bus is managed and operated by the two Router processes. And it plays a role of bridge in the communication between the Router processes which must transfer data indirectly through the shared transit cache.

### A. Shared Transit Cache Structure

The shared transit cache is divided into three parts: transit cache state block, transit head area and transit block area. In the process of shared transit cache initialization, transit heads are set from the end of transit cache state block which is in the low-end of cache area and their corresponding transit blocks are divided in the high-end of cache area at the same time. All the transit blocks form the transit block area, while all the transit heads form the transit block area. Every transit head links its corresponding transit block to form a transit unit. There are two kinds of transit units: network transit unit which is made up of a network transit head and a network transit block, and local transit unit which is made up of a local transit head and a local transit block. The transit cache structure is shown in Fig. 5.

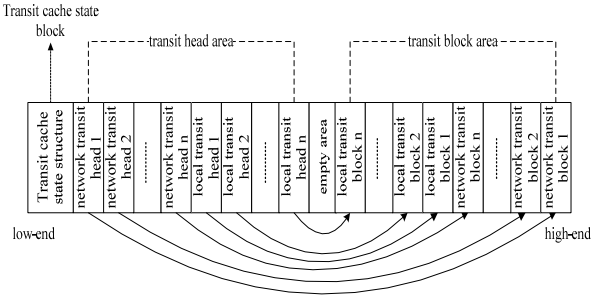



Figure 5. Shared transit cache structure

The transit head area is made up of network transit heads and local transit heads. The two kinds of heads have the same structure that is shown in the table 2. The transit head is used to descript attributes of the transit unit,

store the message head and link all the same type of transit units to be a doubly linked circular list. All the network transit units are linked to be a network doubly linked circular list and all the local transit heads are linked to be a local doubly linked circular list.

TABLE II. TRANSIT HEAD STRUCTURE

| field name | field introduction |
|---|---|
| trans_idle | idle marker, it indicates if the transit unit is used. |
| trans_num | transit unit number, it is the transit unit ID. |
| mes_head | message head, it is used to store the message head. |
| trans_prev | transit head forward pointer, it points to the transit head before itself. |
| trans_next | transit head backward pointer, it points to the transit head after itself. |
| trans_block | transit block pointer, it points to the corresponding block pointer. |

The transit block area only contains two kinds of blocks: network transit block and local transit block. All the blocks are linked to corresponding transit heads. The function of transit block is to store the data of message.

The transit cache state block contains a transit cache state structure which mainly has three fields: network linked list head pointer, local linked list head pointer and residual connection count. The network linked list head pointer points to the first idle transit unit of the network doubly linked circular list. It is used when Router process of network subkernel receives message or Router process of local subkernel sends message. The local linked list head pointer points to the first idle transit unit of the local doubly linked circular list. It is used when Router process of local subkernel receives message or Router process of network subkernel send message. The residual connection count indicates that how many data transfer connections the shared transit cache can still support to create. It is used for the Router process to create a data transfer connection.

### *B. The Operation and Management of Shared Transit Cache*

There are two kinds of transit units in a doubly linked circular list of transit units (both the network doubly linked circular list and the local doubly linked circular list): idle transit unit and non-idle transit unit. In the beginning, every transit unit in the list is the idle transit unit. When an idle transit unit is needed, it will be obtained from the head of the list, become a non-idle transit unit, and then be inserted into the tail of the list. Therefore, the idle transit units are in the front of the list while the non-idle transit units are in the end of the list. The linked list head pointer (both the network list head pointer and the local list head pointer) in the transit state structure points to the first idle transit head of the list. The doubly linked circular list of transit units is shown in the Fig. 6.

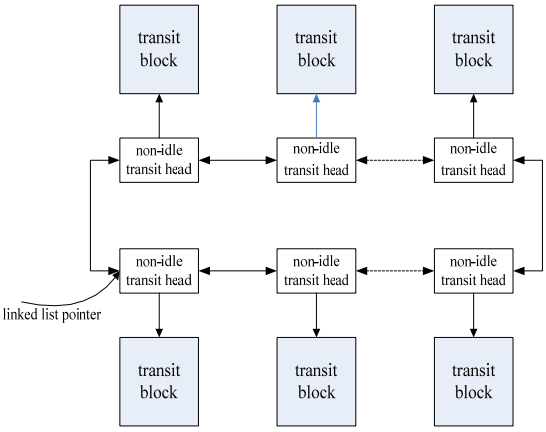



Figure 6. The doubly linked circular list of transit units

A data transfer connection has a pair of transit unites (a network transit unit and a local transit unit) which have the same ID to transfer message. The network transit unit is used by Router process of network subkernel to receive the message that Router process of local subkernel sends. Similarly, the local transit unit is used by Router process of local subkernel to receive the message that Router process of network subkernel sends. These two units are released after the connection is closed. Residual connection count in the transit state structure indicates how many pairs of idle units still can be used.

Before creating a connection, Router process first check the residual connection count to find if there is a pair of idle units which can be used for the new data transfer connection. If there is, it will find an idle network transit unit by the network transit head list pointer, modify the idle marker of the unit to make it turn into a non-idle unit, and then insert it into the end of the list. After that the Router process traverses the local doubly linked circular list to find the idle local transit unit whose number is same to the network transit unit's, and then modify the it's idle marker and insert it into the end of the list. All the messages of the connection are transferred via these two transit units.

If the Router process needs to send a message in a connection, it should traverse the doubly linked circular list of target subkernel from the end to find the transit unit that belongs to this connection, and then put the message into the unit. When the Router process wants to receive message in a connection, it will traverse the doubly linked circular list of itself subkernel from the end to find the transit unit which belongs to the connection, and then get the message from the unit.

After Router process closes a connection, it restores the idle marker of the pairs of the transit units that belong to the connection, and inserts them into the head of their doubly linked circular list to release them.

## V. PRIVATE DATA TRANSFER PROTOCOL FOR INTER-SUBSYSTEM

In order to prevent the network intrusions from entering local subsystem from network subsystem via shared transit cache, a private inter-subsystem data

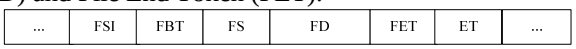
transfer protocol is used to control the inter-subsystem data transfer.

The private protocol is a connection-oriented protocol, it can provide reliable end-to-end connectivity and ensure data to be transferred safely and inerrably between subsystems.

### *A. Protocol Format*

In the process of transferring data, data is packeted as file which is written into or read from shared transit cache. The private protocol format is in manner of data stream.

The protocol format is fairly straightforward, which is shown in Fig. 7, including File Structure Information (FSI), File Begin Token (FBT), File Size (FS), File Data (FD) and File End Token (FET).




FSI:File Structure Information; FBT:Begin Token; FS:File Size; FD:File Data; FET:File End Token; ET Transfer End Token

Figure 7. Private Protocol Format

The private protocol is a peer two-way transfer protocol. The Router process of one subkernel acts as sender, the Router process of another sunkernel acts as receiver at the same time. The sending and receiving algorithm is accordant.

The sending algorithm as follows:

```
send(dest)
   if(dest is a file)
      result:=sendFile(dest);
      push(ET);
   else
      while(untransferred files exist) do
        newFile:=get an untransferred file;
        result:=sendFile(newFile);
        if(newFile is the last file in dest)
             push(ET);
        else
            push(FET);
        end if;
        if(result<0)
           break;
        end if;
      end while;
  end if;
return result;
sendFile(file)
     acquire(FSI); push(FSI);
     acquire(FBT);push(FBT);
     acquire(FS);push(FS);
     rusult:=push(FD);
return result;
```

The receiving algorithm as follows:

```
receive()
   buf:=new_buffer;
   whiel(Shared Transit Cache is not empty) do
      character:=read a byte from Shared Transit Cache;
      if(character is FBT)
          receiveFile(buf);
      else if(character is ET)
          break;
      else if(character is FET)
          receive();
          break;
      end if;
      append the character to the buf;
   end while;
return;
receiveFile(buf)
     Filename:=get the name from the buf;
     FileSize:=read bytes from Shared Transit Cache;
     read FileSize bytes from Shared Transit Cache and
     create the file with name of filename
return;
```

Firstly, communication of both ends packet data as a file or a group of files, then, through shared transit cache, sender send data with the sending algorithm, accordingly, receiver receives data with receiving algorithm. If a file is the last file of file group, the transfer finish mark ET would be sent after transfer completion of the file, otherwise, sending file finish mark FET.

### *B. Protocol Connection Management Based on Signature Verification*

As a connection-oriented protocol, it must set up end-to-end connection before transferring data. Because of the demand of the new secure computer architecture, when using the private protocol for transferring data, both ends of connection should be identity authentication for validation of identity. A switch of transit cache can be used to control whether setting up connection of both ends, for the connection request which do not pass identity authentication, the switch would close transit cache and the connection request would be refused. It requires the establishment of a security authentication mechanism to meet the demand. In the private protocol, connection management based on signature verification is proposed and designed.

Digital signature technology is proposed on the basis of public-key cryptosystem, It can ensure that only the sender can produce information which can not be faked by others, sender uses private-key to sign the message which would be sent, the information which had been signed is a proof of authenticity for a message which is sent by sender. Another important function of digital signature is to verify the integrity of data, because digital signatures can prevent third-party from forging or altering message which had been signed. The private protocol use digital signature technology based on public-key for adding authentication mechanism in the connection management process, therefore, security of protocol is increased and high security requirements of data transfer is met.

The private protocol adopts digital signature technology based on RAS algorithm. The first researchers to discover and publish the concepts of Public-Key Cryptography(PKC) were Whitfield Diffie and Martin Hellman [14], and The Diffie-Hellman key agreement protocol (also called exponential key agreement) was developed, The protocol allows two users to exchange a secret key over an insecure medium without any prior

secrets. The Diffie-Hellman key agreement protocol provided an implementation for secure public key distribution, but didn't implement digital signatures. After reading the Diffie-Hellman paper [14], three researchers at MIT named Ronald Rivest, Adi Shamir, and Leonard Adleman (RSA) began searching for a practical mathematical function to implement a complete PKC approach [15]. After working on more than 40 candidates, they finally discovered an elegant algorithm based on the product of two prime numbers that exactly fit the requirement for a practical public key cryptography implementation.

The basic protocol of digital signature based on RSA is very simple. Roughly speaking, the basic idea is as follows. The protocol has two system parameters $p$ and $g$. They are both public and may be used by all the users in a system. Parameter $p$ is a prime number and parameter $g$ (usually called a generator) is an integer less than $p$, with the following property: for every number $n$ between 1 and $p$-1 inclusive, there is a power $k$ of g such that $n = g^k \bmod p$.

Prior to execution of the protocol, the two parties A and B each obtain a public/private key pair and a certificate for the public key. During the protocol, A computes a signature on certain messages, covering the public value $g^a \bmod p$. B proceeds in a similar way. Even though the third-party is still able to intercept messages between A and B, the third-party cannot forge signatures without A's private key and B's private key. Hence, the enhanced protocol defeats the man-in-the-middle attack.

The way of protocol connection management based on signature verification is adopted for ensure the connection of stability, reliability and security. The private protocol uses four-way handshake to establish connection.

- Creating connection

In a data transfer connection, the data transfer process whose responsibility is to send data is called sender Router process, and the one whose responsibility is to receive data is called receiver Router process. For creating a data transfer connection, the sender Router process obtains a private key, then generating connection request message with signature through the private key. After receiving connection request message, receiver Router process sends validation message with signature to sender Router process, sequence number of validation message is equal to the sequence number of request message plus 1. Sender Router process would validate signature of validation message, if signature is validation, the connection is semi-connected status. Then, receiver Router process sends its connection request message with signature to sender Router process, and randomly select a starting sequence number, sender Router process receives connection request message, it would also send validation message with signature to receiver Router process, sequence number of validation message is equal to the sequence number of request message which is sent by receiver Router process plus 1. If receiver Router process validates signature successfully, the connection between send Router process and receiver Router process would be completely established. The process of creating connection is shown in Fig. 8.

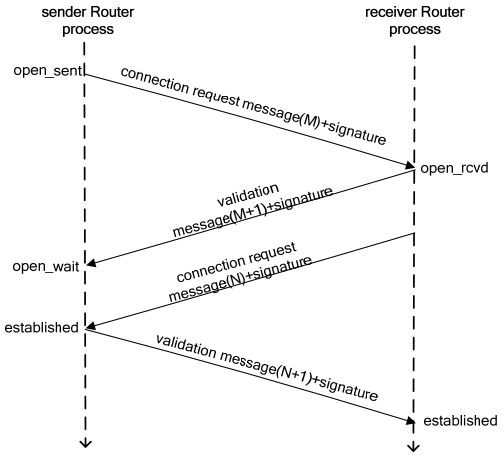



Figure 8. The process of creating connection

- Closing connection

The process of creating connection needs four-way handshake, and the process of closing connection also needs four-way handshake. For closing a data transfer connection, the sender Router process sends a connection close message, the sequence number is the next sequence number of message which would be sent. After receiving connection close message, receiver Router process sends validation message to sender Router process, sequence number of validation message is equal to the sequence number of close message plus 1. Then, receiver Router process sends its connection close message to sender Router process, the sequence number is also the next sequence number of message which would be sent. Sender Router process receives connection close message, it would also send validation message to receiver Router process, sequence number of validation message is equal to the sequence number of close message which is sent by receiver Router process plus 1. After completion of the four interactive messages, the connection also closed. The process of closing connection is shown in Fig. 9.

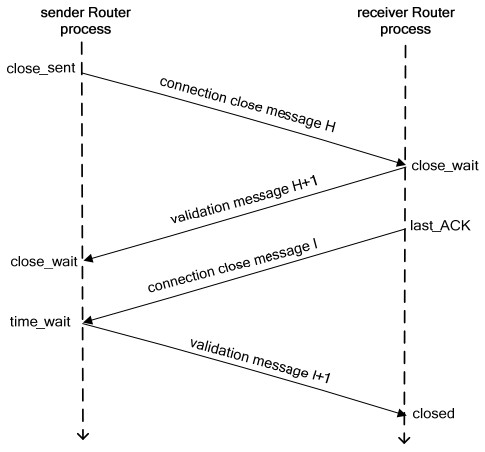



Figure 9. The process of closing connection

## VI. PROTOCOL PERFORMANCE EVALUATION

According to implementation process of the private protocol, transfer performance of the private protocol is analyzed as follows:

Transfer efficiency of the private protocol *V* is defined as the ratio of the requested effective data quantity and the total transfer data quantity.

### A. Transfer Efficiency

Theorem 1. With increase or decrease of transfer data quantity, transfer efficiency *V* increases or decreases.

Supposing transfer data includes *n* files, and every file size is *FileSize* bytes, thus, size of requested data *Demand* is defined as,

$$\text{Demand} = n \times \text{FileSize}. \quad (1)$$

Size of connection request message is *Request* bytes, size of relevant validation message is *Response* bytes. Size of data quantity of creating connection *Cmessage* is defined as,

$$\text{Cmessage} = 2\times(\text{Request} + \text{Response}) \quad (2)$$

Size of data stream prefix in protocol is *Prefix*, thus, size of actual transfer data quantity through protocol *Pdata* is defined as,

$$\text{Pdata} = n\times(\text{Prefix} + \text{FileSize}) \quad (3)$$

The total transfer data quantity *Trandata* is defined as,

$$\text{Trandata} = \text{Cmessage} + \text{Pdata} \quad (4)$$

According to (1) ~ (4), Transfer efficiency *V* is defined as,

$$V = \frac{Demand}{Trandata} = \frac{1}{1+\frac{\mathrm{Pr}efix}{FileSize}+\frac{2\times(\mathrm{Re}quest+\mathrm{Re}sponse)}{n\times FileSize}} \quad (5)$$

Because of size of connection request message and relevant validation message, and value of *Prefix* are constant, from (5), transfer efficiency *V* would change along with transfer data quantity (size of file *FileSize* or number of file *n*), and max is 1.

Theorem 2. Transfer efficiency of the private protocol *V* is in direct ratio to rate of requested data *Demand*/t.

In the private protocol, data transfer operation is serial, transfer time is in direct ratio to data traffic, the equation is defined as,

$$t = \text{const}\times\text{Trandata} \quad (6)$$

According to (5), (6), transfer efficiency *V* is in direct ratio to transfer rate, the equation is defined as,

$$V = \text{const}\times\frac{Demand}{t} \quad (7)$$

### B. Rate of Throughput

Rate of throughput is defined as the reciprocal of service time of unit data.

Theorem 3. With increase of transfer data quantity, rate of throughput of the private protocol increases.

Supposing creating connection time is $T_c$, closing connection time is $T_r$.

Firstly, setting file size *FileSize* is fixedness, and number of files *n* is increasing, then, the relationship between service time of single file $T_n$ and number of files is defined as,

$$T_n = \text{const}\times(\mathrm{Pr}efix + \text{FileSize}) + \frac{T_c+T_r}{n} + \frac{\text{const}\times(\text{Request+Response})}{n}. \quad (8)$$

Next, setting number of files is fixedness (n=6), and file size *FileSize* is increasing, the relationship between transfer time of unit data $T_b$ and file size is defined as,

$$T_b = 6\times\text{const} + \frac{T_c+T_r}{\text{FileSize}} + \frac{\text{const}\times(\text{Request+Response}+6\times\mathrm{Pr}efix)}{\text{FileSize}}. \quad (9)$$

From (8), (9), with increase of transfer data quantity (size of file *FileSize* or number of file *n*), service time of unit data decreases, and rate of throughput of the private protocol increases accordingly.

## VII. SIMULATION AND EXPERIMENT

Service capabilities of the private protocol can be further validated through simulation experiments.

We use development board based on new high secure architecture of network computer for experimenting with the private protocol. The development board which we make is shown in Fig. 10.

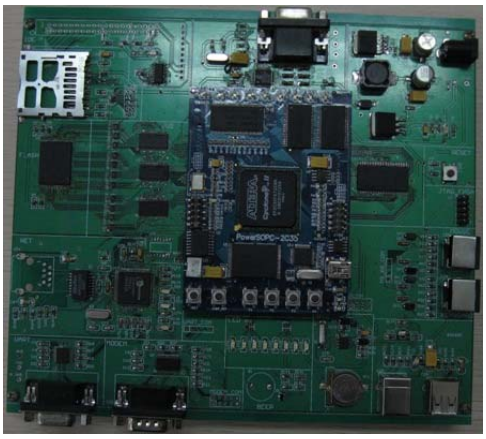


Figure 10. Development board based on new high secure architecture of network computer

When setting file size *FileSize* is fixedness, and number of files *n* is increasing, the relationship between service time of single file $T_n$ and number of files is shown as Fig. 11.

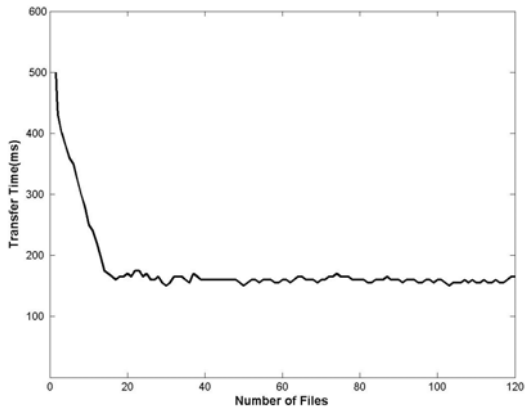



Figure 11. Relationship between service time of single file and number of file

When setting number of files is fixedness(n=6), and file size *FileSize* is increasing, the relationship between transfer time of unit data $T_b$ and file size is shown as Fig. 12.

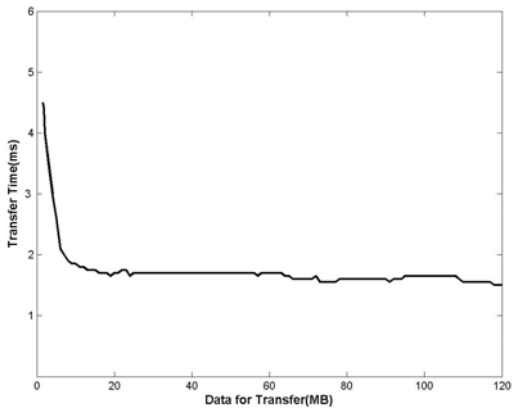



Figure 12. Relationship between service time of unit data and file size

## ACKNOWLEDGMENT

This paper is funded by the National High Technology Research and Development Program of China (863 Program) (No.2006AA01Z110) and the Technology breeding project of Qingdao (No. 09-2-3-19-chg). Thanks for the two projects.

## REFERENCES

[1] Abdoul Karim Ganame, Julien Bourgeois, Renaud Bidou and Francois Spies, "A global security architecture for intrusion detection on computer networks," Computer & Security, Vol. 27, pp. 30-47, March 2008.

[2] Ben Rexworthy, "Intrusion detections systems – an outmoded network protection model," Network Security, Vol. 2009, pp. 17-19, June 2009.

[3] Niansheng Liu, Donghui Guo, Security analysis of Public-key Encryption Scheme Based on Neural Networks and Its Implementing. International Conference on Computational Intelligence and Security, pp. 1327-1330, 2006.

[4] Prayag Narulaa, Sanjay Kumar Dhurandhera, Sudip Misrab and Isaac Woungangc, "Security in mobile ad-hoc networks using soft encryption and trust-based multi-path routing," Computer Communications, Vol. 31, pp. 760-769, March 2008.

[5] Igor Podebrad, Klaus Hildebrandt, Bernd Klauer, " List of Criteria for a Secure Computer Architecture," 2009 Third International Conference on Emerging Security Information, Systems and Technologies

[6] John von Neumann, First Draft of a Report on the EDVAC. Moore School of Electrical Engineering, University of Pennsylvania, June 30, 1945

[7] Trusted Computing Group (TCG), Trusted Platform Module (TPM) Specification Version 1.2 Revision 103, https://www.trustedcomputinggroup.org/specs/TPM/, July, 2007.

[8] IBM Research Report, the role of TPM in enterprise security. 2004

[9] Chuanjin, Wei,Qingbao, and Li,Yan Bai, "The research and realization of network teriminal information switch mechanism", CONTROL&AUTOMATION, 2005, 21(5).

[10] Shuangbao Wang, Fengjing Shao, and Robert S.Ledley, "Connputer—a framework of intrusion-free secure computer architecture", Security and Management 2006, pp. 220-225, 2006.

[11] Tiedong Wang, Fengjing Shao, Rencheng Sun, He Huang. "A hardware implement of bus bridge based on single CPU and dual bus architecture," International Symposium on Computer Science and Computational Technology, Shanghai, China, December 2008.

[12] Sascha Mühlbach, Sebastian Wallner. "Secure communication in microcomputer bus systems for embedded devices," Journal of Systems Architecture, Vol. 54, pp. 1065-1076, November 2008.

[13] Altera Corporation, Avalon Bus Specification Reference Manual, Document Version 2.3, July 2003.

[14] Whitfield Diffie and Martin Hellman, "New Directions in Cryptography," Information Theory, Vol 22, pp. 644-654, November 1976

[15] RL Rivest, A Shamir, L Adleman , "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," Communications of the ACM, Vol 21, pp.120-126, 1978

# A Robust Localization in Wireless Sensor Networks against Wormhole Attack

Yanchao Niu, Deyun Gao, Shuai Gao
National Engineering Laboratory for NGIID, Beijing Jiaotong University, Beijing China
gniux819@gmail.com, {gaody, shgao}@bjtu.edu.cn

Ping Chen
TEDA College, Nankai University, Tianjin, China
chenpingteda@nankai.edu.cn

***Abstract*—Wormhole attack is one of the most devastating threats for range-free localization in wireless sensor networks. To address this issue, we propose a robust localization scheme in wireless sensor networks against wormhole attack, called ConSetLoc, which neither complicated distance measuring devices nor extra complex encrypting algorithms is necessary. With the relationship between hop counts and geographic distance of sensor nodes in the deployment territory, we design a partition method of consistent anchors sets by the convex constraints in geometry which can reduce the effect of bad measurements on estimates, and then present a filtering strategy for the candidate locations with these anchors sets. In addition, we conduct simulation experiments for performance evaluation and the results demonstrate the proposed ConSetLoc can estimate the locations for most of sensor nodes with good accuracy and stability when wormhole attack exists in the network.**

***Index Terms*—Wireless Sensor Networks, Range-free Localization, Wormhole Attack, Hop-Distance Relationship, Convex Constraints, Consistent Set**

## I. INTRODUCTION

Localization technology in wireless sensor networks (WSNs) is one of the most important supporting techniques [1]. Over the past few years, many localization algorithms have been proposed to provide sensors' location. Out of these localization schemes, range-free localization methods have attracted much attention because no extra sophisticated devices for distance measurement for each sensor nodes are necessary. The hardware cost should be taken into consideration when WSNs have to scale to extremely large number of sensor nodes. Most of range-free localization algorithms assume that sensor nodes are deployed in a trusted region. However, many applications in hostile environments like military monitoring may have malicious attacks, in which an adversary may provide incorrect position information, or compromise a beacon node, or replay beacon information to mislead the node position estimation. Range-free localization algorithms are vulnerable in these attacks. The secure localization problem has been studied in these years, and the proposed approaches can be classified into two categories. By the authentication with a cryptographic key, the first ones can guarantee the security of location process away from some attacks, but the computation of cryptographic operation is intensive and it is tedious for key management. The other methods firstly employ detection mechanisms which can passively detect anomalies by observation and deployment knowledge and then localize sensor nodes. Most of these mechanisms need multimodal techniques, for example SeRLoc [2] works with directional antennas and ROCRSSI [3] is based on comparison of received signal strength indicator and secure enhancement method.

Among malicious attacks, wormhole attack [4] is a devastating threat, and it does not require compromising any sensor nodes and knowing cryptographic keys. The attackers use direct low-latency links (called wormhole links) to connect two or more end-points. When a message is received by one end-point, it will be forwarded to other end-points by wormhole links, and then replayed into the network. Therefore, wormhole attack can change the network topology and deteriorate the positioning accuracy for range-free localization in two aspects. On the one hand, wormhole attack can enlarge the neighborhood for the range-free localization based on neighborhood measurement like APIT [5] and DRLS [6]. On the other hand, it can shorten the shortest routing path between two nodes for those based on hop count measurement like DV-Hop [7]. A few schemes have been proposed to detect the wormhole using graph theoretic [8] and metric threshold in neighborhood [9], but these mechanisms cannot always guarantee detection of wormholes.

In this paper, aiming at the range-free localization problem under the wormhole attack, we propose a robust localization scheme, called ConSetLoc, instead of coming up with solutions to the attack detection. To reduce the large errors in ranging measurements due to wormhole attack in our scheme, we firstly focus on the hop-distance relationship between sensor nodes. The hop-distance relationship can be evaluated by an approximate recursive expression when sensor nodes are randomly deployed in

Manuscript received Aug. 9, 2010; accepted Sep. 8, 2010.
Supported by National Basic Research Program of China (No. 2007CB307101) and Natural Science Foundation of China (No. 60802016)
corresponding author: .gniux819@gmail.com


doi:10.4304/jnw.7.1.187-194

a circular region. With the above hop-distance relationship, the ConSetLoc has the following two aspects:

- We design a partition method of consistent anchors sets by the convex constraints in geometry. Each anchor can provide ranging measurement information which can be considered as a ring in geometry. A practical algorithm is developed to achieve the consistent anchors set which can obtain the convex region intersected by these rings. The candidate location of sensor nodes estimated by the convex region is consistent with ranging measurement information.
- We present a filtering strategy for the candidate locations with these consistent anchors sets, and the final location of sensor nodes will be filtered out.

Simulation experiments with wormhole attack have been conducted. The simulation results demonstrate that the proposed ConSetLoc can accurately estimate the locations for most of the nodes when wormhole attacks exist in the network, and ConSetLoc performs better than other methods in literature.

The rest of this paper is organized as follows. In Section , we present a recursive formula to evaluate the relationship between the hop counts and distance information. Section III gives the consistent anchors set partitioned by the geometry constrains and a filtering strategy for the candidate location set in proposed range-free localization algorithm ConSetLoc. In Section , we introduce simulation parameters and present performance evaluation with other methods. Finally, Section VI conclude the paper.

## II. THE HOP-DISTANCE RELATIONSHIP

The hop-distance relationship information can effectively improve the performance of the protocol for wireless sensor networks. Usually, the hop-distance relationship in wireless sensor networks closely related to the deployment method, node density and communication radius. Therefore, the deduction process is tedious and impossible to obtain a exact close form by using the geometric methods [10].

From the research during the last decades, the methods for hop statistics can be divided into two categories, statistical analysis and probability deduction. According to the results of Monte Carlo simulation and statistical analysis, Zhao et al [11] points out that the two-dimensional multi hop-distance relationship approximates to the decaying Gaussian distribution. The formula and the least remaining distance (LRD) [12] have derived the single hop-distance equation based on the node distribution and routing strategies. Vural et al. [13] and Dulman et al. [10] propose the probability density function of the multi hop-distance distribution under the one-dimensional condition. In two-dimensional case, the multi hop-distance recursive formulas have been derived in [10, 14-15] with different assumptions. In this study, we focus on a new localization scheme against wormhole attack with the hop-distance relationship under the two dimensional condition.

Without loss of generality, we assume $\mathcal{N}$ sensor nodes have been deployed randomly in circular region $\mathcal{S}_b$ by a homogeneous Poisson process. For what concerns connectivity, we use the Unit Disk Graph (UDG) model [16], i.e. each node is equipped with the omni-directional antenna and any two nodes can communicate if their distance is less than the radio range $R$. Anchors, denoted by $\mathcal{A}$, which have a priori knowledge about their coordinates have the same radio range $R$. We assume there are two end-points of wormhole of which attack range is the circular region with the radius $R_w$, and the length of wormhole link is denoted by $l_w$. As for the range-free localization, the delay of the wormhole link is omitted. When no wormhole attack exists, $\Phi_h(d)$, defined as the probability of the hop count $h$ of two separated nodes by the known distance $d$ is obtained in the equations(1-2) from [15]. We set $\rho_N$ is the node average density, and $\omega(h)$ denotes the scaling factor for each hop $h$ for the multihop dependence elimination, and $\theta$ can be calculated by equation (3).

- When $h = 1$,

$$\Phi_h(d) = \begin{cases} 1 & d < R \\ 0 & d \geq R \end{cases} \tag{1}$$

- When $h \geq 2$ and $R < d \leq hR$,

$$\Phi_h(d) = \left(1 - exp(\omega(h)\int_{x-R}^{x+R} -2\Phi_{h-1}(r)\cdot\rho_N\theta r dr)\right) \times \left(1 - \sum_{i=1}^{h-1}\Phi_i(d)\right) \tag{2}$$

$$\theta = \arccos\frac{d^2 + r^2 - R^2}{2dR} \tag{3}$$

$\omega(h)$ will be estimated by MAD (mean absolute difference ) [15]. We set $R = 25$ and average neighbor nodes $N_R = \pi\rho_N R^2 = 20$. The experimental data are collected from statistical results in 100 simulations, each of which all nodes are re-deployed randomly. Compared with experimental data, the analytical results are shown in Figure 1.

## III. THE LOCALIZATION SCHEME AGAINST WORMHOLE ATTACK

### A. Measurement information from anchors

Since the hop counts among anchors are affected by the wormhole attack in networks, the error should be as small as possible when we determine the measurement information from candidate location. The distance value $d$ can be calculated using the known anchor positions. By the Equation (1-2), the hop counts set $H(d_{ik}) = \{h_1, h_2, \cdots, h_n\}$ and the corresponding probability $PH(d_{ik}) = \{p_{h_1}, p_{h_2}, \cdots, p_{h_n}\}$ can be obtained. In order to reduce the error of hop counts by

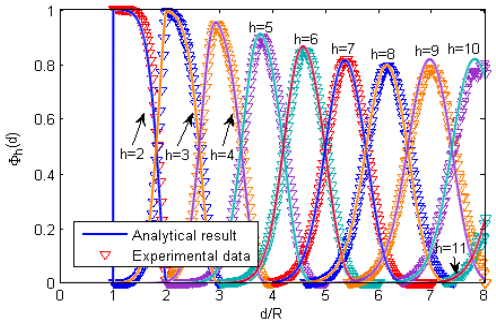



Figure 1. Analytical results and experimental data of $\Phi_h(d)$.

wormhole attack, the estimated hop count $\tilde{h}$ between any two anchors $i$ and $k$ is calculated by (4).

$$\tilde{h} = \frac{1}{\sum_{h_i \in PH(d_{ik})} \frac{p_{h_i}}{h_i}} \tag{4}$$

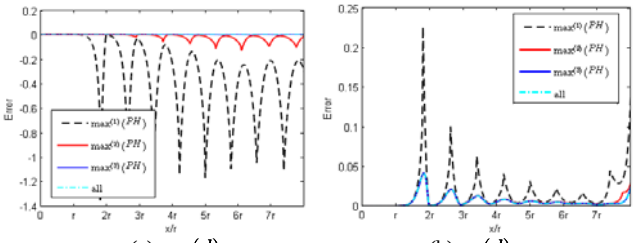
Because the sum of the two largest probabilities in $\Phi_h(d)$ with a known distance can be approximated to 1, we can simplify $\tilde{h}$ by the equation (5). We set $p_{h_{(1)}} = \arg\max_{PH(d)}\{p_{h_i}\}$ and $p_{h_{(2)}} = \arg\max_{PH(d)\setminus\{p_{h_{(1)}}\}}\{p_{h_i}\}$.

$$\tilde{h} = \frac{h_{(1)}h_{(2)}}{p_{h_{(1)}}h_{(2)} + p_{h_{(2)}}h_{(1)}} \tag{5}$$

We set the relative error of hop counts as $\xi_h = \frac{h-\tilde{h}}{h}$, and the mean and the variance of $\xi_h$ is calculated:

$$m_\xi(d) = 1 - \tilde{h} \sum_{h_i \in PH(d)} \frac{\Phi_{h_i}(d)}{h_i} \tag{6}$$

$$v_\xi(d) = \tilde{h}^2 \Big[ \sum_{h_i \in PH(d)} \frac{\Phi_{h_i}(d)}{h_i^2} - \Big( \sum_{h_i \in PH(d)} \frac{\Phi_{h_i}(d)}{h_i} \Big)^2 \Big] \tag{7}$$



(a) $m_\xi(d)$ (b) $v_\xi(d)$

Figure 2. The mean and the variance of $\xi_h$

$m_\xi(d)$ and $v_\xi(d)$ are depicted in Figure 2 with the $\Phi_h(d)$. $max_{(1)}(PH)$ is the method for $\tilde{h}$ calculation with the largest probabilities$p_{h_{(1)}}$, and $max_{(2)}(PH)$ represents the equation (5). $max_{(3)}(PH)$ is the method for $\tilde{h}$ calculation with three largest probabilities and *all* represents the equation (4). From Figure 2(a), it is obvious that $m_\xi(d) \approx 0$ by $max_{(2)}(PH)$ and we set $v_\xi(d) = \sigma^2(d)$.

Each anchor $i$ can calculate the single hop distance to every anchor $k$ with the distance information $d_{ik} = \|\mathbf{x}_i - \mathbf{x}_k\|$ and hop count $h_{ik}$, i.e. $c_{ik} = d_{ik}/h_{ik}$. We sort the $c_{ik}$ as $c_{i(1)} \geq c_{i(2)} \geq \cdots \geq c_{i(n)}$. As is known, when the inequality $\frac{a_1}{b_1} \leq \frac{a_2}{b_2}$ with positive numbers $a_1$, $a_2$, $b_1$ and $b_2$, the inequality $\frac{a_1}{b_1} \leq \frac{a_1+a_2}{b_1+b_2} \leq \frac{a_2}{b_2}$ will hold. We set

$$c_i = \frac{\sum_k \|\mathbf{x}_i - \mathbf{x}_k\|}{\sum_k h_{ik}} \qquad i,k \in \mathcal{A} \text{ and } i \neq k \tag{8}$$

and therefore,

$$\frac{d_{i(n)}}{h_{i(n)}} = c_{i(n)} \leq c_i \leq c_{i(1)} = \frac{d_{i(1)}}{h_{i(1)}} \tag{9}$$

According to the above conclusions, e.g., $m_\xi(d) \approx 0$ and $v_\xi(d) = \sigma^2(d)$, the inequalities $h_{i(n)} \leq \frac{\tilde{h}_{i(n)}}{1-\sigma(d_{i(n)})}$ and $h_{i(0)} \geq \frac{\tilde{h}_{i(1)}}{1+\sigma(d_{i(1)})}$ are hold, therefore the range of the average hop count distance satisfies the following:

$$\overline{c}_i = \frac{d_{i(n)}(1-\sigma(d_{i(n)}))}{\tilde{h}_{i(n)}} \leq c_i \leq \frac{d_{i(1)}(1+\sigma(d_{i(1)}))}{\tilde{h}_{i(1)}} = \underline{c}_i \tag{10}$$

The individual node $j$ receives the information $\overline{c}_i$ and $\underline{c}_i$, and calculates the upper limit $d_M(ji) = \overline{c}_i \times h_{ji}$ and the lower limit $d_m(ji) = \underline{c}_i \times h_{ji}$.

With location reference $\mathbf{x}_i$ of anchor $i$, the maximum distance $d_M(ji)$ and the minimum distance $d_m(ji)$, we know that the node $j$ locates in the ring centered at $\mathbf{x}_i$ with the inner radius $d_m(ji)$ and the outer radius $d_M(ji)$. The candidate position of node $j$ will be located into the ring region $Rg_j(i)$, and the anchor information can be represented by $I_i = \langle \mathbf{x}_i, d_M(ji), d_m(ji) \rangle$.

Let $\Delta_i = d_M(ji) - d_m(ji) = \tau \times d_m(ji)$ and $\tau \in (\frac{1}{6}, \frac{1}{5})$ is hold with $\Phi_h(d)$ in Figure 1, therefore we have $\Delta_i \ll d_m(ji)$.

### *B. Partitioning anchors set by contradiction constrains*

Each node to be localized can collect ring information from many anchors and the overlay area of these rings can be seemed as the estimate location region of the node. However, the overlay area may be error or discontinuous because of the measurement error and the effect on hop count caused by wormhole attack, so that it is hard to correctly estimate the node location by all candidate rings. For this reason, we firstly present a partitioning method for several consistent anchor sets according to the contradiction constrains in geometry. Without the loss of generality, take the node $j$ for an example, the anchors information set of node $j$ can be denoted as $\mathcal{I}_j^{(0)} = \{I_i | i \in \mathcal{A}\}$.

For any two anchors information $I_a, I_b \in \mathcal{I}_j^{(0)}$, if one of the formula (11-13) is satisfied, two rings $Rg_j(a)$ and $Rg_j(b)$ will not overlap each other.

$$\|\mathbf{x}_a - \mathbf{x}_b\| > d_M(ja) + d_M(jb) \tag{11}$$

$$\|\mathbf{x}_a - \mathbf{x}_b\| + d_M(ja) < d_m(jb) \tag{12}$$

$$\|\mathbf{x}_a - \mathbf{x}_b\| + d_M(jb) < d_m(ja) \tag{13}$$

We call the inequality (11-13) as contradiction constrains $\mathcal{L}^c$, and if $\{I_a, I_b\} \nexists \mathcal{L}^c$, $Rg_j(a)$ and $Rg_j(b)$ will have intersection region into which node $j$ may be located. We partition the anchors set $\mathcal{I}_j^{(0)}$ to $\mathcal{I}_j^{(1)}$ with several subsets $Is(k)$ by constraint $\mathcal{L}^c$ which $\mathcal{I}_j^{(1)}$ is:

$$\mathcal{I}_j = \{Is(k) = \{I_a, I_b, \cdots\} |\ \forall I_a, I_b \in Is(k), \{I_a, I_b\} \nexists \mathcal{L}^c\} \tag{14}$$

The partition process is described in Algorithm 1.

**Algorithm 1.** Partition method by contradiction constrains

**Require:** Constrain $\mathcal{L}^c$, the anchors set $\mathcal{A}$ and size $t$

**Ensure:** Output $\mathcal{I}_j = \{Is\}$

1: $S_i[n_i] \leftarrow 1$.
2: **for all** $i = 0$ to $t$ **do**
3: **for all** $j = 0$ to $t$
4: **if** $j \neq i$ **AND** $\{n_j, n_k\} \nexists \mathcal{L}^c$ with all $S_i[n_k] = 1$
5: $S_i[n_j] \leftarrow 1$
6: **end if**
7: **end for**
8: **end for**
9: Remove the repeated sequence of $\{S_i\}_{i=1,\cdots,t}$ to $\{S_i'\}$
10: $Is(i) \leftarrow \{n_k\}$ with all $S_i'[n_k] = 1$

However subset $Is(k)$ in $\mathcal{I}_j^{(1)}$ is not a consistent set. As shown in figure 3, any two of $I_1$, $I_2$ and $I_3$ do not satisfy contradiction constrains $\mathcal{L}^c$, but $Rg_j(1,2)$, $Rg_j(1,3)$ and $Rg_j(2,3)$ will not have the continuous region which means that no consistent location meets simultaneously three anchor conditions. Therefore, we need more constrains to partition anchor sets.

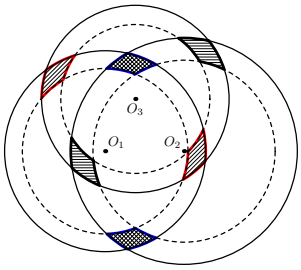



Figure 3. Three rings intersect with each other

### C. Partitioning anchors set by convex constrains

As the above analysis, we will mainly focus on the convex constrains of $\mathcal{I}_j^{(1)}$. Firstly, we would consider the geometric relationship between the candidate rings $Rg_j(a)$ and $Rg_j(b)$ determined by $I_a$ and $I_b$, and there are the following three cases:

(1) The ring center distance $d = \|\mathbf{x}_a - \mathbf{x}_b\|$ between $Rg_j(a)$ and $Rg_j(b)$ satisfies:

$\mathcal{C}_1$: $\max(d_M(ja) + d_m(jb), d_m(ja) + d_M(jb)) < d < d_M(ja) + d_M(jb)$

$\mathcal{C}_2$: $\min(d_M(ja) + d_m(jb), d_m(ja) + d_M(jb)) < d < \max(d_M(ja) + d_m(jb), d_m(ja) + d_M(jb))$,

$\mathcal{C}_3$: $d_m(ja) + d_m(jb) < d < \min(d_M(ja) + d_m(jb), d_m(ja) + d_M(jb))$

(2) $d$ satisfies $\mathcal{C}_4$ and the central angle $\theta_j(a,b)$ corresponding to the bound of the overlay area $Rg_j(a,b)$ of $Rg_j(a)$ and $Rg_j(b)$ cannot be larger than the threshold $\vartheta$

$\mathcal{C}_4$: $\max(|d_M(ja) - d_m(jb)|, |d_M(jb) - d_m(ja)|) < d < d_m(ja) + d_m(jb)$

(3) $d$ satisfies $\mathcal{C}_4$ while $\theta_j(a,b)$ would be larger than the threshold $\vartheta$ and $d$ satisfies $\mathcal{C}_5$.

$\mathcal{C}_5$: $|d_m(ja) - d_m(jb)| < d < \max(|d_M(ja) - d_m(jb)|, |d_M(jb) - d_m(ja)|)$

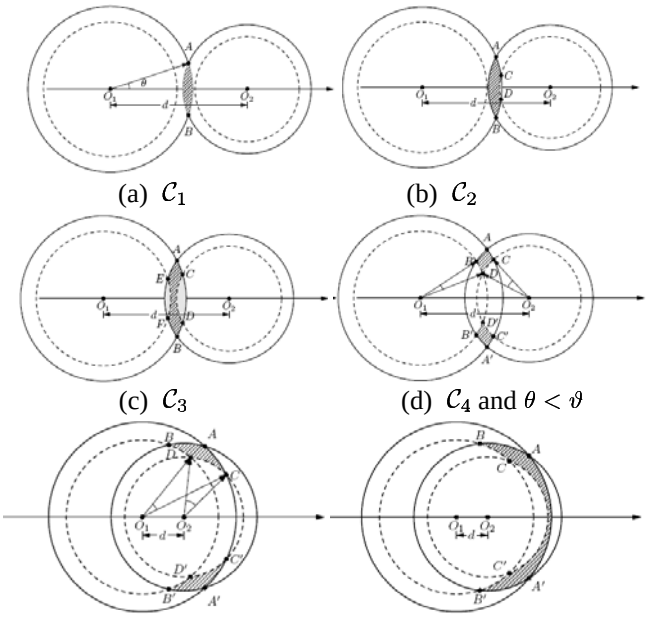



Figure 4. the geometric relationship between $Rg_j(a)$ and $Rg_j(b)$

For case (1), when $d$ satisfies $\mathcal{C}_1$, as shown in Figure 4(a), $Rg_j(a,b)$ is a convex region, and when $d = d_m(ja) + d_m(jb) + \max(\Delta_a, \Delta_b)$, the area and the bound length of the convex region are both maximal. Meanwhile, in order to avoid the convex region of $Rg_j(a,b)$ being too small, $\sigma$ is used and the upper bound of $d$ is reduced to $d_M(ja) + d_M(jb) - \sigma \min(\Delta_a, \Delta_b)$.

When $d$ satisfies $\mathcal{C}_2$, $Rg_j(a,b)$ is a non-convex region as shown in Figure 4(b). The range of $d$ is $\max(d_M(ja) + d_m(jb), d_m(ja) + d_M(jb)) - \min(d_M(ja) + d_m(jb), d_m(ja) + d_M(jb)) = |\Delta_a - \Delta_b| \ll d$. If $|\Delta_a - \Delta_b| \leq \min(\Delta_a, \Delta_b)$, e.g. $0.5 \leq \frac{\Delta_a}{\Delta_b} \leq 2$, the convex region by two circle with $d_M(ja)$ and $d_m(jb)$ will be large enough and $Rg_j(a,b)$ can be regarded as convex region. When $d$ satisfies $\mathcal{C}_3$ shown in Figure 4(c), $Rg_j(a,b)$ is the concave region which will not be considered in this paper. We classify $\mathcal{C}_1$ and $\mathcal{C}_2$ into $\mathcal{C}_1'$ as shown in the formula (15-16), and $Rg_j(a,b)$ will be approximately regarded as the convex region.

$$d_m(ja) + d_m(jb) + \max(\Delta_a, \Delta_b) \leq d \leq d_M(ja) + d_M(jb) - \sigma \min(\Delta_a, \Delta_b) \quad (15)$$

$$d_m(ja) + d_m(jb) + \min(\Delta_a, \Delta_b) \leq d \leq d_M(ja) + d_M(jb) + \max(\Delta_a, \Delta_b) \quad (16)$$

$$s.t. \quad 0.5 \leq \frac{\Delta_a}{\Delta_b} \leq 2$$

We set $\Delta_a = \tau \times d_m(ja)$, $\rho = d_m(jb)/d_m(ja)$ and $\kappa = d/d_m(ja)$. The region boundary of $Rg_j(a,b)$ is two arcs which central angles denote $\theta_a$ and $\theta_b$ in the formula (17).

$$\begin{aligned}\theta_a(\kappa,\rho) &= 2\times\cos^{-1}\left(\frac{\kappa^2+(1+\tau)^2(1-\rho^2)}{2\kappa(1+\tau)}\right)\\ \theta_b(\kappa,\rho) &= 2\times\cos^{-1}\left(\frac{\kappa^2+(\rho^2-1)(1+\tau)^2}{2\kappa(1+\tau)\rho}\right)\end{aligned} \quad (17)$$

When $\rho\in[0.5,2]$ and $\kappa=1+\rho+\tau\times\min(1,\rho)$, $\theta_a$ and $\theta_b$ will reach the maxima, whereas $\rho>2$ or $\rho<0.5$ and $\kappa=1+\rho+\tau\times\max(1,\rho)$, $\theta_a$ and $\theta_b$ can obtain the maximum values. We depict the curve of $\theta_a$ and $\theta_b$ in Figure 5 with a fixed $\tau$, and it is shown that neither $\theta_a$ or $\theta_b$ cannot exceed $\frac{\pi}{2}\approx 1.57$.

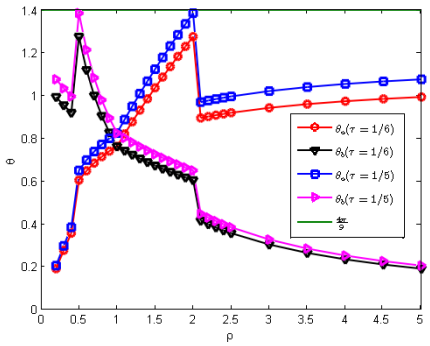



Figure 5. the curve trend of $\theta_a(\kappa,\rho)$ and $\theta_b(\kappa,\rho)$

For case (2), when $d$ satisfies $\mathcal{C}_4$, the $Rg_j(a,b)$ region is shown in Figure 4(d) and Figure 4(e). When central angle $\theta$ is small enough, $\theta-2*\sin(\theta/2)\simeq 0$ and arc length $R\theta$ is close to chord length $2R\sin(\theta/2)$. We set $\vartheta$ for the threshold of central angle. If the central angle $\theta_1=\angle BO_1D<\vartheta$ and $\theta_2=\angle CO_2D<\vartheta$ in Figure 4(d), $Rg_j(a,b)$ will be deemed to the convex region. Similarly, if $\theta_1$ or $\theta_2$ is larger than $\vartheta$ as shown in Figure 4(e), $Rg_j(a,b)$ is a concave region. $\theta_1$ and $\theta_2$ are calculated in the Equation (18-19).

$$\begin{aligned}\theta_1 &= \cos^{-1}\left(\frac{d^2+d_m^2(ja)-d_M^2(jb)}{2d\times d_m(ja)}\right)\\ &\quad -\cos^{-1}\left(\frac{d^2+d_m^2(ja)-d_m^2(jb)}{2d\times d_m(ja)}\right)\end{aligned} \quad (18)$$

$$\begin{aligned}\theta_2 &= \cos^{-1}\left(\frac{d^2+d_m^2(jb)-d_M^2(ja)}{2d\times d_m(jb)}\right)\\ &\quad -\cos^{-1}\left(\frac{d^2+d_m^2(jb)-d_m^2(ja)}{2d\times d_m(jb)}\right)\end{aligned} \quad (19)$$

Due to high complicated computation of $\theta_1$ and $\theta_2$, the Equation (18-19) need to be simplified. We set $\rho=d_m(jb)/d_m(ja)$, $\kappa=d/d_m(ja)$, and $\Delta_a$ and $\Delta_b$ are approximately proportional to $d_m(ja)$ and $d_m(jb)$. Therefore, $\Delta_a=\tau d_m(ja)$ and $\Delta_b=\tau\rho d_m(ja)$. The expression of $\theta_1$ and $\theta_2$ become,

$$\theta_1=\cos^{-1}\left(\frac{\kappa^2+1-\rho^2}{2\kappa}-\frac{\rho^2(2\tau+\tau^2)}{2\kappa}\right)-\cos^{-1}\left(\frac{\kappa^2+1-\rho^2}{2\kappa}\right) \quad (20)$$

$$\theta_2=\cos^{-1}\left(\frac{\kappa^2+\rho^2-1}{2\kappa\rho}-\frac{2\tau+\tau^2}{2\kappa\rho}\right)-\cos^{-1}\left(\frac{\kappa^2+\rho^2-1}{2\kappa\rho}\right) \quad (21)$$

Figure 6 shows the curve of $\theta_1$ and $\theta_2$ varied with $\rho$ and $\kappa$. We can conclude: (1) when $\rho<1$, $\theta_1<\theta_2$ and the range of $\kappa$ for $\theta_2$ is smaller than that for $\theta_1$; (2) when $\rho\geq 1$, $\theta_1>\theta_2$ and the range of $\kappa$ for $\theta_1$ is smaller than that for $\theta_2$; (3) if $\rho$ is small or large enough, either $\theta_1$ or $\theta_2$ may not satisfy the threshold $\vartheta$.

Therefore, we can determine the range of $\kappa$ by the equation (20) when $\rho\geq 1$. We set $z=(\kappa^2+1-\rho^2)/2\kappa$ and $\varepsilon=\rho^2(2\tau+\tau^2)/2\kappa$, it can be found as $\cos^{-1}(z-\varepsilon)\simeq\cos^{-1}(z)+\frac{\varepsilon}{\sqrt{1-z^2}}$, i.e., $\theta_1\simeq\frac{\varepsilon}{\sqrt{1-z^2}}<\vartheta$. In the sense, we have that,

$$(k^2)^2-2(1+\rho^2)k^2+((1-\rho^2)^2+4(\frac{\tau}{\vartheta})^2\rho^2)<0 \quad (22)$$

Notice that $\theta_1<\vartheta$ only when $4\rho^2(1-(\frac{\tau}{\vartheta}\rho)^2)\geq 0$, that is to say, $1\leq\rho\leq\frac{\vartheta}{\tau}$. We have,

$$\sqrt{1+\rho^2-2\rho\sqrt{1-(\frac{\tau}{\vartheta}\rho)^2}}\leq\kappa\leq\sqrt{1+\rho^2+2\rho\sqrt{1-(\frac{\tau}{\vartheta}\rho)^2}} \quad (23)$$

Similarly, when $\rho<1$, we have that,

$$(k^2)^2-2(1+\rho^2)k^2+((\rho^2-1)^2+4(\frac{\tau}{\vartheta})^2)<0 \quad (24)$$

from which we obtain when $\frac{\tau}{\vartheta}\leq\rho<1$,

$$\sqrt{1+\rho^2-2\sqrt{\rho^2-(\frac{\tau}{\vartheta})^2}}\leq\kappa\leq\sqrt{1+\rho^2+2\sqrt{\rho^2-(\frac{\tau}{\vartheta})^2}} \quad (25)$$

We classify the equation (23) and (25) into the constrain $\mathcal{C}_4'$, and it follows that,

$$\begin{aligned}&\sqrt{d_m^2(ja)+d_m^2(jb)-2d_m(jb)\sqrt{d_m^2(ja)-(\frac{\Delta_b}{\vartheta})^2}}\\ \leq d\leq&\sqrt{d_m^2(ja)+d_m^2(jb)+2d_m(jb)\sqrt{d_m^2(ja)-(\frac{\Delta_b}{\vartheta})^2}}\\ &s.t.\quad 1\leq d_m(ja)/d_m(jb)\leq\tfrac{\vartheta}{\tau}\end{aligned} \quad (26)$$

$$\begin{aligned}&\sqrt{d_m^2(ja)+d_m^2(jb)-2d_m(ja)\sqrt{d_m^2(jb)-(\frac{\Delta_a}{\vartheta})^2}}\\ \leq d\leq&\sqrt{d_m^2(ja)+d_m^2(jb)+2d_m(ja)\sqrt{d_m^2(jb)-(\frac{\Delta_a}{\vartheta})^2}}\\ &s.t.\quad \tfrac{\tau}{\vartheta}\leq d_m(ja)/d_m(jb)\leq 1\end{aligned} \quad (27)$$

For case (3), $Rg_j(a,b)$ is the concave region which will not be considered in this paper.

As stated previously, if $I_a$ and $I_b$ satisfy the constrain $\mathcal{C}_1'$ in the equation (15-16), $Rg_j(a,b)$ formed by $I_a$ and $I_b$ is the convex region and two central angles corresponding two boundary arcs are less than $\pi/2$. If $I_a$ and $I_b$ satisfy the constrain $\mathcal{C}_4'$ in the equation (26-27), $Rg_j(a,b)$ will be two disconnected regions called $Rg_j^{(1)}(a,b)$ and $Rg_j^{(2)}(a,b)$. $\mathcal{C}_1'$ and $\mathcal{C}_4'$ are referred to as the convex constrains.

Following that, the geometric relationship would be considered for three non-collinear candidate rings $Rg_j(a)$, $Rg_j(b)$ and $Rg_j(c)$ determined by $I_a$, $I_b$ and $I_c$, any two of which satisfy the convex constrains. There are mainly two following case.

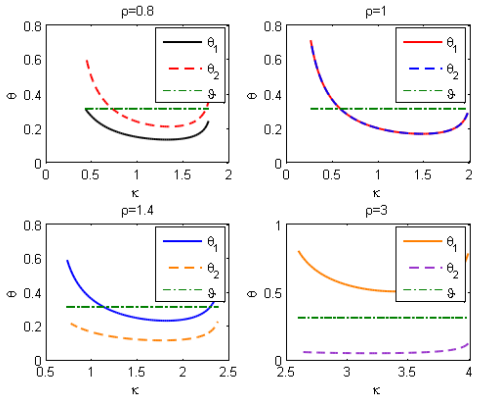



Figure 6. the curve of $\theta_1$ and $\theta_2$ with different $\rho$ and $\kappa$

(1) If the ring center distance $d$ between $Rg_j(a)$ and $Rg_j(b)$ satisfy the constrains $\mathcal{C}_1'$, $Rg_j(a,b)$ will have two points of intersection $\mathbf{x}_p$ and $\mathbf{x}_q$. Let $d_p = \|\mathbf{x}_p - \mathbf{x}_c\|$ and $d_q = \|\mathbf{x}_q - \mathbf{x}_c\|$. If the equations (28-29) do not hold, the region $Rg_j(a,b,c)$ intersected by $Rg_j(a,b)$ and $Rg_j(c)$ will be a convex region.

$$d_p < d_m(jc) \quad \text{and} \quad d_q < d_m(jc) \tag{28}$$

$$d_p > d_M(jc) \quad \text{and} \quad d_q > d_M(jc) \tag{29}$$

(2) If the distance $d$ between $Rg_j(a)$ and $Rg_j(b)$ satisfy the constrains $\mathcal{C}_4'$, two disconnected regions $Rg_j^{(1)}(a,b)$ and $Rg_j^{(2)}(a,b)$ are shown in Figure 4(d). To reduce the computational complexity, $Rg_j^{(1)}(a,b)$ and $Rg_j^{(2)}(a,b)$ are approximated by two circles whose radii are $r_v = \sqrt{\Delta_a^2 + \Delta_b^2}/2$ and centers called $\mathbf{x}_u$ and $\mathbf{x}_v$, are intersection points by two circle with centers of $\mathbf{x}_a$, $\mathbf{x}_b$ and radii of $d_m(ja)+\Delta_a/2$, $d_m(jb)+\Delta_b/2$ respectively. According to the conclusion in [17], if $\mathbf{x}_a$, $\mathbf{x}_b$ and $\mathbf{x}_c$ lie on the same line and the difference $\Delta$ is much less than the inner radius of the ring, the intersection of $Rg_j(a)$, $Rg_j(b)$ and $Rg_j(c)$ only have one continuous region. Let $d_v = \|\mathbf{x}_v - \mathbf{x}_c\|$ and $d_w = \|\mathbf{x}_w - \mathbf{x}_c\|$. If the equations (30-31) do not hold, the region $Rg_j(a,b,c)$ intersected by $Rg_j(a)$, $Rg_j(b)$ and $Rg_j(c)$ will be a convex region.

$$d_m(jc) < d_w - r_v \text{ and } d_m(jc) < d_v - r_v \tag{30}$$

$$d_M(jc) > d_w + r_v \text{ and } d_M(jc) > d_v + r_v \tag{31}$$

We call the constrains above that let $Rg_j(a,b,c)$ be the convex region as $\mathcal{L}^t$. The corresponding anchor set $\mathcal{I}_j^{(2)}$ will be $\mathcal{I}_j^{(2)} = \{It(k) = \{I_a, I_b, I_c, \cdots\} \mid \forall I_a, I_b, I_c \in It(k), \{I_a, I_b, I_c\} \exists \mathcal{L}^t\}$, and the partition algorithm for $\mathcal{I}_j^{(2)}$ is shown in Algorithm 2. The subset $\{It\}$ in $\mathcal{I}_j^{(2)}$ can be accepted as the consistent location references, i.e. the intersection of rings determined by $I_a, I_b, I_c, \cdots$ in $It(k)$ will be a continuous region in geometry.

### D. A filtering strategy for anchor sets

For each subset in $\mathcal{I}_j^{(2)} = \{It(k)\}$, a candidate location $\widetilde{\mathbf{x}}_j(k)$ of the node $j$ can be solved by the least square method. We propose a filtering strategy for these candidate locations to obtain the final estimated location.

For each $\widetilde{\mathbf{x}}_j(k)$ and all anchors information set $\mathcal{I}_j^{(0)}$, we have the distance set $\mathbf{D}(k) = \{d_{k1}, d_{k2}, \cdots, d_{ki}, \cdots\}$ in which $d_{ki} = \|\widetilde{\mathbf{x}}_j(k) - \mathbf{x}_i\|$. With $\mathbf{D}(k)$ and hop counts to all anchors $\mathbf{H} = \{\tilde{h}_1, \tilde{h}_2, \cdots, \tilde{h}_i, \cdots\}$ in equation (5), the probability set $\mathbf{\Phi}(k) = \{\Phi_{\tilde{h}_1}(d_{k1}), \Phi_{\tilde{h}_2}(d_{k2}), \cdots, \Phi_{\tilde{h}_i}(d_{ki}), \cdots\}$ is obtained by hop-distance relationship. We use two parameters $P_m(k)$ and $C(k)$ to estimate the final location of node $j$:

**Algorithm 2.** Partition method by convex constrains

**Require:** Constrains $\mathcal{C}_1'$、$\mathcal{C}_4'$ and $\mathcal{L}^t$, the anchors set $\mathcal{A}$

**Ensure:** Output $\mathcal{I}_j^{(2)} = \{It\}$

1: Obtain the set $\mathcal{I}_j^{(1')}$ of size $t'$ with anchors set $\mathcal{A}$ and constrains $\mathcal{C}_1'$、$\mathcal{C}_4'$ by **Algorithm** 1.
2: $\mathcal{I}_j^{(2)} \leftarrow \emptyset$
3: **for all** $i = 0$ to $t'$ **do**
4: **repeat**
5: Select any pair $n_a$ and $n_b$ from $Is(i)$ in $\mathcal{I}_j^{(1')}$, and obtain $Ir = \{n_j\}$ which $\forall n_j \in Ir$, $\{n_a, n_b, n_j\} \exists \mathcal{L}^t$
6: Obtain the $\{It(l)\}$ with $Ir$ and constrain $\mathcal{L}^t$ with $n_a$ and $n_b$ by **Algorithm** 1.
7: $It(l) \leftarrow It(l) \cup \{n_a, n_b\}$
8: $\mathcal{I}_j^{(2)} \leftarrow \mathcal{I}_j^{(2)} \cup \{It(l)\}$
9: **until** complete $Is(i)$ traversal
10: **end for**

(1) $P_m(k)$ is the mean of the probability $\mathbf{\Phi}(k)$:

$$P_m(k) = \frac{1}{n} \sum_{i=1}^{n} \Phi_{h_i}(d_{ki}) \tag{32}$$

(2) $C(k)$ is the number of $\Phi_{\tilde{h}_i}(d_{ki})$ smaller than the threshold $\Phi_0$ in $\mathbf{\Phi}(k)$:

$$C(k) = \frac{1}{n} \sum_{i=1}^{n} (\Phi_{h_i}(d_{ki}) < \Phi_0) \tag{33}$$

Due to the presence of wormhole attack, the hop counts set $\mathbf{H}$ may not be true. Therefore, we use two parameters to estimate the location of node $j$ in the equation (34). A candidate location set with $\min\{C(k)\}+1$is filtered, and the final estimated location $\widetilde{\mathbf{x}}_j(k_0)$ is selected by the maximum of $P_m(k)$.

$$\begin{aligned} k_0 \quad = \quad & \arg_{k^*} \max P_m(k^*) \\ & \text{s.t.} \quad \forall k^* \quad C(k^*) \le \min\{C(k)\} + 1 \end{aligned} \tag{34}$$

## IV. PERFORMACE EVALUATION

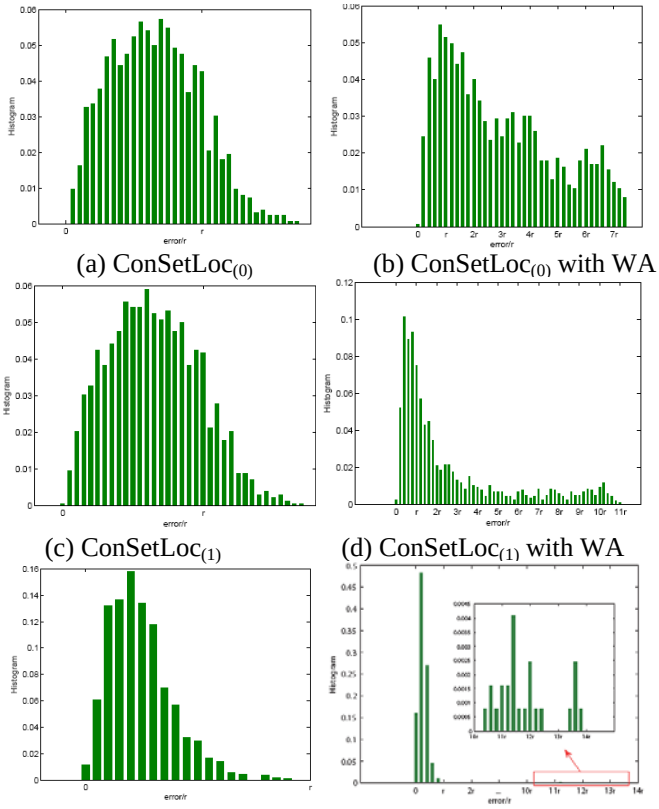
The section presents several simulations for performance evaluation of the ConSetLoc under wormhole attack. In our simulation setting, sensor nodes $N = 1280$ are deployed randomly in a circular region with the radius $R_b = 200$ by a planar Poisson process. We vary the anchor fraction $\rho_a$, and the average local neighborhood $N_R$ to performance comparisons with MMSE (Minimum Mean Square Estimate)[18], PDM (Proximity-Distance Map)[19] and LMS (Least Median of Squares)[20]. Each experiment has been carried out for 50 trials with different random seeds. And the average localization error $\bar{\xi} = \frac{1}{N} \sum_i e_i$ is calculated.

### A. Comparisons of localization error distribution

Let ConSetLoc$_{(0)}$ be the localization scheme with the anchor set $\mathcal{I}_j^{(0)}$ and ConSetLoc$_{(1)}$ be the scheme with the anchor set $\mathcal{I}_j^{(1)}$. In our experiment parameters, we set anchor fraction $\rho_a = 5\%$, the radio range $R = 25$ and

the length of wormhole link $l_w = 11R$. The simulation results of the histogram for localization error $e_j = \|\mathbf{x}_j - \widetilde{\mathbf{x}}_j\|$ are shown in Figure 7 with and without wormhole attack.

When no wormhole attack exists, the histograms for localization error $e_j$ of ConSetLoc$_{(0)}$ and ConSetLoc$_{(1)}$ are comparable, and $e_j$ are within $(0, 2R)$ shown in Figure 7(a) and 7(c). By contrast, all localization errors of ConSetLoc are less than $R$ in Figure 7(e). This is because less contradiction constrains for $\mathcal{I}_j^{(0)}$ while the ConSetLoc algorithm can filter out all inconsistent location references of $\mathcal{I}_j^{(0)}$.



(a) ConSetLoc$_{(0)}$ (b) ConSetLoc$_{(0)}$ with WA

(c) ConSetLoc$_{(1)}$ (d) ConSetLoc$_{(1)}$ with WA

(e) ConSetLoc (f) ConSetLoc with WA

Figure 7 the histograms of localization error without and with wormhole attack (WA)

When there is wormhole attack in the network, ConSetLoc$_{(1)}$ in Figure 7(d) has higher proportion of sensor nodes with lower localization errors compared with histograms in Figure 7(b). However, ConSetLoc has the least impact on wormhole attack and the majority of sensor nodes' estimated locations lie with $R$ of the true locations. There are only $2\%$ sensor nodes that have huge localization error, and the measured information indicates that these nodes locate nearby one wormhole and are estimated incorrectly to another wormhole because of majority of location references.

### B. *Effect of the anchor fraction $\rho_a$*

In this experiment, we investigate the effect of the anchor fraction on the performance of ConSetLoc compared with MMSE, LMS and PDM.

We vary $\rho_a$ from 2% to 10%. Figure 8 depicts the average localization error $\overline{\xi}$. From the figure, PDM performs a litter better than MMSE and LMS, while ConSetLoc has the smallest localization error. As the anchor fraction increases, the estimated geographic distances increase, but more false distances may be used to position estimation. When the anchor fraction increases over 4%, the performance of the four methods will nearly remain unchanged. ConSetLoc has better filtering ability with more consistent sets than those methods for wormhole attack-resistant.

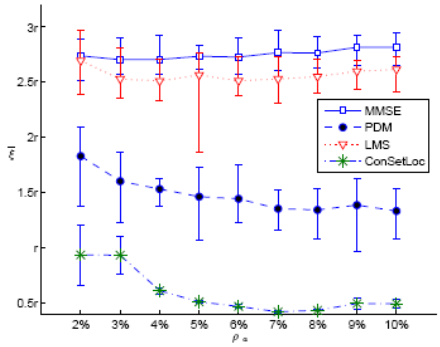



Figure 8. Effect of anchor fraction for localization error $\overline{\xi}$

### C. *Effect of the average local neighborhood $N_R$*

In this experiment, the average local neighborhood varies from 12.8 to 23.2 with the corresponding adjustment of radio range $R$. The results of MMSE, PDM, LMS and ConSetLoc are shown in Figure 9. As the average local neighborhood increases, the network provides stronger connectivity and less impact on hop counts. We can see that the localization error of MMSE and LMS reduce obviously, whereas PDM and ConSetLoc remain almost unchanged. PDM is closely related to the network topology and more volatile than ConSetLoc for the average localization error. For the average localization error, ConSetLoc outperforms the other three localization methods.

## V. CONCLUSION

Secure localization against wormhole attack for wireless sensor network is challenging not only because wormhole attack is immune to cryptographic techniques but special hardware devices to detection present additional overhead. In this paper, we present a robust range-free localization algorithm ConSetLoc without coming up with detection solution for wormhole attack. With the hop-distance relationship, we utilize convex constrains in geometry to partition consistent anchors set and propose a filtering strategy for sensors' candidate location in the ConSetLoc algorithm. Simulation results demonstrate that the ConSetLoc algorithm can provide good localization precision for majority of sensor nodes and performs better than other methods in literature.

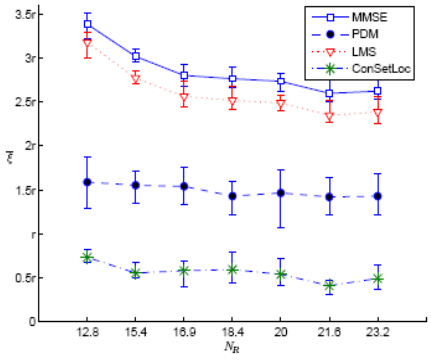



Figure 9. Effect of average local neighborhood $N_R$

## REFERENCES


[1] D. Niculescu, Positioning in Ad Hoc Sensor Networks, IEEE Networks, pp. 24 – 29, July/August 2004

[2] Lazos L and Poovendran R. SeRLoc: Secure range-independent localization for wireless sensor networks. In Proc. of the 2004 ACM Workshop on Wireless Security, pp. 21-30, 2004, ACM Press.

[3] K. Wu et al. Robust range-free localization in wireless sensor networks. Mob. Netw. Appl. 2007, vol. 12(5), pp. 392–405.

[4] Hu, Y. C et al. Packet leashes: a defense against wormhole attacks in wireless networks. INFOCOM, Vol.3:1976-1986, 2003

[5] Tian He, et al., Range-free localization schemes for large scale sensor networks. In MobiCom '03: Proceedings of the 9th annual international conference on Mobile computing and networking, pp. 81-95, 2003. ACM Press.

[6] Sheu JP et al., A distributed localization scheme for wireless sensor networks with improved grid-scan and vector-based refinement. IEEE transactions on mobile computing, 7(9):1110-1123, Sep 2008.

[7] D. Niculescu and B. Nath. Dv based positioning in ad hoc networks. Telecommunication Systems, 22(14):267-280, January 2003.

[8] R. Maheshwari et al. Detecting wormhole attacks in wireless networks using connectivity information. In IEEE Conference on Computer Communications INFOCOM, pp. 107-115, 2007.

[9] Yurong Xu, Yi Ouyang, et. al., Analysis of Range-Free Anchor-Free Localization in a WSN under Wormhole Attack, Proceedings of the 10th ACM/IEEE International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM'07), pp. 344-351, Chania, Crete Island, Greece, Oct 22-26, 2007.

[10] Stefan Dulman, et al. On the hop count statistics for randomly deployed wireless sensor networks. Int. J. Sen. Netw., vol.1(1/2):89-102, 2006.

[11] Liang Zhao and Qilian Liang. Hop-distance estimation in wireless sensor networks with applications to resources allocation. EURASIP Journal on Wireless Communications and Networkin, 2007:8

[12] Swades De, et al. Bounds on hop distance in greedy routing approach in wireless ad hoc networks. Int. J. Wire. Mob. Comput., vol.1(2):131-140, 2006.

[13] Serdar Vural and Eylem Ekici. Probability distribution of multi-hop-distance in one-dimensional sensor networks. In Comput. Netw., vol.51(13), pp. 3727-3749, 2007.

[14] E. Ekici, et al. A probabilistic approach to location verification in wireless sensor networks. In Communications, 2006. ICC '06. IEEE International Conference on, vol. 8, pp. 3485-3490, 2006.

[15] Xiaoyuan Ta, et al. Evaluation of the probability of k-hop connection in homogeneous wireless sensor networks. In Global Telecommunications Conference, 2007. GLOBECOM'07. IEEE, pp. 1279-1284, 2007.

[16] Clark, B.N. et al., Unit Disk Graphs. Discrete Mathematics, Vol. 86:165-177, 1991.

[17] Zhong S, et al. Towards a theory of robust localization against malicious beacon nodes. INFOCOM, Vol. 1-5:2065-2073, 2008.

[18] Savvides, et al. Dynamic fine-grained localization in ad-hoc networks of sensors. In Proc. 7th Ann. Intl. Conf. on Mobile Computing and Networking, pages 166-179, 2001.

[19] Hyuk Lim and J. C. Hou. Localization for anisotropic sensor networks. Infocom'05, vol. 1, pp. 138-149, 2005

[20] Li Z et al. Robust statistical methods for securing wireless localization in sensor networks. In Proc. of the Int'l Symp. on Information Processing in Sensor Networks, Washington: IEEE Computer Society Press, pp.91-98, 2005



**Niu Yanchao**, received the B.S. degree in electronic and information engineering in 2004 from Dalian University of Technology. He is Ph.D. student in the National Engineering Laboratory for Next Generation Internet Interconnection Devices at Beijing Jiaotong University. His current research interests include wireless sensor network and localization protocol.

**Gao Deyun,** received the B.S. and M.S. degree in electrical engineering and the Ph.D. degree in computer science from Tianjin University, Tianjin, China, in 1994, 1999, and 2002, respectively. He spent one year as a Research Associate with the Department of Electrical and Electronic Engineering, Hong Kong University of Science and Technology, Kowloon, Hong Kong. He then spent three years as a Research Fellow with the School of Computer Engineering, Nanyang Technological University, Singapore. Since 2007, he has been with Beijing Jiaotong University, Beijing, China, as an Associate Professor with the School of Electronics and Information Engineering. His research interests include wireless sensor network, mobile Internet, next-generation networks.

**Shuai Gao** received the BS degree in 2001 and MS degree in 2004 in communication and information system from Beijing Jiaotong University. He is currently a lecture in School of Electronic and Information Engineering at Beijing Jiaotong University. His current research interests include mobility in wireless sensor networks, Internet routing.

**Chen Ping** received her B.S. and Ph.D degree in computer science form Beijing Jiaotong University, in 2003 and 2009, respectively. She is now a lecture in Nankai University. Her research interests include network optimization, metaheuristics, etc.

# A Water Quality Monitoring Method Based on Fuzzy Comprehensive Evaluation in Wireless Sensor Networks

Jian Shu[1,2], Ming Hong[1,3]
[1]Internet of Things Technology Institute, Nanchang Hang Kong University
[2] School of Software, Nanchang Hang Kong University
Email: {shujian@jxjt.gov.cn, hong19860320@hotmail.com}

Linlan Liu[1,3] and Yebin Chen[1,2]
[3] School of Information Engineering, Nanchang Hang Kong University
Email: {linda_cn68@yahoo.com, chenhb46@163.com}

***Abstract*—A novel water quality monitoring method named WQMMFCE is proposed based on fuzzy comprehensive evaluation model to solve the issue of high energy consumption caused by centralized approach under the background of water quality monitoring. The weights of all factors which are necessary for fuzzy comprehensive evaluation can be obtained firstly by using the binary expert evaluation. The information needs to transmit is decided according to water quality grade which is quantitatively analyzed directly by using fuzzy comprehensive evaluation rather than transmitting large amount of raw data to the monitoring center. Simulation result shows that the method has a great advantage over the centralized one on energy saving and can prolong the lifetime of the network.**

***Index Terms*—wireless sensor networks, fuzzy comprehensive evaluation, water quality monitoring**

## I. INTRODUCTION

Poyang Lake is the largest freshwater lake in china, and it also is one of freshwater lakes whose water quality is best. The data from water environmental monitoring department shows that 97 percent water from the lake whose quality is better than grade III in 2002. While it dropped to 82.1% in 2006 and 44.4% in the first half year of 2009. The proportion decreased year by year [1]. Furthermore, in recent years, several cyanobacteria crisis are broken out in the third largest freshwater lake - Tai Lake. The seriously deteriorated water has threatened the safety of the drinking water. At present, the regular fixed-point sampling measurement is used for monitoring water quality in most area of china, but it can't comprehensively reflect water quality and other environmental factors in time. Especially it can't track the sudden water pollution, air pollution and other serious incidents dynamically and timely.

This paper is sponsored by the National Nature Science Foundation of China(No.60773055),Jiangxi Nature Science Foundation (No.2009GZS0089), and Jiangxi Key Technology R&D Program(No.2009BGA01000).


doi:10.4304/jnw.7.1.195-202

Wireless Sensor Network (WSN) is a wireless network system which is constituted by a large number of micro, cheap and low power sensors nodes deployed in the environment monitored area by means of wireless multi-hop (Ad Hoc) [2]. The monitored information is sensed and processed by the node simply. Then the data is sent to the Sink node through multi-hop. The Sink node sends the data to the monitoring center through satellite channels, internet or mobile communication network, etc. Therefore, WSN can comprehensively monitor air, water, soil and even bird migration habits in lake area in time.

Domestic and overseas research institutions have begun to research the application of wireless sensor networks in remote monitoring. Some well known applications are "GreenOrbs" [3], "FireBug system" [4], and "location and emergency linkage system for mines" etc. They are applied in forest ecological monitoring, forest fire fighting, and coal mine safety monitoring respectively. At present, the traditional centralized data processing method is used in most remote monitoring systems. However, the centralized method has the problem of large energy consumption. According to the characteristic of water monitoring, this paper proposes a water quality monitoring method based on fuzzy comprehensive evaluation in wireless sensor networks (WQMMFCE) to alleviate the problem of large energy consumption cased by the centralized method.

## II. SYSTEM MODE

Under the background of water quality monitoring, a large number of nodes are deployed in multiple monitored areas. The nodes in each monitored area form a wireless network. Each network has one sink node, some high-power relay nodes, and some sensor nodes. Sink takes responsible for transmitting the collected data to the monitoring center. High-power relay nodes take responsible for forwarding the data from cluster heads to sink node. Sensor nodes collect data periodically. The monitoring Center is constituted by software and

hardware device. The architecture of the system is shown in Fig. 1.

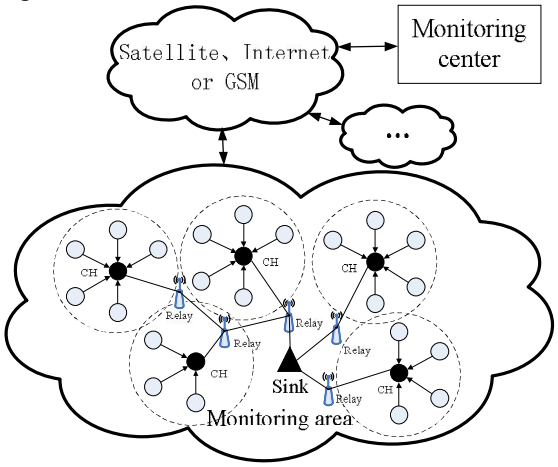



Figure 1. The architecture of system.

- The power supply module of sink is constituted by solar panel and rechargeable lithium battery. The node uses solar for power supply and charging the battery during day time. While at night, it uses rechargeable lithium battery for power supply. The computing ability of Sink node is stronger than other nodes, and it uses GSM module to transmit the collected data to the monitoring center.
- The same as Sink node, the high-power relay node uses solar and rechargeable lithium battery for power supply. Compared with sensor node, the high-power relay node adds the power amplifier circuit which is opened only when the data is forwarding and closed immediately after transmission. Its transmission distance can be up to 1 ~ 2km.
- Sensor node uses two 1.5V batteries for power supply. TIMSP430 chip is used as processor and Chipcom's CC2420 chip is used for wireless transceiver module. The sensor node is equipped with several sensors. TinyOS2.0 operating system is used. Its transmission distance can be up to 60~100m. During the network initialization, the network forms the cluster by using an improved clustering algorithm. During run time, the work mode that sleeping, collecting and processing sensory data periodically is adopted by the cluster member node.
- Hardware devices in monitoring center include server, gateway node and client. Gateway node is equipped with GSM module whose responsibility is to collection data comes from Sink nodes deployed in each monitoring area, and transmit the data to the server through the serial port. The server processes and stores the received packet. The client accesses the data stored in the server through the Internet for further processing.
- Software in the monitoring center includes server and client software. Server software is used for collecting and analyzing the data from each monitoring area. It also take responsible for forwarding the data to the client through the Internet by VPN tunnel once the client asked for the data. Client software is used for receiving data from the server and showing water quality distribution map of the entire monitoring area by combining the GIS information.

WQMMFCE is designed based on the following assumptions:

- All nodes are homogeneous and the energy is limited. The number and type of sensors in each node are the same. There are dissolved oxygen (DO), chemical oxygen demand (COD) and ammonia ($NH_3$-N) [5].
- All the nodes are located and fixed, that is, the monitoring center has the location information of each node, such as longitude, latitude and altitude.
- Cluster-based network topology is used.

## III. Fuzzy Comprehensive Evaluation

Fuzzy comprehensive evaluation is an overall evaluation which is used for evaluating object or environment affected by variety of factors. Its most significant feature is that it can handle fussy factors well. Comprehensive evaluation can solve the problem that quantitative analysis is hard to be implemented by analytical method due to the effects of various complex and uncertain factors [6]. The concrete step of this method is described as follows:

### *A. Establishing factor set, weight set and judgment set*

- Factor set: U= $\{u_1, u_2, \ldots, u_n\}$ is a set of various properties which will affect final evaluation. In this paper, it is the set of the necessary sensor types carried by each node: U={dissolved oxygen(DO), chemical oxygen demand (COD), ammonia ($NH_3$-N)}.
- Weight set: W= $\{w_1, w_2, \ldots, w_i, \ldots, w_n\}$. Where $w_i$ is the weight of factor i.
- Judgment set: V= $\{v_1, v_2, \ldots, v_n\}$ is a set of all possible evaluation results for the evaluation objects. The purpose of fuzzy comprehensive evaluation method is to select the best evaluation result from the judgment set based on considering all the factors. According to GB3838-2002[7], the water quality is divided into five levels: I, II, III, IV and V. the best level is Grade I. Therefore, in this paper, V= {I, II, III, IV, V}.

### *B. Determining the weight of different factors*

The grade of water quality is evaluated by considering various factors which have different effects. So the weight of each factor should be considered. This paper will use expert evaluation and the binary comparison method to determine the weights.

Expert evaluation [8] is a common used method to determine the weights of each factor. The method firstly distributes the factors to the experts in the form of table. Then, each expert scores the factor based on considering the importance of each factor according to the knowledge

and experience. Finally, the weight is decided by calculating average value. The method collects wisdom of all the experts, but it is difficult to compare all factors at the same time for any expert.

Binary comparison method [9] is known as paired comparison or intercomparison. The method make paired comparison of all the factors. Then the relation among the factors is determined by superposition calculation.

This paper combines the advantages of two methods and proposes a binary expert evaluation to solve the problem that it is hard to assign the weights for each factor. The steps of the method are described as follows:

Step 1: Determining experts, and distributing the questionnaire of the comparison of main factors $x_i$ (i=1, 2, 3,…, n) to all experts. Then the forms are taken back. The format of questionnaire is shown in Table I.

TABLE I.
THE FORMAT OF QUESTIONNAIRE

| Factors | Factors | | | |
|---|---|---|---|---|
| | $x_1$ | $x_2$ | … | $x_n$ |
| $x_1$ | $a_{11}=1$ | $a_{12}$ | … | $a_{1n}$ |
| $x_2$ | $a_{21}$ | $a_{22}=1$ | … | $a_{2n}$ |
| … | … | … | … | … |
| $x_n$ | $a_{n1}$ | $a_{n2}$ | … | $a_{nn}=1$ |

Where $a_{ij}$ is the comparison scale of factor i and factor j. The value of $a_{ij}$ can be an integer ranged from 1 to 9.

Factor i is the same important as factor j if $a_{ij}$ equals 1. The larger the value of $a_{ij}$ is, the more important is factor i compared with factor j.

Step 2: The final scale matrix $A_{nxn}$ is obtained by averaging the scale from each expert, namely $\overline{a}_{ij}=\frac{1}{m}\sum_{k=1}^{m}a_{ij}^{k}$ (k=1, 2…m; k represents different experts). $A_{nxn}$ is shown in formula(1).

$$A_{n\times n}=\begin{bmatrix}\overline{a}_{11} & \overline{a}_{12} & \cdots & \overline{a}_{1n}\\ \overline{a}_{21} & \overline{a}_{22} & \cdots & \overline{a}_{2n}\\ \cdots & \cdots & \cdots & \cdots\\ \overline{a}_{n1} & \overline{a}_{n2} & \cdots & \overline{a}_{nn}\end{bmatrix} \quad (1)$$

Step 3: The weight vector $W=\{w_1,w_2,...,w_n\}$ is calculated by geometric averaging the row vector of scale matrix $A_{nxn}$. The calculation formula is shown in formula (2):

$$W_i=\frac{\sqrt[n]{\prod_{j=1}^{n}\overline{a}_{ij}}}{\sum_{i=1}^{n}\sqrt[n]{\prod_{j=1}^{n}\overline{a}_{ij}}} \quad (2)$$

### C. Establishing membership function and determining the membership matrix

In water quality monitoring, membership is used for describing the fuzzy boundaries of the water quality grade. It can overcome the disadvantage of discontinuity of water quality grade so as to make the evaluation result more reasonable. Lower semi-trapezoidal function whose calculation is simple is adopted to describe the affiliation between each factor and each water quality grade due to the limited computing capability of the node. The functions are shown in formula (3), (4), and (5):

- Membership function under grade I (j=1) is shown in formula (3):

$$u_{ij}=\begin{cases}1 & c_i\le s_{ij}\\ \dfrac{s_{i,j+1}-c_i}{s_{i,j+1}-s_{ij}} & s_{ij}<c_i<s_{i,j+1}\\ 0 & c_i\ge s_{i,j+1}\end{cases} \quad (3)$$

- Membership function under grade II (j=2), III (j=3), and IV (j=4) is shown in formula (4):

$$u_{ij}=\begin{cases}1 & c_i=s_{ij}\\ \dfrac{c_i-s_{i,j-1}}{s_{ij}-s_{i,j-1}} & s_{i,j-1}<c_i<s_{ij}\\ \dfrac{s_{i,j+1}-c_i}{s_{i,j+1}-s_{ij}} & s_{ij}<c_i<s_{i,j+1}\\ 0 & c_i>s_{i,j+1}\end{cases} \quad (4)$$

- Membership function under grade V (j=5) is shown in formula (5):

$$u_{ij}=\begin{cases}1 & c_i\ge s_{ij}\\ \dfrac{c_i-s_{i,j-1}}{s_{ij}-s_{i,j-1}} & s_{i,j-1}<c_i<s_{ij}\\ 0 & c_i\le s_{i,j-1}\end{cases} \quad (5)$$

Where, $u_{ij}$ is the membership value between factor i (i=1,2,3,…,n) and the water quality grade j (j=1,2,3,4,5). $C_i$ is the current observed value of factor i. $S_{ij}$ represents the standard value of factor i when the water quality grade is j. The standard values can be found in GB3838-2002. Thus, the membership matrix as shown in formula (6) can be obtained.

$$R_{n\times 5}=\begin{bmatrix}u_{11} & u_{12} & u_{13} & u_{14} & u_{15}\\ u_{21} & u_{22} & u_{23} & u_{24} & u_{25}\\ \cdots & \cdots & \cdots & \cdots & \cdots\\ u_{n1} & u_{n2} & u_{n3} & u_{n4} & u_{n5}\end{bmatrix} \quad (6)$$

### D. Determination of fuzzy synthesize judgement model

The water quality is evaluated comprehensively through the main factor highlighted decision model $M(\bullet,\vee)$ [6] after the membership matrix $R_{nx5}$ is obtained. The formula as shown in formula (7) can be obtained by the model.

$$\underset{\sim 1\times 5}{B}=W_{1\times n}\bullet R_{n\times 5} \quad (7)$$

Where, $b_j=\bigvee_{i=1}^{n}w_i\bullet u_{ij}(j=1,2,3,4,5)$ is the element of comprehensive evaluation vector $\underset{\sim 1\times 5}{B}$, $w_i$ is the i-th element of weight vector W, while $u_{ij}$ is the element of membership matrix $R_{nx5}$.

The water quality grade can be obtained according to the principle of maximum membership after comprehensive evaluation vector $B_{\sim 1\times 5}$ is obtained. For example, if $b_j = \max(b_1, b_2, b_3, b_4, b_5)$, then the water quality grade is determined to be j. When the evaluation vector obtained by calculation is $B_{\sim 1\times 5} = \{0.35143, 0.14857, 0.39000, 0.01, 0\}$, then, $b_3$ equals 0.39000 which is the biggest value in $B_{\sim 1\times 5}$, so the water grade of this region should be determined as III.

## IV. A Water Quality Monitoring Method Based on Fuzzy Comprehensive Evaluation in Wireless Sensor Networks

In this paper, wireless sensor network will be used for water quality monitoring in lake. If the traditional centralized approach is used, the monitored data is sent to the monitoring center after simple processing. For practical applications, more factors need to be monitored, so, large amount of data is need to be transmitted, which will increase the possibility of collision, reduce quality of service, and shorten the lifetime of the entire network caused by high energy consumption.

In this paper, fuzzy comprehensive evaluation is adopted to calculate the water quality. The sensor node can calculate the water quality using fuzzy comprehensive evaluation after sensing and collecting the data. Whether the pollution information should be transmitted or not is determined by the grade of water quality. The processed data is transmitted to cluster head. Then the data is compressed and transmitted to the sink by the cluster head. Finally monitoring center obtains the data.

### *A. WQMMFCE processing*

The main steps of WQMMFCE are described as follows:

Step 1: The network is initialized to clusters. In this paper, tri-color method which is mentioned in paper [10] is used for generating clustered network topology. It is one of TopDisc algorithms. There is only one hop from cluster member nodes to cluster head to avoid the existence of multi-hop in a single cluster. In the tri-color method, nodes have three states which are represented by white, black and gray. White represents undiscovered node, black represents cluster head node, and gray represent cluster member nodes. All nodes are marked as white before clustering. Then Sink node sends the network clustering message and marks itself black. White nodes change themselves into gray once they received the message from a black node and wait for a certain time period to send the network clustering message, the waiting time is inversely proportional to the distance from itself to the black node. When a white node receives the message from a gray node, it will wait for a period of time firstly, the length of waiting time is inversely proportional to the distance between itself and the gray node. As it's very difficult to obtain the exact distance between the nodes in the wireless sensor network, clustering effect will be not satisfactory. So, RSS value is used to replace the actual distance which is mentioned in literature [11]. In the method, it is assumed that each node has a global ID. The waiting time is calculated by formula (8).

$$T_w = a \cdot \sqrt[b]{LOSS} + c \cdot \log_{10}(ID) \quad (8)$$

Where $T_w$ is the waiting time, LOSS represents the received signal strength (RSS), a, b, and c are experience control parameters which can be determined by many experiments.

Step 2: Cluster member nodes collect the data, such as dissolved oxygen (DO), chemical oxygen demand (COD) and ammonia ($NH_3$-N), etc. Supposed that the data obtained from node 1 and node 2 in a period is shown in table II.

TABLE II.
THE FACTOR VALUES OF NODE 1 AND NODE 2 COLLECTED IN A PERIOD

| Nodes | Factors | | |
|---|---|---|---|
| | DO | COD | $NH_3$-N |
| 1 | 8.83 | 6.10 | 0.41 |
| 2 | 7.15 | 10.50 | 1.52 |

Step 3: Cluster member nodes calculate the membership value $u_{ij}$ after the data is collected, then the membership matrix $R_{nx5}$ is gotten.

The membership function for each factor can be obtained according to the standard value under different water quality grades in GB3838-2002. For example, the standard values of $NH_3$-N under grade I to grade V are 0.15, 0.5, 1.0, 1.5, and 2.0 respectively.

The membership functions for $NH_3$-N are described as follows:

- $NH_3$-N's membership function under grade I (j=1) is shown in formula (9):

$$u_{i1} = \begin{cases} 1 & c_i \le 0.15 \\ \dfrac{0.5 - c_i}{0.5 - 0.15} & 0.15 < c_i < 0.5 \\ 0 & c_i \ge 0.5 \end{cases} \quad (9)$$

- $NH_3$-N's membership functions under grade II (j=2), III (j=3), and IV (j=4) are shown in formula (10),(11) and (12) respectively:

$$u_{i2} = \begin{cases} 1 & c_i = 0.5 \\ \dfrac{c_i - 0.15}{0.5 - 0.15} & 0.15 < c_i < 0.5 \\ \dfrac{1.0 - c_i}{1.0 - 0.5} & 0.5 < c_i < 1.0 \\ 0 & c_i > 1.0 \end{cases} \quad (10)$$

$$u_{i3}=\begin{cases}1 & c_i=1.0\\ \dfrac{c_i-0.5}{1.0-0.5} & 0.5<c_i<1.0\\ \dfrac{1.5-c_i}{1.5-1.0} & 1.0<c_i<1.5\\ 0 & c_i>1.5\end{cases} \quad (11)$$

$$u_{i4}=\begin{cases}1 & c_i=1.5\\ \dfrac{c_i-1.0}{1.5-1.0} & 1.0<c_i<1.5\\ \dfrac{2.0-c_i}{2.0-1.5} & 1.5<c_i<2.0\\ 0 & c_i>2.0\end{cases} \quad (12)$$

- $NH_3$-N's membership function under grade V (j=5) is shown in formula (13):

$$u_{i5}=\begin{cases}1 & c_i\geq 2.0\\ \dfrac{c_i-1.5}{2.0-1.5} & 1.5<c_i<2.0\\ 0 & c_i\leq 1.5\end{cases} \quad (13)$$

Memberships of $NH_3$-N for node 1 and node 2 under each water quality grade are calculated by formulas (9)~(13), and they are $u_3$=(0.25714,0.74286,0,0,0) and $u_3$'=(0,0,0,0.96,0.04). Similarly, memberships of DO for two nodes under each water quality grade are $u_1$ = (1, 0, 0, 0, 0) and $u_1$'= (0.76667, 0.23333, 0, 0, 0). Memberships of COD under each water quality grade are $u_2$ = (0, 0, 0.975, 0.025, 0) and $u_2$'= (0, 0, 0, 0, 1). Then the membership matrixes of the two nodes are obtained and shown in formula (14) and (15).

$$R_{3\times5}=\begin{bmatrix}1 & 0 & 0 & 0 & 0\\ 0 & 0 & 0.975 & 0.025 & 0\\ 0.25714 & 0.74286 & 0 & 0 & 0\end{bmatrix} \quad (14)$$

$$R_{3\times5}'=\begin{bmatrix}0.76667 & 0.23333 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 1\\ 0 & 0 & 0 & 0.96 & 0.04\end{bmatrix} \quad (15)$$

Step 4: Cluster member nodes calculate the comprehensive evaluation vector $\underset{\sim 1\times5}{B}$. Then the water quality grade is obtained by the main factor highlighted decision model.

It is assumed that the value of weight vector ($W_{DO}$, $W_{COD}$, $W_{NH3\text{-}N}$) is (0.3, 0.4, 0.3), which has been determined by experts binary estimation method. Formula (7) can be used to calculate the final evaluation vectors for the two nodes. There are $\underset{\sim 1\times5}{B}=\{0.35143,0.14857,0.39000,0.01,0\}$ and $\underset{\sim 1\times5}{B}'=\{0.230001,0.069999,0,0.288,0.412\}$, According to the principle of maximum membership, the water quality grade of node 1 can be determined to be grade III as $b_3$=0.39000 which is the maximum value. Similarly, water quality grade of node 2 is grade V.

Step 5: Node ID and water quality grade are only to be transmitted to the cluster head if the water quality is equal or better than grade III. Otherwise it indicates that the pollution in this region is very serious, the pollution information such as type and value of pollutants obtained by SL395-2007 [12] should be transmitted to the cluster head together with the water quality grade and node ID. Data format is shown in Fig. 2. Where, a) 2 Bytes are used for storing the node ID. b) 3 bits are used for storing Water quality grades whose range is from 1 to 5, c) 2 bits can be used for storing the number of the types of pollutants which is determined mainly by the sensor types. d) The type ID and the value of pollutants can be stored by using 2 bits and 2 Bytes respectively. The information introduced in c) and d) only needs to be transmitted when there is a serious pollution. The advantage of the algorithm is that it can reduce data transmission and so reduce energy consumption.

In the example, only the node number and the water quality grade need to be transmitted as the water quality grade of node 1 is grade III. Its data format is shown in Fig. 3(a). While for the node 2, it indicates that this area has serious pollution as the water quality grade is grade V. So the information of pollutants (COD and $NH_3$-N) needs to be transmitted. Besides, the node number and the water quality grade also need to be transmitted. Its data format is shown in Fig. 3(b).

Step 6: The cluster heads firstly compresses the data collected from their member nodes. Then the compressed data is sent to the sink. Data format is shown in Fig. 4.

The length of data format is not fixed, because it is mainly related to the collected information of all the cluster member nodes. The total number of water quality grade is determined according to the grade information of each cluster member node and it can be stored by using 3 bits as its maximum value is 5. The data of various water quality grades are followed by, and it includes specific information of all cluster member nodes under the water grade. The data is arranged as follows: water quality grade is stored first, and then, the number of nodes which under the grade is stored, finally, is the specific information of nodes. Its format is similar to Fig. 2.

The compression is implemented by combining the nodes whose water quality grade are the same. Data compression can reduce network congestion.

In the example, it is assumed that the cluster head of nodes 1 and 2 is node 3. According to the principles mentioned above, the data format of node 3 is shown in Fig. 5.

| 2B | 3b | 2b | The data of 1-th pollutant | | The data of 2-th pollutant | ... | The data of M-th pollutant |
|---|---|---|---|---|---|---|---|
| | | | 2b | 2B | | | |
| Node ID | Grade of water quality | Number of pollutants' types | Type ID of the polluant | Exceeding value of the pollutant | | | |

Figure 2. Data format of cluster members.

| 2B | 3b |
|---|---|
| 1 | 3 |

(a)

| 2B | 3b | 2b | The data of 1-th pollutant | | The data of 2-th pollutant | |
|---|---|---|---|---|---|---|
| | | | 2b | 2B | 2b | 2B |
| 2 | 5 | 2 | 2 | 10.5 | 3 | 1.52 |

(b)

Figure 3. Data format of node 1 and node 2.

| 3b | The data of 1-th grade of water quality | | | | | | | | | | | The data of 2-th grade of water quality | ... | The data of 5-th grade of water quality |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 3b | 2B | The data of 1-th node | | | | | | The data of 2-th node | ... | The data of N-th node | | | |
| | | | 2B | 2b | The data of 1-th pollutants | | The data of 2-th pollutant | ... | The data of M-th pollutant | | | | | |
| | | | | | 2b | 2B | | | | | | | | |
| Number of water quality grades | Grade of water quality | Number of nodes under the grade | Node ID | Number of pollutants' types | Type ID of the polluant | Exceeding value of the pollutant | | | | | | | | |

Figure 4. Data format of cluster heads.

| 3b | The data of water quality grade III | | | The data of water quality grade V | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | 3b | 2B | The data of node 1 | 3b | 2B | The data of node 2 | | | | | |
| | | | 2B | | | 2B | 2b | The data of 1-th pollutant | | The data of 2-th pollutant | |
| | | | | | | | | 2b | 2B | 2b | 2B |
| 2 | 3 | 1 | 1 | 5 | 1 | 2 | 2 | 2 | 10.5 | 3 | 1.52 |

Figure 5. Data format of node 3.

Step 7: The monitoring center generates the water quality map according to the received data. Fig. 6 shows an example of water quality map.

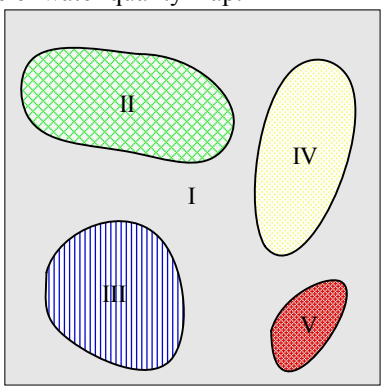



Figure 6. Water quality monitoring map in monitored area.

### B. Analysis of energy consumption

When the distance between two nodes is less than $d_0$, free space model is adopted, otherwise, Multi-channel attenuation model is used [13, 14]. The energy consumption is shown in formula (16). Where $E_{elec}$ is energy consumed by transmission circuit for sending or receiving one bit data. $\varepsilon_{fs}$ and $\varepsilon_{amp}$ represent the energy for amplifier in free space model and Multi-channel attenuation model respectively. The length of the data is k bit.

$$E_{Tx}(k,d)=\begin{cases}k*(E_{elec}+\varepsilon_{amp}*d^2) & d<d_0\\ k*(E_{elec}+\varepsilon'_{amp}*d^4) & d\geq d_0\end{cases} \quad (16)$$

The energy consumption of receiving k bit data is shown in formula (17):

$$E_{Rx}(k)=k*E_{elec} \quad (17)$$

The energy consumed by calculation is neglected, since the computation involved in WQMAFCE is simple and the energy consumption of calculation is much lower than that of communication.

The energy consumption of the algorithm proposed in the paper is compared with the centralized algorithm according to the communication model under the circumstance that only one cluster exists. Provided that the number of the cluster member nodes is N, and 2 Bytes is needed for storing the node ID. Each node equipped with M kinds of sensors, each sensor data needs 2 Bytes to store. The free space model is used for

member nodes transmitting data to the cluster head, and the average distance between them is d. The Multi-channel attenuation model is used for cluster head transmitting data to the Sink, and the distance between them is d'.

The energy consumption per round of the two algorithms is compared as follows:

*1) Centralized Algorithm*

*a)* The energy consumption of cluster members sending data to the cluster head is shown in formula (18):

$$E_{Tx} = N*(M+1)*16*(E_{elec} + \varepsilon_{amp} * d^2) \quad (18)$$

*b)* The energy consumption of cluster head receives data from all the members is shown in formula (19):

$$E_{Rx} = N*(M+1)*16*E_{elec} \quad (19)$$

*c)* The energy consumption of the cluster head sends data to the Sink is shown in formula (20):

$$E'_{Tx} = N*(M+1)*16*(E_{elec} + \varepsilon'_{amp} * d'^4) \quad (20)$$

*2) WQMMFCE*

The data volume for transmission is determined by the current water quality: if the water quality is equal or better than grade III, only the grade information and node ID need to be transmitted. Otherwise, it needs to transmit the pollution information together with the grade information and node ID. The WQMMFCE performs best if the water quality is equal or better than grade III by taking data transmission as a metric. It is the worst case if the water is worse than grade III. The two cases are analyzed respectively as follows:

*a)* The energy consumption of cluster members sending data to the cluster head is shown in formula (21) and (22):

For the best case:

$$E_{Tx} = N*19*(E_{elec} + \varepsilon_{amp} * d^2) \quad (21)$$

For the worst case:

$$E_{Tx} = N*(M*18+21)*(E_{elec} + \varepsilon_{amp} * d^2) \quad (22)$$

*b)* The energy consumption of cluster head receiving data from all the members is shown in formula (23) and (24):

For the best case:

$$E_{Rx} = N*19*E_{elec} \quad (23)$$

For the worst case:

$$E_{Rx} = N*(M*18+21)*E_{elec} \quad (24)$$

*c)* The energy consumption of the cluster head sends data to the Sink is shown in formula (25) and (26):

For the best case:

$$E'_{Tx} = (22+N*16)*(E_{elec} + \varepsilon'_{amp} * d'^4) \quad (25)$$

For the worst case:

$$E'_{Tx} = [98+N*(M+1)*18]*(E_{elec} + \varepsilon'_{amp} * d'^4) \quad (26)$$

## VI. Simulation Analysis

In order to verify the validation of the method, OMNet++ is used for simulation. 200 nodes are deployed in an area of 1000x1000 uniformly. The coordinates of Sink is (500, 1000). Other parameters are set as follows: $E_{elec}$ is 50nJ/bit, $\varepsilon_{amp}$ is 10pJ/(bit*m^2), $\varepsilon'_{amp}$ is 0.0013pJ/(bit*m^4).The initial energy of each node is 2J and the experiment runs 2000 rounds.

Fig. 7 shows the comparisons of energy consumption between WQMMFCE and the centralized one. The energy consumption is increasing with the increment of running rounds for both methods. For the best case of WQMMFCE, its energy consumption is much lower than the centralized one. While for the worst case, its energy consumption is slightly higher than the centralized one. But in actual situation, the large-scale pollution can be occurred in short duration low-frequency. Therefore, WQMMFCE can achieve a very good energy saving effect due to node ID and water quality grade are only need to be transmitted most of the time.

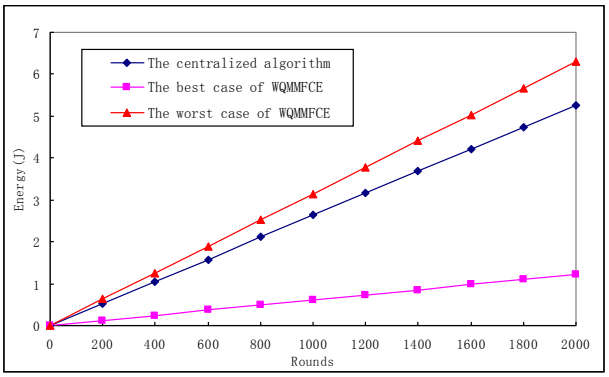



Figure 7. Comparison of energy consumption.

## V. Conclusion

A water quality monitoring method based on fuzzy comprehensive evaluation is proposed under the background that the wireless sensor network is used for the application of water quality monitoring. The key point of the method is that the weight of factors should be calculated firstly by using binary expert evaluation in fuzzy comprehensive evaluation. Then each sensor node calculates the water quality by using fuzzy comprehensive evaluation method. Whether the pollution information should be transmitted or not is determined by the grade of water quality. Simulation results show that it can reduce energy consumption and prolong the network life compared with the centralized method.

## References

[1] 11th Five-Year Plan on National Comprehensive Disaster Mitigation Unveiled: http://jzs.mca.gov.cn/article/zjz/200801/20080100009537.shtml.

[2] Chong C Y, Kumar S P, "Sensor networks: Evolution, opportunities and challenges," *Proceedings of the IEEE*, vol. 91, pp. 1247-1256, 2003.

[3] GreenOrbs: http://www.greenorbs.org.

[4] Design and Construction of a Wildfire Instrumentation System Using Networked Sensors: http://firebug.sourceforge.net.

[5] 2008 Report on the State of the Environment in China: http://english.mep.gov.cn/standards_reports/soe/soe2008/201002/t20100224_186070.htm.
[6] Xie Li-jian, Liu Cheng-ping, *Fuzzy mathematics method and its application*. Wuhan: Huazhong University of Science & Technology Press Co Ltd, 2006.
[7] GB3838-2002, Environmental quality standard for surface water.
[8] Tang Yao-ping, "The methods to Fetch i in difference degree coefficient of set pair analysis and its applications," *Mathematics in practice and theory*, vol. 39, pp. 67-70, 2009.
[9] Cao Yong-qiang, Wang Ben-de, Liu Jin-lu, "Water quality assessment model based on dualistic factor contrast for Indexes weight calculation and its application," *International Journal Hydroelectric Energy*, vol. 20, pp. 19-21, 2002.
[10] Deb B, Bhatnagar S, Nath B, "A topology discovery algorithm for sensor networks with applications to network management," *DCS Technical Report DCS-TR-441, Rutgers University*, 2001.
[11] Jian Shu, Ronglei Zhang, Linlan Liu, Zhenhua Wu and Zhiping Zhou, "Cluster-based Three-dimensional Localization Algorithm for Large Scale Wireless Sensor Networks," *Journal of Computers*, vol. 4, pp. 585 -592, July 2009.
[12] SL395-2007, Technological regulations for surface water resources quality assessment.
[13] Wang A, Heinzelman W B, Sinha A, et al, "Energy-scalable protocols for battery-operated micro-sensor networks," *Journal of VLSI Signal Processing*, vol. 29, pp. 223-237, 2001.
[14] Yuan Lingyun, Wang Xingcha, Zhao Yanfang, Gan Jianhou, "Emergent Event Monitoring Based on Event-Driven and Minimum Delay Aggregation Path in Wireless Sensor Network," *Chinese Journal of Sensors and Actuators*, vol. 22, pp. 1312-1317, 2009.

**Jian Shu** is a professor in school of Software, Nanchang Hangkong University, China. He received a Ms. in Computer Networks from Northwestern Polytechnical University in 1990. His research interests include wireless sensor network, embedded system and software engineering. He is the director of Internet of the Things Technology Institute, Nanchang Hangkong University.

**Ming Hong** is a postgraduate student of Nanchang Hangkong University. His research interest is wireless sensor network.

**Linlan Liu** is a professor in school of Information Engineering, Nanchang Hangkong University, China. She received a Bs. in Computer Application from National University of Defence Technology in 1988. Her research interests include Software Engineering and Distributed System.

**Yebin Chen** is a teacher in school of software, Nanchang Hangkong University, China. He received a M.S. in Computer Application Technology from Nanchang Hangkong University in 2010. His research interest is wireless sensor network.

# Capacity of 60 GHz Wireless Communication Systems over Fading Channels

Jingjing Wang[1, 2]
[1]Department of Electrical Engineering, Ocean University of China, Qingdao, China
[2]College of information Science & Technology, Qingdao University of Science & Technology, Qingdao,China
Email: wangjingjing@qust.edu.cn

Hao Zhang[1,3], Tingting Lv[1] and T. Aaron Gulliver[3]
[3]Department of Electrical Computer Engineering, University of Victoria, Victoria, Canada
Email: zhanghao@ouc.edu.cn, lvtingting33@163.com, agullive@ece.uvic.ca

***Abstract*—This paper considers the channel capacity of 60GHz wireless communications systems over Rayleigh fading channels and Ricean fading channels. The SNR and therefore capacity varies according to the communication distance. The capacity is presented for line-of-sight (LOS) and non-line-of-sight (NLOS) channels given based on a 60GHz link budget model. Phase shift keying (PSK) modulation is considered under FCC power constraints for the unlicensed 59-64GHz radio spectrum. The channel capacity over Rayleigh fading channels is compared with the capacity in additive white Gaussian noise channels. The paper also investigates the channel capacity of 60GHz wireless communications systems over Ricean fading channels and gives the channel capacity compariasion with q-ary PSK modulation over Ricean fading channel, AWGN channel and Rayleigh channel when the SNR per symbol is given. The results show that a 60GHz wireless system is more suitable for short range communications less than 100 meters rather than long distances.**

***Index Terms*—Channel capacity, AWGN, Rayleigh fading, Ricean fading, 60GHz**

## I. INTRODUCTION

The desire for unrestricted access to information and in particular multimedia spurs the growth of wireless communications. However, the lower frequency spectrum is almost completely occupied. Fortunately, an abundance of widely available spectrum around 60 gigahertz (GHz) is available to support high-rate, unlicensed wireless communications. The up to 7 GHz of bandwidth is very suitable for short-range wireless communication, and is an excellent prospect for future system development.

The regulations and standards of each country are slightly different according to IEEE 802.15.3c [1][2][3][4][5]. In 2001, the United States Federal Communications Commission (FCC) set aside 7 GHz of contiguous spectrum between 57 and 64 GHz for unlicensed use. In 2000, the Ministry of Public Management, Home Affairs, Posts, and Telecommunications (MPHPT) of Japan issued 60 GHz radio regulations for unlicensed utilization in the 59–66 GHz band [2]. The 54.25–59 GHz band is, however, allocated for licensed use. The maximum transmit power in the unlicensed band is limited to 10 dBm with a maximum allowable antenna gain of 47 dBi. Unlike in North America, Japanese regulations specify that the maximum transmission bandwidth must not exceed 2.5 GHz. There is no specification for RF radiation exposure and transmitter identification requirements. In Europe, point-to-point fixed services in the 64–66 GHz band is recommended. Figure 1 shows the international unlicensed spectrum around 60GHz [1][2].

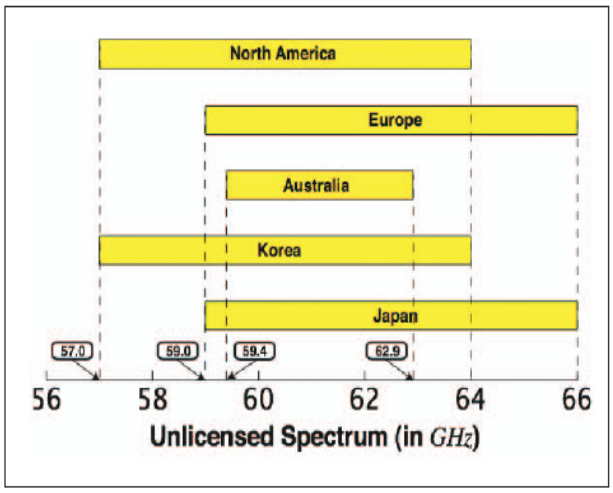



Figure 1. International unlicensed spectrum around 60 GHz.

With 7 GHz of bandwidth, 2-3Gbit/s high definition media interface (HDMI) or wireless gigabit Ethernet could be achieved even using simple modulation methods, for example, PAM, PSK and QAM. Furthermore, 60 GHz regulations allow for a much higher transmit power (10W) compared to existing wireless local area network (WLAN) and wireless personal area

This work is supported by Outstanding Youth Fund of ShanDong province under Grant no.JQ200821 and New Century Educational Talents Plan of Chinese Education Ministry under Grant no. NCET-08-0504


doi:10.4304/jnw.7.1.203-209

network (WPAN) systems. The higher transmit power is necessary to overcome the higher path loss at 60 GHz.

60 GHz wireless systems present several challenges that have made deployment difficult [2][3][4]. The 60 GHz channel has 20 to 40 dB increased free space path loss compared to lower frequency bands, and suffers from 15 to 30 dB/km of atmospheric absorption depending on the conditions. In addition, multipath effects are vastly reduced at 60 GHz making non-line-of-sight (NLOS) communications very difficult. Furthermore, increased phase noise, limited amplifier gain, and the need for transmission line modeling of circuit components due to the ultra high frequencies [2][3][4][5] are challenges for millimeter-wave transceivers.

IBM engineers have reported the development of 60 GHz front-end chip sets [2][5], and the first experimental 60 GHz transmitter and receiver chips using a high-speed alloy of silicon and germanium (SiGe). Meanwhile, researchers from UCLA, UC Berkeley Wireless Research Center (BWRC), and other universities and institutes are using widely available and inexpensive complementary metal oxide semiconductor (CMOS) technology to build 60 GHz transceiver components.

To date, there have been few results on the channel capacity of these systems. The capacity of a 60 GHz wireless communication system over an AWGN channel was presented in [6]. However, Rayleigh fading channels and Ricean fading channels are more practical, especial in non-line-of-sight (NLOS) and part line-of-sight (LOS) environments. This paper investigates the capacity of 60 GHz systems over Rayleigh fading channels and Ricean fading channels considering the FCC transmission rules.

The rest of the paper is organized as follows. Section II presents the 60GHz link budget model and Shannon capacity. The channel capacity over Rayleigh fading channels and Ricean fading channels is calculated in Section III, and results to illustrate the capacity are given in Section IV. Finally, Section V concludes the paper.

## II. 60G LINK BUDGET MODEL AND SHANNON CAPACITY

### A. *60GHz Link Budget Models*

In 2001, the FCC set aside 7 GHz of spectrum between 57 and 64 GHz for wireless communications. Their rules limit the equivalent isotropic radiated power in this band to a maximum power density of 9 μW/cm$^2$ at 3 meters from the radiating source [7]. This means that 40 dBm transmit power is the legal power limit with an antenna having 0 dBi gain.

The link budget model according to Friis free-space path loss formula is

$$P_r = P_t + G_t + G_r - P_L \tag{1}$$

where Pt is the transmit power, Pr is the received power at distance d, Gt and Gr are antenna gain for the transmit and receive antennas respectively, both assumed to be 0 dB for simplicity. All expressions are in decibels (dB). It is shown in [8] that the received signal strength is dominated by the distance from the transmitter and the receiver, taking into account the oxygen absorption. The general path loss model can be expressed as

$$P_L(dB) = 10\log_{10}\left(\frac{4\pi d}{\lambda}\right)^n \tag{2}$$

where λ is the wavelength corresponding to the center frequency fc, n is the path loss exponent which can be approximated as 1.55 for line-of-sight (LOS) channels and 2.44 for non-line-of-sight (NLOS) in an indoor home environment (5-15m) [10]. In conference room environments, n can be approximated as 1.77 in line-of-sight (LOS) channels and 3.85 in non-line-of-sight (NLOS) environments [11]. Using the frequency range from 57 to 64 GHz, the constraint on the transmit power is

$$Pt \le 40dBm \tag{3}$$

If thermal noise as the primary source of interference, the required sensitivity at the receiver can be calculated as

$$S_r = NF + F + SNR \tag{4}$$

where NF is the noise floor calculated by thermal noise

$$N = kTWF \tag{5}$$

F is the noise figure (optimistically) assumed to be 0 dB, SNR is the signal to noise ratio at the receiver, k is Boltzmann's constant, and T is the room temperature (typically 290K). For the 60 GHz system, the noise floor is calculated as -76 dBm. To ensure adequate performance at the receiver, the minimum received power should be greater than or equal to the required sensitivity. Thus the relationship between the system performance and maximum communication distance can be derived as

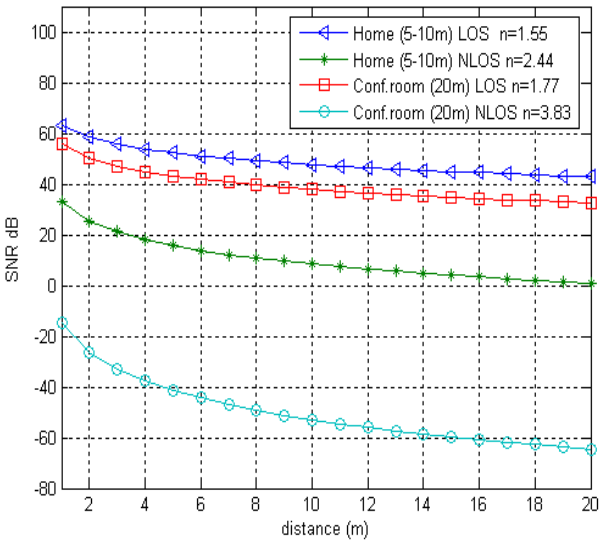



Figure 2. Received SNR versus communication distance for a 60GHz wireless system over LOS and NLOS channels and employing the FCC in band power density limit.

$$SNR \le 116 - 10\log_{10}\left(\frac{4\pi d}{\lambda}\right)^n \quad (6)$$

The relationship between SNR and distance is shown in Fig. 2. This shows that there is about 20dB in attenuation in LOS environments in both the home (5-15m) and conference room (20m) environments. However, in NLOS channels, the SNR degradation as distance increases is very significant, especially in conference room environments.

### B. Shannon Capacity

Channel capacity can be calculated according to the Shannon capacity [12] which is given by

$$C = W\log(1+SNR) \quad (7)$$

where W is the system bandwidth, SNR is the receive signal to noise ratio, defined as Eb/N0, where Eb is the energy per bit and N0 is the noise power spectral density. The relationship between the capacity and communication distance is then given by

$$C \le W\log_2(1+10^{(116-10\log_{10}(\frac{4\pi d}{\lambda})^n)/10}) \quad (8)$$

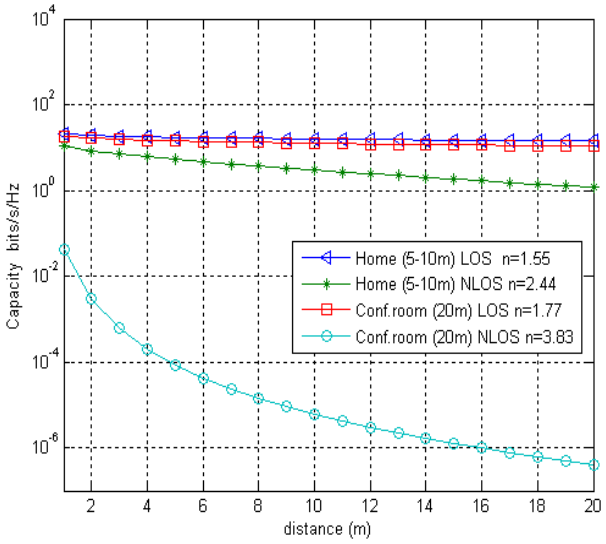



Figure 3. The achievable Shannon capacity versus communication distance.

Figure 3 shows the Shannon capacity limit for the LOS and NLOS cases. It can be observed that there is only a small capacity decrease for the LOS cases (n=1.55 and n=1.77), while the operating distance for the NLOS n=3.83 case is limited to several meters because the capacity decreases drastically as a function of distance. To improve the capacity, we can either increase the bandwidth or signal-to-noise ratio (SNR), or both. According to (1), improving the transmit antenna gain Gt and/or the receive antenna gain Gr can also improve the capacity.

## III. 60 GHz CHANNEL CAPACITY OVER RAYLEIGH AND RICEAN FADING CHANNELS

### A. Channel Capacity with PSK Modulation over AWGN Channels

The Shannon capacity expression (7) corresponds to continuous-valued inputs and outputs. However, a channel employing multilevel/phase modulation, for example PAM, PSK or QAM modulation, has discrete-valued inputs and continuous-valued outputs, which imposes an additional constraint on the capacity calculation. We consider modulation channels with discrete-inputs and continuous-outputs, which was capacity given in [12] as

$$C = \max_{P(x_i)} \sum_{k=0}^{q-1} \int_{-\infty}^{\infty} p(y\,|\,x_k) P(x_k) \log_2 \frac{p(y\,|\,x_k)}{p(y)} dy \quad (9)$$

where

$$p(y) = \sum_{i=0}^{q-1} p\left(y|x_i\right) P\left(x_i\right) \quad (10)$$

*xk* is the discrete-valued input, and y is the continuous-valued output, modeled as

$$y(t) = x(t) + w(t) \quad (11)$$

where w(t) is additive white Gaussian noise (AWGN) with variance N0/2 in each dimension.

Assuming an equal a priori probability real or complex signal constellation, i.e. $P(x_i)=1/q$, the channel capacity of an AWGN channel with q-ary modulation is then [6][13]

$$\begin{aligned} C &= \log_2(q) - \frac{1}{q}\sum_{k=0}^{q-1} \mathbf{E}_{y|x_k}\left\{\log_2 \frac{\sum_{i=0}^{q-1} p\left(y|x_i\right)}{p\left(y|x_k\right)}\right\} \\ &= \log_2(q) - \frac{1}{q}\sum_{k=0}^{q-1} \mathbf{E}_{y|x_k}\left\{\log_2 \sum_{i=0}^{q-1} \exp\left[-\frac{|x_k + w - x_i|^2 - |w|^2}{2\sigma^2}\right]\right\} \end{aligned} \quad (12)$$

where E[.] denotes expectation, w is complex white Gaussian noise, modeled as a Gaussian distributed random variable with zero mean and variance $\sigma^2$ in each real dimension. Equation (12) can be evaluated by Monte Carlo simulation. Note that (12) is a universal formula which applies to q-ary PAM/PSK/QAM, although PSK is commonly used in 60 GHz systems. With normalized signal energy, the relationship between channel capacity and SNR can be evaluated using (12), as well as the relationship between channel capacity and communication range.

### B. Channel Capacity with PSK Modulation over Rayleigh and Ricean Fading Channels

On wireless channels, channel capacity is typically degraded by fading phenomena which arise from multipath propagation. In NLOS cases, the channel can be modeled using a Rayleigh distribution. While in most cases, there are LOS paths and NLOS paths, so the channel can be modeled using a Ricean distribution. The

complex process received at the output of a noisy flat-fading wireless channel is then

$$y(t) = h(t)x(t) + w(t) \tag{13}$$

where h(t) is a generally time-correlated ergodic fading complex sequence independent of x(t) and w(t), and w(t) is complex zero mean AWGN with variance N0/2 in each dimension. Assuming coherent detection at the receiver, the effect of fading is reduced to multiplication of the transmitted symbol x(t) by the real nonnegative random variable h(t), which represents the envelope of the complex fading. Therefore, without loss of generality, we can rewrite (13) in an equivalent sampled form.

$$y(n) = h(n)x(n) + w(n) \tag{14}$$

With perfect channel state information available at the receiver, it is known that the capacity of the channel in (14) can be directly obtained by averaging the corresponding conditional capacity $\tilde{C}(h)$ with respect to the probability density function (pdf) of the fading gain. This leads to the following expression for the channel capacity of fading channels for an equiprobable signal constellation

$$C = \int_0^\infty \tilde{C}(h)p(h)dh \tag{15}$$

where

$$\tilde{C}(h) = log_2(q) - \frac{1}{q}\sum_{k=0}^{q-1} E_{y|x_k}\left\{log_2 \sum_{i=0}^{q-1} exp\left[-\frac{|y-hx_i|^2 - |y-hx_k|^2}{2\sigma^2}\right]\right\} \tag{16}$$

With Rayleigh fading, h can be modeled as a complex Gaussian variable with zero mean and variance $\sigma^2$ in each dimension. Rayleigh fading model assumes that all paths are NLOS paths and there is no a dominant path.

While for Ricean fading, h can be modeled as a complex Gaussian variable with means m for the real and imaginary parts, and variance $\sigma^2$ in each dimension. The Ricean parameter is defined as $\beta = \frac{m^2}{\sigma^2}$ which means the ratio of the LOS paths power to the NLOS paths power.

## Ⅳ. NUMERICAL RESULTS

Monte Carlo simulation was used to evaluate the channel capacity of a 60 GHz communication system over Rayleigh fading channels under FCC restrictions. These results are compared with the AWGN channel capacity.

To provide a basis for comparison, Fig. 4 gives the normalized channel capacity of q-ary PSK modulation over an AWGN channel [6]. This shows that the achievable data rate for BPSK is 3.75 Gbps with 5 GHz bandwidth and an SNR of 0 dB.

The normalized channel capacity of q-ary PSK over Rayleigh fading channels is shown in Fig. 5. This shows that the achievable data rate for BPSK is about 2.9 Gbps with 5GHz bandwidth at an SNR of 0 dB. Thus there is a significant capacity decease due to the fading.

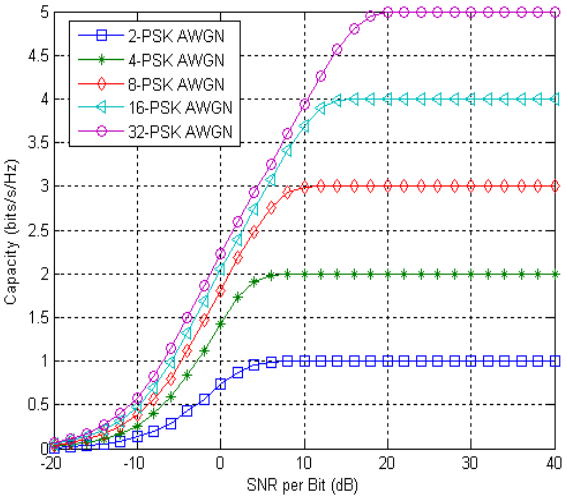



Figure 4. Channel capacity with q-ary PSK modulation over an AWGN channel.

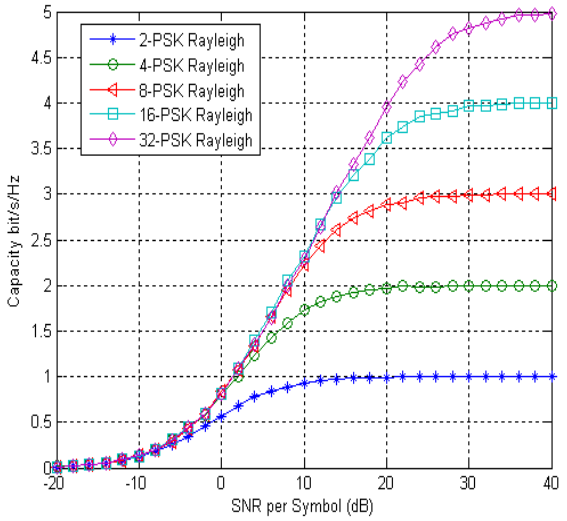



Figure 5. Channel capacity of q-ary PSK over Rayleigh fading channels.

A comparison of q-ary PSK capacity over AWGN and Rayleigh fading channels is given in Fig. 6. As expected, there is no significant difference in the capacities when the SNR is less than 5dB, since the noise dominates performance. However, for values of SNR from 5dB to 20 dB, the capacity decease due to the fading is obvious. In addition, the capacity decease over Rayleigh fading channels is greater as q increases.

The channel capacity of a 60 GHz communication system over Ricean fading channels under FCC restrictions is simulated below. These results are compared with the AWGN channel and the Rayleigh fading channel capacity.

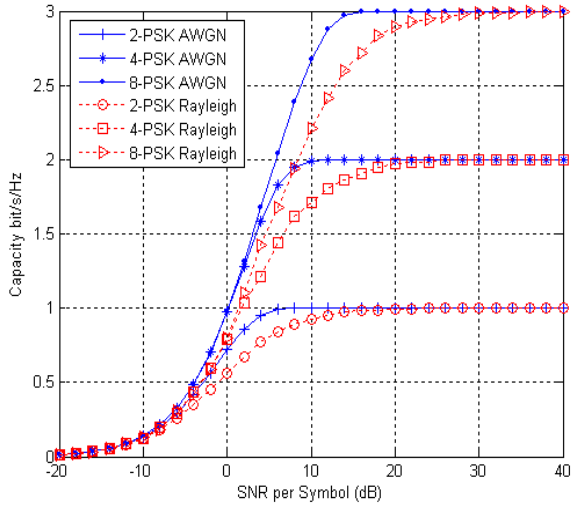



Figure 6. Comparison of q-ary PSK channel capacity over AWGN and Rayleigh fading channels.

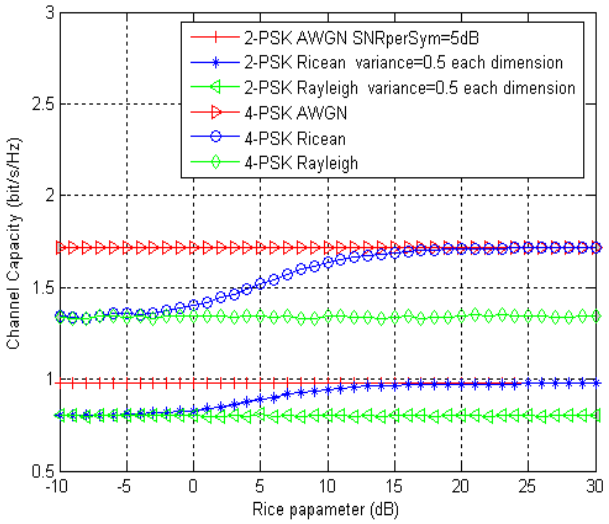



Figure 7. Channel capacity compariasion with q-ary PSK modulation over Ricean fading channel, AWGN channel and Rayleigh channel when the SNR per symbol is 5 dB.

To provide a basis for comparison, Fig. 7 gives the normalized channel capacity of q-ary PSK modulation over Ricean fading channel, AWGN channel and Rayleigh channel when the SNR per symbol is 5 dB. This shows that the achievable data rate for BPSK and 4PSK is increasing with the Ricean parameter increasing.

Fig. 8 gives the normalized channel capacity of q-ary PSK modulation over Ricean fading channel, AWGN channel and Rayleigh channel when the SNR per symbol is 10 dB. Fig. 2 shows that the achievable data rate for BPSK and 4PSK is increasing with the Ricean parameter too. But the increasing of channel capacity when the SNR per symbol is 5 dB is more clearly compared to the SNR per symbol with 10 dB.

Fig.7 and Fig.8 show that when the Ricean parameter is small, for example, less than -5dB, the channel capacity over Ricean fading channel is near to the channel capacity over Rayleigh fading channel because there is much more NLOS paths than LOS paths when the Ricean parameter is small. On the contrary, when the Ricean parameter is big, for example, bigger than 15dB, the channel capacity over Ricean fading channel is near to the channel capacity over AWGN channel because there is much more LOS paths than NLOS paths when the Ricean parameter is big.

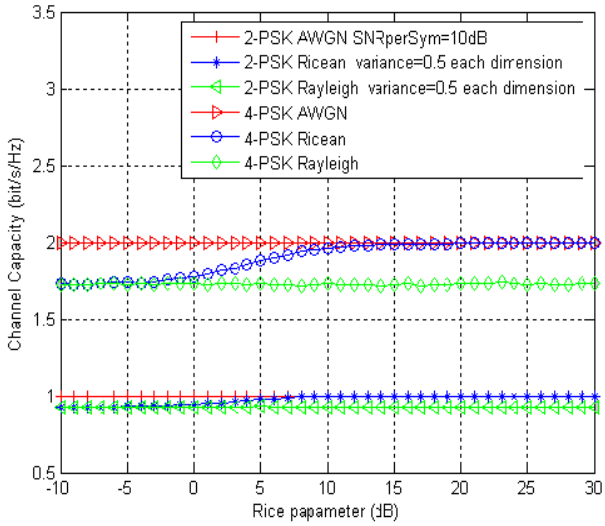



Figure 8. Channel capacity compariasion with q-ary PSK modulation over Ricean fading channel, AWGN channel and Rayleigh channel when the SNR per symbol is 10 dB.

## V. CONCLUSIONS

The capacity of 60GHz wireless communications over Rayleigh fading channels and Ricean fading channels was investigated for PSK modulation employing FCC in band power density limits. In NLOS cases, the channel can be modeled using a Rayleigh distribution. While in most cases, there are LOS paths and NLOS paths, so the channel can be modeled using a Ricean distribution.

The relationship between channel capacity and SNR or communication range was demonstrated over Rayleigh fading channels in different channel conditions. As expected, the lower the path loss exponent, the longer the communication range. It can be concluded that a 60GHz wireless system is more suitable for short range communications less than 1 km rather than long distances. The q-ary PSK channel capacity over Rayleigh fading channels was shown to be less than the capacity in AWGN channels, particularly in the 5-20 dB SNR range. The capacity decease over Rayleigh fading channels is more serious with increasing q. The relationship between channel capacity and Ricean parameter was demonstrated in different SNR conditions. As expected, the bigger the Ricean parameter, the greater the channel capacity.

## ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their constructive comments and questions

which greatly improved the paper. The authors would like to thank the support of Outstanding Youth Fund of ShanDong Province and New Century Educational Talents Plan of Chinese Education Ministry.

## REFERENCES

[1] F. Giannetti, M. Luise, and R. Reggiannini, "Mobile and personal communications in 60 GHz band: A survey," Wirelesss Personal Comun., vol. 10, pp. 207–243, 1999.

[2] S.K. Yong and C.-C. Chong, "An overview of multigigabit wireless through millimeter wave technology: Potentials and technical challenges," EURASIP J. on Wireless Commun. and Networking, vol. 2007, 2007.

[3] C. Doan, S. Emami, D. Sobel, A. Niknejad, and R. Brodersen, "Design considerations for 60 GHz CMOS radios," IEEE Commun. Mag., vol. 42, no. 12, pp. 132–140, Dec. 2004.

[4] M. Karkkainen, M. Varonen, P. Kangaslahti, and K. Halonen, "Integrated amplifier circuits for 60 GHz broadband telecommunication," Analog Integrated Circuits and Signal Processing, vol. 42, pp. 37–46, 2005.

[5] N. Guo, R.C. Qiu, S.S. Mo, and K. Takahashi, "60 GHz millimeter-wave radio: Principle, technology, and new results," EURASIP J. on Wireless Commun. and Networking, vol. 2007, 2007.

[6] H. Zhang and T.A. Gulliver, "On capacity of 60 GHz wireless communications over AWGN channels," Proc. Canadian Conf. on Electrical and Computer Eng., pp. 936-939, May 2009.

[7] ERC Recommendation 12-09, "Radio Frequency Channel Arrangement for Fixed Service Systems Operating in the Band 57.0 - 59.0 GHz which do not Require Frequency Planning, The Hague 1998 Revised Stockholm", Oct. 2004.

[8] N.D. Hawkins, R. Steele, D.C. Rickard, and C.R. Shepherd, "Path loss characteristics of 60 GHz transmissions", Elect. Lett., vol. 21, no. 22, pp. 1054 – 1055, Oct. 1985.

[9] Federal Communications Commission, "Amendment of Parts 2, 15 and 97 of the Commission's Rules to Permit Use of Radio Frequencies Above 40 GHz for New Radio Applications", FCC 95-499, ET Docket No. 94-124, RM-8308, Dec. 1995. Available via: ftp://ftp.fcc.gov./pub/Bureaus/Engineering_Technology/ Orders /1995/fcc95499.txt

[10] M. K. Simon and M. S. Alouini, Digital Communication over Fading Channels, 2nd Ed., Wiley-IEEE Press, New York, NY, USA, 2004.

[11] M. Fiacoo and S. Saunders, Final report for OFCOM - Indoor Propagation Factors at 17GHz and 60GHz, Aug. 1998.

[12] J.G. Proakis, "Digital Communications," 4th Ed., McGraw-Hill, 2001.

[13] H. Zhang and T.A. Gulliver, "Capacity and error probability analysis for orthogonal space-time block codes over fading channels," IEEE Trans. Wireless Commun., vol. 4, no. 2, pp. 808-819, Mar. 2005.

[14] P. Pagani, I. Siaud, N. Malhouroux, W. Li, "Adaptation of the France Telecom 60 GHz channel model to the TG3c framework," IEEE 802.15-06-0218-00-003c, April 2006.

[15] M. Fiacco, M. Parks, H. Radi and S. R. Saunders, "Final Report: Indoor Propagation Factors at 17 and 60GHz," Tech. report and study carried out on behalf of the Radiocommunications Agency, University of Surrey, Aug. 1998.

[16] C. R. Anderson and T.S. Rappaport, "In-Building Wideband Partition Loss Measurements at 2.5 and 60 GHz." IEEE Trans. Wireless Comm. vol. 3, no. 3, pp. 922-928, May 2004.

[17] A. Bohdanowicz, "Wideband Indoor and Outdoor Radio Channel Measurements at 17 GHz" UBICOM Technical Report, Jan 2000

[18] H. J. Thomas et al., "An experimental study of the propagation of 55 GHz millimeter waves in an urban mobile radio environment," IEEE Trans. Veh. Tech., vol. 43, no. 1, pp. 140-146, Feb 1994.

[19] H. B Yang, M. H. A. J. Herben and P. F. M. Smulders, "Impact of Antenna Pattern and Reflective Environment on 60 GHz Indoor Radio Channel Characteristics, "IEEE Antennas and Wireless Propagation Letter, Vol. 4, 2005.

[20] S. Collonge, G. Zaharia and G. E. Zein, "Influence of the human activity on wideband characteristics of the 60GHz indoor radio channel," IEEE Trans. Wireless Comm. vol. 3, no. 6, pp. 2389-2406, Nov 2004.

[21] P. F. M. Smulders, "Broadband Wireless LANs: A Feasibility Study," Ph.D. Thesis, Eindhoven University, 1995

[22] A. Sadri, A. Maltsev and A. Davydov., "IMST Time Angular Characteristics Analysis," IEEE 802.15-06-0141-01-003c, Denver, USA, Mar 2006.

[23] A. Saleh and R. Valenzuela, "A Statistical Model for Indoor Multipath Propagation," IEEE J. Select. Areas Commun., Vol. SAC-5, No. 2, pp. 128-137, Feb. 1987.

[24] Q. Spencer, B. D., Jeffs, M. A. Jensen and A. L. Swindlehurst, "Modeling the Statistical Time and Angle of Arrival Characteristics," IEEE J. Sel. Areas Commun., Vol. 18, No. 3, pp. 347-360, Mar. 2000.

[25] C. -C. Chong, C, -M. Tan. D. I. Laurenson, S. McLaughlin, M. A. Beach, and A. R. Nix, "A new statistical wideband spatio-temporal channel model for 5-GHz band WLAN systems," IEEE J. Sel. Areas Commun., vol. 21, no. 2, pp. 139–150, Feb. 2003.

[26] R. C. Qiu and I. Lu, "Multipath resolving with frequency dependence for wideband wireless channel modeling," IEEE Trans. Vech. Technol., vol. 48, no. 1, pp. 273–285, Jan 1999.

[27] S. Emami, Z. Lai, A. Mathew and B. Gaucher, "Channel Model Based on IBM Measured Data," IEEE 802.15-06-0191-00-003c.

[28] M. S. Choi, G. Grosskopf and D. Rohde," Statistical Characteristics of 60 GHz Wideband Indoor Propagation Channel, IEEE PIMRC'05, Sept. 2005.

[29] M. Steinbauer and A. F. Molisch, "Directional channel models", Chapter 3.2 (pp. 132-193) of "Flexible Personalized Wireless Communications", 2001, John Wiley & Sons, U.K..

[30] C. -C. Chong and S. K. Yong, "A Generic Statistical-Based UWB Channel Model for High-Rise Apartments," IEEE Trans. Antennas Propat. , vol. 53, no. 8, pp 2389-2399, Aug. 2005.

**Jingjing Wang** was born in Anhui, China, in 1975. She received her B.S. degree in industrial automation from Shandong University, Jinan, China, in 1993, the M.Sc. degree from control theory and control engineering, Qingdao University of Science & Technology, Qingdao,China in 2002.

From 1997 to 1999, she was the assistant engineer of Shengli Oilfield, Dongying, China. From 2002 to now, she is a associate professor at the College of information Science & Technology, Qingdao University of Science & Technology. Her research interests include 60GHz wireless communication, and ultrawideband radio systems.

**Hao Zhang** was born in Jiangsu, China, in 1975. He received his B.S. degree in telecom engineering and industrial management from Shanghai Jiaotong University, Shanghai, China, in 1994, the M.B.A. degree from New York Institute of Technology, Old Westbury, NY, in 2001, and the Ph.D. degree in electrical and computer engineering from the University of Victoria, Victoria, BC, Canada, in 2004.

From 1994 to 1997, he was the Assistant President of ICO(China) Global Communication Company, Beijing, China. He was the Founder and CEO of Beijing Parco Company, Ltd., Beijing, China, from 1998 to 2000. In 2000, he joined Microsoft Canada, Vancouver, BC, as a Software Engineer, and was Chief Engineer at Dream Access Information Technology, Victoria, BC, Canada, from 2001 to 2002. He is currently an Adjunct Assistant Professor with the Department of Electrical and Computer Engineering, University of Victoria. His research interests include ultrawideband radio systems, MIMO wireless systems, and spectrum communications.

**Tingting Lv** was born in Qingdao, China, in 1983. She received her B.S. degree in communication engineering from Hunan University, Changsha, China, in 2006, the M.Sc. degree from Department of Electrical Engineering, Ocean University of China, Qingdao, China, in 2009. Her research interests include ultrawideband radio systems, MIMO wireless systems, and 60GHz wireless communication.

**T. Aaron Gulliver** (S'87–M'89–SM'96) received the B.Sc. and M.Sc. degrees in electrical engineering from the University of New Brunswick, Fredericton, NB, Canada, in 1982 and 1984, respectively, and the Ph.D. degree in electrical and computer engineering from the University of Victoria, Victoria, BC, Canada, in 1989.

From 1989 to 1991, he was a Defence Scientist with the Defence Research Establishment Ottawa, Ottawa, ON, Canada, where he was primarily involved in research on satellite communications. From 1990 to 1996, he was with the Department of Systems and Computer Engineering, Carleton University, Ottawa. From 1996 to 1999, he was a Senior Lecturer with the Department of Electrical and Electronic Engineering, the University of Canterbury, Christchurch, New Zealand. He is currently a Professor at the University of Victoria. His research interests include wireless communications, algebraic coding theory, cryptography, and spread spectrum systems.

Dr. Gulliver is a member of the Association of Professional Engineers of Ontario.

# Data Synchronization and Resynchronization for Heterogeneous Databases Replication in Middleware–based Architecture

Guoqiong Liao
School of Information Technology, Jiangxi University of Finance and Economics, Nanchang 330023, China
Jiangxi Key Laboratory of Data and Knowledge Engineering, Nanchang 330023, China
liaoguoqiong@163.com

***Abstract*—Currently more and more web applications and telecommunication applications are required to be separation with database layer. A general solution is to employ a middleware-based architecture, including an application client tier, a database front end (DB-FE) tier, also called middleware tier, and a database backend (DB-BE) tier. This paper mainly focuses on the issues of data synchronization and resynchronization in the case of failures for the architecture. Firstly, a synchronous update everywhere replication model and a replication 2-phase commit protocol (R2PC) are discussed, which can increase update success ratio of replication transactions through relaxing the commit constraints, and enhance DB-BEs' availability using transaction retry and discard mechanisms. Then, a novel method using request logs is suggested for data resynchronization. The method only records the missing transaction requests for the unavailable DB-BEs in the DB-FEs and links the logs belonging to a site together, to reduce the resynchronization time. Moreover, the method doesn't interrupt the normal transaction processing in the DB-BEs, so system throughput using request logs is higher than using transaction logs. Experiences validate that the suggested methods have better performance.**

***Index Terms*—synchronization replication, data resynchronization, heterogeneous databases, middleware-based architecture**

## I. INTRODUCTION

With the growing numbers of subscribers and the introduction of new additional services, a large of web applications and telecommunication applications such as home location register (HLR) and home subscriber service (HSS), are required to be separation with database layer and high availability. A general solution is to employ a middleware-based replication architecture shown as Figure 1[1~5].

The architecture consists of three tiers:

- The application client (AC) tier, which initiates requests from clients. An AC may connect to one or more DB-FEs for better performance, e.g. load balancing and connection redundancy.
- The database front end (DB-FE) tier, also called middleware tier, which is responsible of receiving the requests from ACs, routing the requests to DB-BEs, and passing on the results, either a success or failure, back to the ACs.
- The database backend (DB-BE) tier, which holds the actual user/subscriber data, executes the transaction requests from BE-FEs, and returns results/responses back to the DB-FEs. For guaranteeing high availability of databases, N (N≥3, Figure 1 shows a case of N=3) DB-BEs are organized into a cluster for data redundancy, which could be heterogeneous and co-located or spread out to a wider geographic region.

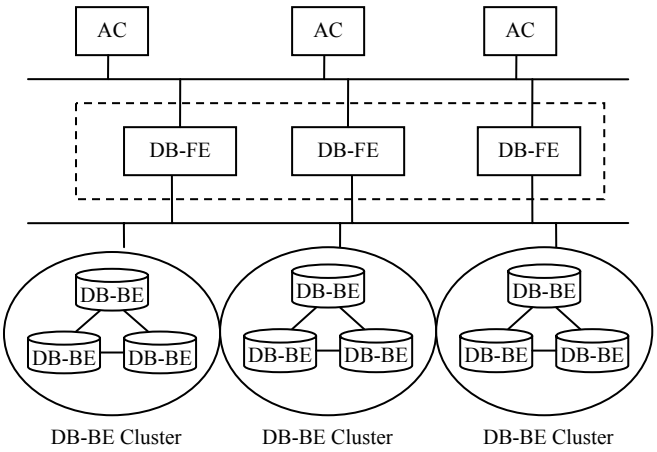



Figure 1. The middleware-based replication architecture for high availability database applications

It is assumed that these elements, including the ACs, DB-FEs and DB-BEs, are all connected to form a network and they are supported by some network protocols, e.g. Ethernet, TCP/IP, ATM, or etc. Some characteristics of such a network are:

- Redundant connection. There are redundant physical connections between ACs and DB-FEs, DB-FEs and DB-BEs. With redundant connections, any single point of failure will not have impact to the overall system service availability.
- Replicated Data. All data objects in the databases are fully replicated in N DB-BEs among one cluster to make data continuously available to the DB-FEs in the event of a component failure, which could be due to a hardware node, software or a connection failure. Whenever an object in the database is to be updated, the update has to be




doi:10.4304/jnw.7.1.210-217

replicated in all of the N instances to ensure a consistent database view, i.e., synchronous replication.

The purpose of synchronous replication is to guarantee all replicas of every data object in a cluster always have the same image. But one problem of synchronous replication is it may lead to an increased response time due to waiting for the completion of all the replicas' updates. Another problem is an update request has to be rejected if one replica can't be updated successfully, to ensure data consistency across all replicas. The increased level of update failure will have negative impact to the high availability concept for such architecture.

In the other hand, there will be instances of network or other failures that stop data from being replicated to all replicas. When this occurs, the replicas on the failed servers will become out of synchronization. Therefore, a resynchronization process must take place to ensure the failed servers can catch up of the missed updates. How to "catch up" the missed updates but without interrupting the services in the active servers is also a challenge for this architecture.

This paper mainly focuses on data synchronous and data resynchronization in the case of failures in the middleware-based replication architecture. The remaining parts of the paper are organized as following. Section 2 is the introduction of related works. Section 3 gives a synchronous update everywhere replication model and a commit framework for the architecture. Section 4 gives some rules and mechanisms for replication transactions. Section 5 describes a replication 2-phase commit protocol, including commit state graphs and algorithms both for the coordinator and participants. Section 6 suggests a novel resynchronization method using request logs. Section 7 is performance testing and evaluation. Section 8 concludes the paper.

## I. RELATED WORKS

In general, database replication protocols can be categorized into two classes by the time when update propagation to databases takes place, eager vs. lazy[6]. In eager replication schemes, i.e. synchronous replication, updates are propagated within the boundaries of a transaction. In this scheme, the user does not receive the commit notification until sufficient copies in the system have been updated. Lazy replication schemes, i.e. asynchronous replication, on the other hand, update a local copy, commit and only some time after the commit, the propagation of the changes takes place. The first approach provides consistency in a straightforward way but it is expensive in terms of message overhead and response time. Lazy replication allows a wide variety of optimizations, however, since copies are allowed to diverge, inconsistencies might occur.

In the other hand, by the dimension of where the updates can take place, replication protocols can also be classified into primary copy (master) and update everywhere (group) approaches[7]. The primary copy approach requires all updates to be performed first at one copy (the primary or master copy) at a master server and then at the other copies (the secondary copies) at secondary servers. This simplifies replica control at the price of introducing a single point of failure and a potential bottleneck. The update everywhere approach allows any copy to be updated, thereby speeding up access but at the price of making coordination more complex.

In most present industrial products, one standard solution for synchronous replication is the Two-Phase Commit (2PC) protocol such as XA model[8], which makes global decision by following rules:

- all vote "YES" $\Rightarrow$ global commit
- one votes "NO" $\Rightarrow$ global abort

But for replication transactions, the rule of making global commit is too strict since it requires all replicas can be updated successfully, and the rule of making global abort is too relaxed since it will lead to aborting a lot of transactions due to a single site's failure. Moreover, an update transaction can't be committed if one replica always say "NO". Therefore, the rules are inconsistent with the requirement of high availability.

In addition, as long as one site votes "NO", the transaction will be aborted regardless of the reason of voting "NO". In general, if a replication transaction can commit in one site, it should also execute successfully in all other available sites if there is no site failures and link failures. However, it is possible for a replication transaction that one site votes "YES" while another site votes "NO" due to some temporal software or hardware failures, like temporary memory shortage, deadlock, or software bug, etc. In this case, the whole transaction will also be aborted, which will lead to wasting a lot of system resources.

There are many 2PC-related protocols, like presumed commit (PC), presumed abort (PA, three-phase commit (3PC)[9], and etc, which are designed to enhance the performance of 2PC. Although 3PC eliminate the blocking in 2PC, all of them don't give any special treatment for replication transactions.

Now almost all databases replication products, including disk-resident databases (such as Oracle, DB2, Sybase, etc.) and In-Memory Database (IMDB) (such as Timesten), use transaction logs to handle resynchronization. That is, when receiving a resynchronization request from a failed server, the active DB-BE will send the logs of the missed committed transactions (called resynchronization logs) to it. Then the failed server can replay the missed updates by the logs.

Since it is necessary for each transaction to write its logs to a stable log file before it commits, no additional log overhead needs for resynchronization in the method based on transaction logs. But the method has some disadvantages for resynchronization [10-12]:

- As one replication server has lost communication for a long time, a great amount of logs need be stored in the active DB-BEs, even if many checkpoint and backup operations have been completed.
- The transaction logs are essentially designed for recovery processing, to undo the uncommitted transactions and redo the committed transactions.

Since all transactions are recorded into the logs by appending data at the end of the log, the fetcher of the resynchronization logs in the active server will take time to distinguish which logs are required for resynchronization. Also the update process in the failed servers will spend time to analyze which part in a log is the after image of the missed data.

- The procedure of resynchronization may interrupt the normal transaction processing in the active server, since the transaction logs must be accessed exclusively by both the transaction manager (log write) and the fetcher of the resynchronization logs (log read) in the active server.
- The methods can't be extended to the replication environment with heterogeneous databases, since heterogeneous databases have different log formats.

So far as we know, no any special resynchronization solution has been suggested for the three-tier architecture.

## II. REPLICATION MODEL IN THE MIDDLEWARE-BASED ARCHITECTURE

### A. Synchronous update everywhere replication model

In the middleware-based architecture, the middleware (DB-FE) tier acts as an important role for all replication transactions. When a DB-FE receives a request from an AC, it will determine which cluster the request should be forwarded according to data distributed information, and then select one or more DB-BEs in the cluster to process the request.

We use synchronous update everywhere replication model for the architecture since it needs less response time than the primary copy model (Figure 2). In the model, a replication procedure can be divided into five phases:

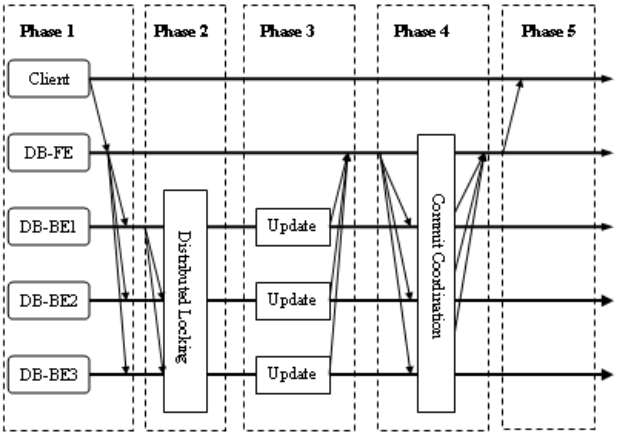



Figure 2. Synchronous update everywhere replication model

**Phase 1** (Request forwarding): After receiving an update request from a Client, the DB-FE will forward the request to all available DB-BEs in the specified cluster.

**Phase 2** (Lock Coordination): One of the database servers sends a lock request to all other servers for asking whether to grant the lock or not. If the lock can be granted by all sites, the transaction can proceed. If not, the transaction will be delayed till all sites agree to grant the lock.

**Phase 3** (Execution): When all the locks are granted, the operation will be executed at all sites.

**Phase 4** (Commit Coordination): During the commit coordination phase, a commit protocol is used to make sure that all sites can commit the transaction. The commit coordinator is acted by DB-FE.

**Phase 5** (Client Response): When the transaction is completed, the DB-FE returns a response, either success or failure to the Client.

In this model, a DB-FE can forward transaction requests by their type:

- If the request is a read request, the DB-FE will forward it to one of the available DB-BEs in the cluster by the load balance strategy used.
- If the request is an update request, the DB-FE will forward it to all available DB-BEs in the cluster, instead of a master server.

Therefore, updates on all replicas of a data object in the model are done almost simultaneously, so it can reduce the response time largely. However, in order to ensure the consistency of all copies, distributed locking approaches should be used to resolve data access conflicts, i.e., an object can only be accessed after all of its replicas have been locked. Therefore, for each update transaction, a lock coordinator phase should be added, which will increase transaction overhead.

### B. Middleware-based Commit Framework

In the model mentioned above, a new kind of commit protocol should be designed to synchronize all N replicas in a cluster. The DB-FE in the middleware tier will act as a coordinator, and all DB-BEs in the cluster are the participants of the replication transactions. The commit framework for the middleware-based replication is shown as Figure 3.

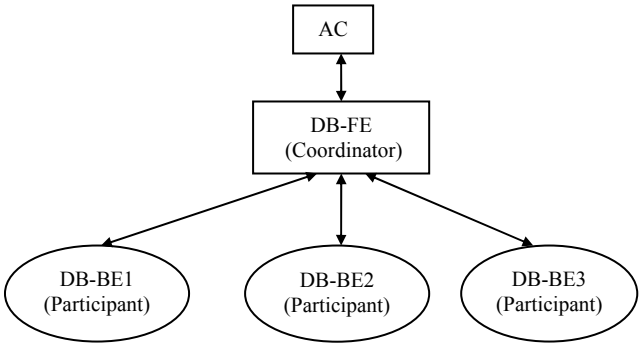



Figure 3. The commit framework for the middleware-based replication

In this framework, before a DB-FE forwards a request, it should know the up-to-date state of all DB-BEs. Therefore each DB-FE should keep heartbeats with all DB-BEs. In the view of DB-FEs, a DB-BE should be in one of the following states at any time (Figure 4):

- *DOWN* state indicates the connection between the DB-FE and the DB-BE is broken, i.e., without any heartbeat message.

- *OUT-OF-SYNC* state indicates the connection between the DB-FE and the DB-BE is linked but the data in the DB-BE is out of synchronization. When a DB-BE is in "OUT-OF-SYNC" state, it can't become "SYNC" till catching up with all the missed updates through a resynchronization process.
- SYNC state indicates the data in the DB-BE is consistent with other DB-BEs in the same cluster.

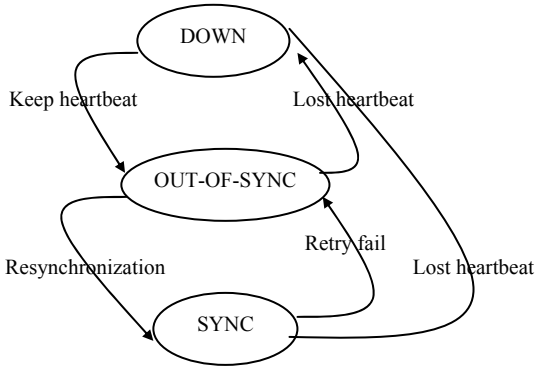



Figure 4. State graph of DB-BEs

## III. RULES AND MECHANISM FOR REPLICATION TRANSACTIONS

In order to ensure high availability of the replication databases, we should design a new replication commit protocol for the middleware-based replication transactions with synchronous requirements.

### *A. Base commit rules for replication transactions*

It has been discussed in Part I that the commit rules of 2PC are not suitable for the synchronous replication transactions. First, two base commit rules for replication transactions are suggested as following:

*RR1. If one participant votes "YES", the coordinator can make a global commit decision.*

*RR2. Only if all participants vote "NO", the coordinator can make a global abort decision.*

According to the new rules, a replication transaction may commit as long as one site votes "YES", and whether it can abort must wait for a global abort decision. This is just opposite to the standard 2PC, which allows a site to abort unilaterally, and it must wait for a global commit decision after voting "YES".

However, although the participants are allowed to commit unilaterally in the end, it can't commit till receiving a global commit decision, to ensure all replicas can be updated synchronously.

*RR3. The participant who votes "YES" can commit only receiving a global commit decision from the coordinator.*

### *B. Transaction retry mechanism*

In order to reduce the probability of transaction abort due to temporary failures, if at least one participant has voted "YES", the participant who has voted "NO" should retry.

Figure 5 is an example of retry mechanism. Both DB-BE1 and DB-BE2 voted "YES", and DB-BE3 voted "NO". The coordinator should send a retry message to DB-BE3. When C receives a retry message, it will retry the transaction till timeout or success to revote "YES". No matter what DB-BE3 votes, the coordinator can make a global commit decision and send the decision to both DB-BE1 and DB-BE2. The additional work is, if DB-BE3 cannot success till timeout, the coordinator will indicate its state as "OUT-OF-SYNC", and does not forward any new request to it until it has gotten the missed updates. Therefore, DB-BE1 and DB-BE2 can keep database services unceasingly, and the inconsistent replica in DB-BE3 will not be seen by applications.

Therefore, we have following rule:

*RR4. If there is any participant voting "YES", the participants voting "NO" should retry till timeout or success to revote "YES".*

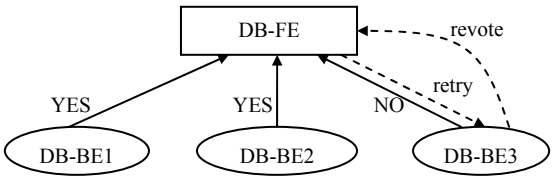



Figure 5. Transaction retry mechanism

### *C. Transaction discarding mechanism*

In replication environment, two kinds of DB-BEs should become unavailable, i.e., the sites in "DOWN" state or in "OUT-OF-SYN" state. For ensuring the applications can always see the consistent replicas of all data objects, the DB-FEs never forward requests to the unavailable DB-BEs. Therefore, in order to improve the availability, the DB-FEs can discard following participants in the decision phase:

- The DB-BEs become "DOWN" because of broken link.
- The DB-BEs vote "NO" or timeout after retrying.

As to the first kind of transactions, since their sites can't communicate with DB-FEs, the data objects in the sites cannot be accessed by DB-FEs. When the sites resume connection to DB-FEs, they will enter "OUT-OF-SYNC" state first, and then do resynchronization to get the missed updates, which will be discussed in Part V.

As to the second kind of transactions, when they are discarded, the states of their sites will be marked as "OUT-OF-SYNC" at once, and the DB-FE doesn't forward any new request to it until it becomes "SYNC".

In Figure 6, DB-BE3 is in "DOWN" state. Since the coordinator cannot receive any vote from DB-BE3 as timeout, it can decide to commit the transactions on DB-BE1 and DB-BE2, and discard the replication transaction in DB-BE3. But it should identify the state of DB-BE3 as "DOWN", and does not forward any read or write requests to it.

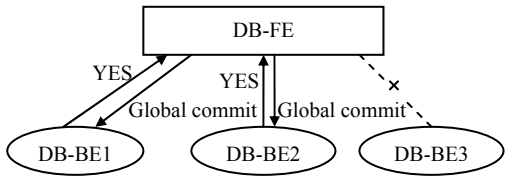



Figure 6. Transaction discard mechanism

*RR5. If the coordinator finds a participant in "DOWN" state during commit processing, the participant will be discarded.*

*RR6. If there is at lease a participant who votes "YES" and a participant who votes "NO", the coordinator will discard the participants voting "NO", and denote their states as "OUT-OF-SYNC".*

## IV. REPLICATIOIN TWO-PHASE COMMIT PROTOCOL (R2PC)

A two-phase commit protocol integrating the features discussed in Part VI is called a replication two-phase commit protocol (R2PC), which can also be divided into two phases:

- Vote phase: the participants vote their execution results "YES" or "No" to the coordinator.
- Decision phase: the coordinator make a global commit or abort decision and forward it to the participants

### *A. Coordinator's state transition diagram and commit algorithm*

Figure 7 is the state transition diagram for the coordinator, including INITIAL, WAIT, COMMIT, and ABORT states. After a coordinator send "prepare" message, it enters a WAIT state. Whether it enters COMMIT or ABORT state is determined by the votes it has received.

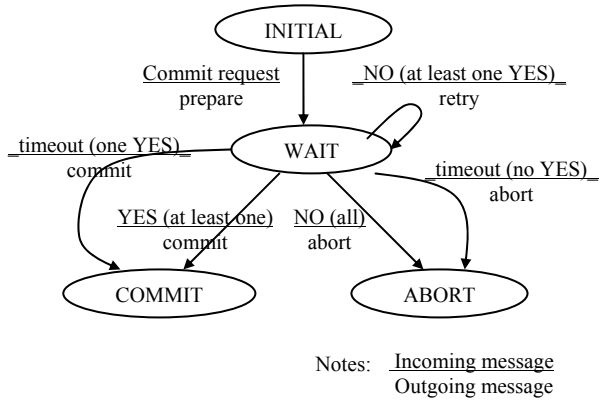



Figure 7. Coordinator 's state transition diagram

The commit algorithm of the coordinator can be described as follows.

*BEGIN*
*Pa:={all available sites};*
*if Pa≠Null {(*
*initiates a global transaction;*
*writes START to global log and flush;*
*sends transaction request to the sites in Pa;*
*wait until receives commit request from AC;*
*sends "PREPARE" request to DB-BEs;*
*while NOT receive all "YES" votes OR not timeout {*
*if has received a "YES" vote then*
*if receives a "NO" vote then*
*send RETRY message to it*
*}*
*Py:={participants voting "YES"};*
*Pn:={participants voting "NO"};*
*if Py=Pa { // all sites vote "YES"*
*write COMMIT to global log and flush;*
*send global commit to the participants in Py;*
*}*
*else {*
*if Pn = Pa { // all sites vote "NO"*
*write ABORT to global log and flush;*
*send global abort to the participants in Pn;*
*}*
*else {*
*send global commit to the participants in Py;*
*identify the states of the sites in Pn as"OUT-OF-SYNC";*
*identify the states of the sites not in Pn and Pa as "DOWN"*
*}*
*write DONE to global log and flush;*
*}*
*END*

### *B. Participant's state transition diagram and commit algorithm*

Figure 8 is the state transition diagram for the participants, which includes INITIAL, WAIT, READY, and ABORT states. After a participant votes commit, it enters READY state, and if it votes "NO" or timeout, it enters WAIT state. The WAIT state can be changed to READY state if the participant retry successfully. Participants will take different actions as timeout in different states.

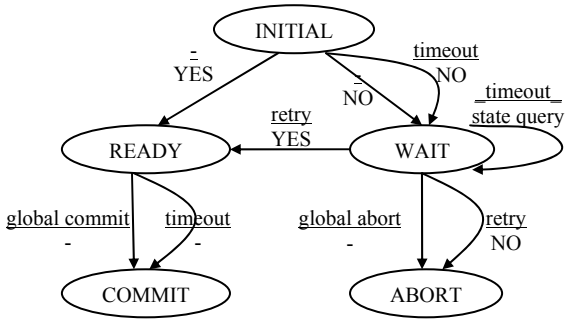



Figure 8. Participant's state transition diagram

The commit algorithm of the participants can be described as follows.

*BEGIN*

```
  RETRY_FLAG=0;
  start transaction;
  if receives PREPARE request {
   RP: if executes successfully {
         writes PREPARE COMMIT log and flush;
        votes "YES" to coordinator;
        waits till receive the global decision or timeout;
        writes COMMIT to local log and flush;
        commits transaction;
        return
       }
       else {
        if RETRY_FLAG=0 {
           writes PREPARE ABORT  log and flush;
           votes "NO" to coordinator;
           wait till receive the global decision or timeout;
           if receive RETRY command {
              RETRY_FLAG=1;
              re-execute the transaction;
              goto RP;
           }
           if receive global abort decision {
             write ABORT to local log and flush;
             abort transaction
           }
       }
   }
END
```

## V. RESYNCHRONIZATION METHOD BASED ON REQUEST LOGS

The purpose of resynchronization procedure is to get the missed updates before the failed system resumes services. If a replication transaction can execute successfully on the replicas over all sites, the update information of the transaction needn't be resynchronization since nobody missed it. Therefore, only the information of committed write transactions which are missed by some sites need be kept.

Instead of the transaction logs, request logs are suggested for the middle-based replication architecture to handle resynchronization, i.e., only the requests of the write replication transactions missed by failed sites should be recorded, such that resynchronization can be accomplished by re-executing the missed requests in the resynchronization sites.

### *A. Resynchronization procedure based on request logs*

The DB-FE is both a request forwarder and a transactions coordinator, so it knows about which requests have been forwarded, whether and when the requests have committed or aborted, and which requests are missed by the DB-BEs. Thus, whenever a DB-FE makes a global commit decision in the decision phase of R2PC, it will check whether the transaction is missed by any DB-BEs. If it is, the DB-FE will record the transaction request both into its memory and a stable storage.

The format of request logs should be as *<USN, TR, SBE>*. Of which, USN is the update sequence number, TR is the transaction request, SBE is the set of DB-BE missing the update. The requests belonging to a DB-BE can be linked together.

The resynchronization procedure based on request logs is shown as Figure 9:

(1) After the DB-FE receives a heartbeat message from an "OUT-OF-SYNC" DB-BE, it sends a resynchronization command to the DB-BE;

(2) The DB-BE starts a resynchronization procedure and send a ready message to the DB-FE;

(3) The DB-FE fetches all missing request logs of the DB-BE, and sends them to it;

(4) The DB-BE executes the missing request by the order recorded in the log;

(5) After the DB-BE finishes resynchronization, it sends an end resynchronization message to the DB-FE;

(6) The DB-FE identifies the state of the DB-BE as "SYNC".

(7) The DB-BE can resume normal service.

### *B. Resynchronization order of requests*

The replay order of the missed requests is very important for guaranteeing the correctness of resynchronization in the failed sites. It is not necessary for the failed DB-BEs to replay the missed requests strictly by the same order as they have committed at the active servers, but it must ensure all conflicted requests can be replayed with the same order as they have committed.

For example, say 3 missed transactions Ta, Tb and Tc, and Ta is conflicted with Tc. Their commit order in an active DB-BE is <Ta, Tb, Tc>. The replay orders both of <Ta, Tb, Tc> and <Tb, Ta, Tc> are said to be right for the failed DB-BE, since all of them can ensure Ta is replayed before Tc.

In fact, a DB-FE doesn't know the actual commit order of the transactions in the DB-BEs. But it knows the order of the global commits of all requests for it is the coordinator. If the replication two-phase locking (R2PC) is used to ensure the consistency of databases, the order of making global commit can be used as the replay order in the failed servers, since the conflicted transactions can't vote YES at the same time.

### *C. An example*

Figure 9 is an example of request logs, which keep the missed update records for both DB-BE2 (i.e., U001~U004) and DB-BE3 (i.e., U003 and U004). An update sequence number (USN) is the global commit sequence of a transaction request, which determines the replay order in the failed DB-BEs. Figure 10 shows the resynchronization orders of the missed requests for both DB-BE2 and DB-BE3.

```
<U001, R1, DB-BE2>
<U002, R3, DB-BE2>
<U003, R4, DB-BE2, DB-BE3>
<U004, R2, DB-BE2, DB-BE3>
```

Figure 9. An example of request logs

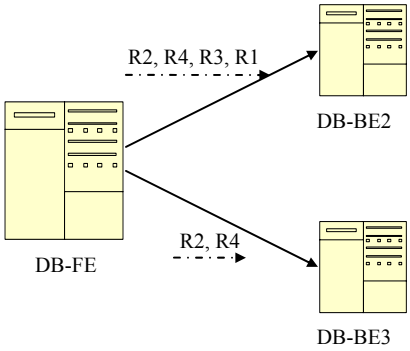



Figure 10. Resynchronization order of the missed requests

## VI. PERFORMANCE TESTING

In this section, the performances both of R2PC and the resynchronization method using request logs are tested.

For R2PC, we first compare it with the standard 2PC protocol on update success ratio. Then, we will test the impact of transaction retry mechanism in R2PC.

For the resynchronization method, we first compare it with the method using transaction logs on resynchronization time and transaction throughput of the system.

The performance evaluation is performed on a simulation replication environment, including 3 ACs, 2 DB-FEs, 2 DB-BE Clusters, which form a local network.

### *A. Experimental results*

Figure 11 shows the comparison results of update success ratio under various transaction generating ratio. For the commit condition is relaxed in R2PC, i.e., as long as one DB-BE votes "YES", then the replication transaction can commit. But in 2PC, only all DB-BEs in a cluster vote "YES", the replication transaction can commit. Therefore, the update success ratio of R2PC is higher than that of 2PC.

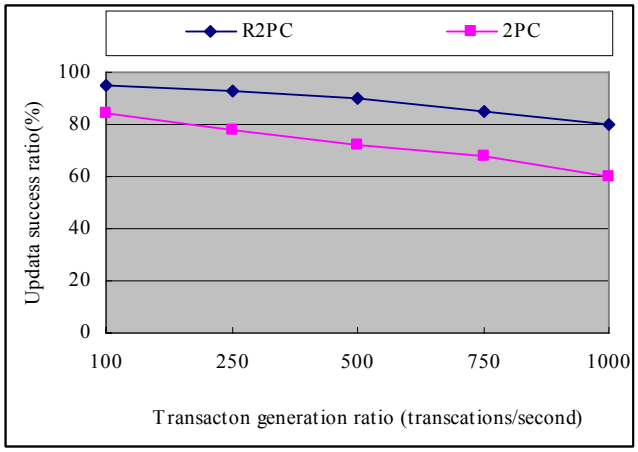



Figure 11. Comparison of update success ratio

Figure 12 shows the impact of retry mechanism on the availability ratio of DB-BEs. In R2PC, if there is a DB-BE votes "YES", the replication transaction can commit. But the sites voting "NO" or without voting are denoted as an unavailability state: "OUT-OF-SYNC" or "DOWN", which will reduce the availability of DB-BEs. In order to reduce the probability of unavailability due to temporal failures, R2PC allows the site voting "NO" re-executes the transaction if some one else has votes "YES". Therefore, the retry mechanism can increase the availability of DB-BEs.

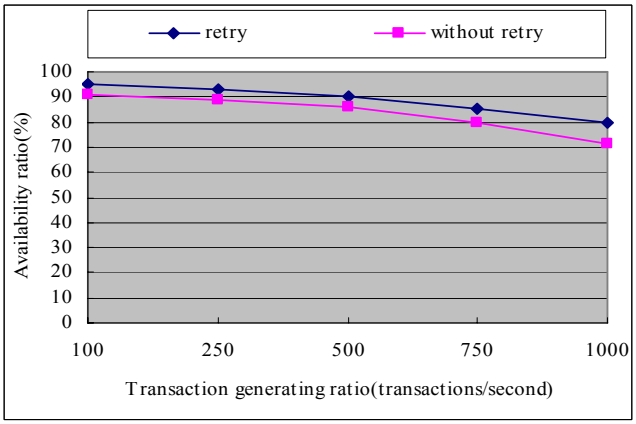



Figure 12. Comparison of DB-BEs' availability

Figure 13 shows the comparison of resynchronization time using different methods. For each replication transaction in the experience only updates one or 2 data objects, so they are generally short transactions. That is, the times of resynchronization operations using both the transaction logs and the request logs are almost the same. But since all transactions logs are append-only logs, the fetcher of in the active server will take time to distinguish which logs are required for resynchronization, and the update process in the failed servers have to spend time to analyze which part in a log is the after image of the missing data. In the method using request logs, all logs belonging to one site are linked together, so it is easy for the DB-FEs to fetch the desired logs, and the format of the request logs is uniform. Therefore, it can save the resynchronization time.

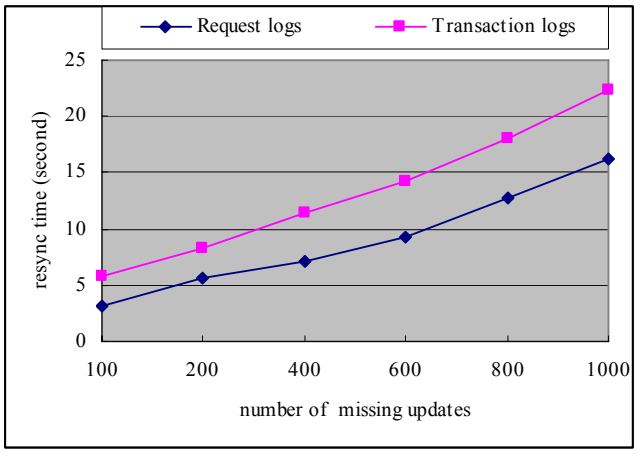



Figure 13. Comparison of resynchronization time

Figure 14 shows the impact on system throughput using different logs. The resynchronization procedure using transaction logs may interrupt the normal transaction processing in the active server, since the transaction logs must be accessed exclusively by both the

transaction manager (log write) and the fetcher of the resynchronization logs (log read) in the active server. But the resynchronization method using request logs doesn't interrupt normal transaction processing in the "SYNC" DB-BEs, so no any overhead needs be added for resynchronization processing in the "SYNC" DB-BEs.

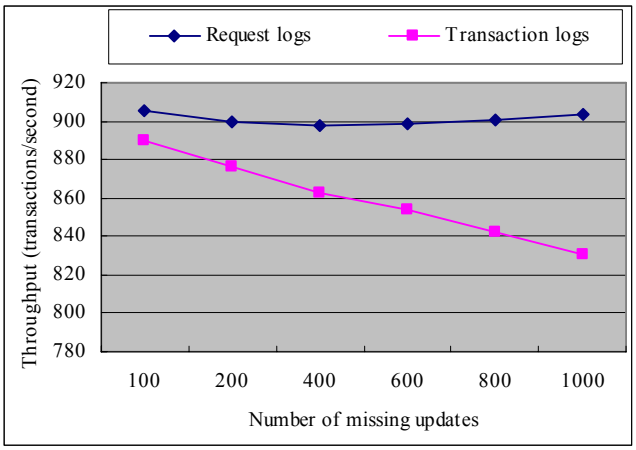



Figure 14. Comparison of transaction throughout

## VII. CONCLUSIONS

This paper focuses on the synchronous and resynchronization mechanism for the heterogeneous databases replication under a middleware-based architecture. The main advantages can be gained from the suggested methods as following:

- Because a replication transaction will be aborted only if all "SYNC" sites have voted "NO", the availability of systems is greatly enhanced.
- It always provides a consistent view to clients, since only the replicas in the "SYNC" sites can be accessed by clients.
- The number of transactions who are aborted due to the temporal failures can be reduced largely through the transaction retry mechanism.
- The suggested resynchronization method doesn't interrupt normal transaction processing in the "SYNC" DB-BEs.
- No any overhead is added for the "SYNC" DB-BEs during resynchronization procedure. In other words, the DB-BEs care nothing about resynchronization .
- The resynchronization procedure is independent of DB-BEs' failures. That is, even if all other DB-BEs in the cluster have failed, the "OUT-OF-SYNC" DB-BEs can still become "SYNC" after finishing the resynchronization with a DB-FE.

## ACKNOWLEDGMENT

This work is supported by National Natural Science Foundation of China (No.60863016), Natural Science Foundation of Jiangxi, China (No.2008GQS0019)

## REFERENCES

[1] J. Chen, G.Soundararajan, C.Amza, "Autonomic Provisioning of Backend Databases in Dynamic Content Web Servers", IEEE International Conference on Autonomic Computing (ICAC06), IEEE Press, June 2006, pp. 231- 242.

[2] Y. Lin, B. Kemme, M. Patiño-Martínez, etc., "Middleware Based Data Replication Providing Snapshot Isolation", ACM SIGMOD International Conference on Management of Data, San Jose, CA, USA, May 1995, pp. 419-430.

[3] J. M. Milan-Franco, R. Jiménez-Peris, M. Patiño-Martínez, etc., "Adaptive middleware for data replication", The 5th ACM/IFIP/USENIX international Conference on Middleware, Toronto, Canada, October 2004, pp. 175-194.

[4] C. Plattner, G. Alonso, "Ganymed: Scalable Replication for Transactional Web Applications", The 5th ACM/IFIP/USENIX international Conference on Middleware, Toronto, Canada, October 2004. pp. 155 – 174.

[5] M. Patiño-Martinez, R. Jiménez-Peris, B. Kemme, etc., "MIDDLE-R: Consistent database replication at the middleware level", ACM Transactions on Computer Systems (TOCS), Nov. 2005, Volume 23, Issue 4, pp. 375-423.

[6] K. Daudjee, K. Salem, "Lazy Database Replication with Snapshot Isolation", The 32nd International Conference on Very Large Data Bases (VLDB), Seoul, Korea, September 2006, pp. 715–726

[7] F. D. Munoz, H. Decker, J. E. Armendariz, etc., "Database Replication Approaches", Institute Tecnology of Information, Technical Report TR-ITI-ITE-07/19, October 5, 2007

[8] "XA in high-availability clusters", IBM Informix Dynamic Server (IDS), Version 11.50, http://publib.boulder.ibm.com/infocenter/idshelp/v115/index.jsp?topic=/com.ibm.admin.doc/ids_admin_1323.htm

[9] P. A. Bernstein, V. Hadzilacos, N. Goodman, Concurrency control and recovery in database system. AddisonWesley, Reading, MA, USA, 1987.

[10] E. Cecchet, , G. Candea, A. Ailamaki," Middleware-based database replication: the gaps between theory and practice",. ACM SIGMOD International Conference on Management of Data, Vancouver, Canada, Mar. 2008, pp. 739–752.

[11] M. Wiesmann, A. Schiper, "Comparison of database replication techniques based on total order broadcast", IEEE Transaction. on Knowledge and Data Engineering, April 2005, Vol. 17(4), pp. 551–566.

[12] D. Francesc Munoz-Escoı, Jeronimo Pla-Civera, Marıa Idoia Ruiz-Fuertes,etc., "Managing transaction conflicts in middleware-based database replication architectures", Symposium on Reliable Distributed Systems, Dec. 2006, pp. 401–410,

Guoqiong Liao, born on 1969, received his Ph. D in Computer Software and Theory from Huazhong University of Science and Technology (HUST), Wuhan City, China in 2003.

He worked in HUST for post-doctor research till 2005. Then, he worked as a Research Scientist in Siemens Corporate Technology in China. Now he is an associate Professor in the School of Information Technology, Jiangxi University of Finance and Economics, Nanchang, China. He has published a number of papers in the area on transaction processing and high availability on real-time and mobile databases. His research interests involved embedded, real-time and mobile databases, and more recently on RFID data management and middleware.

# Call for Papers and Special Issues

## Aims and Scope.

Journal of Networks (JNW, ISSN 1796-2056) is a scholarly peer-reviewed international scientific journal published monthly, focusing on theories, methods, and applications in networks. It provide a high profile, leading edge forum for academic researchers, industrial professionals, engineers, consultants, managers, educators and policy makers working in the field to contribute and disseminate innovative new work on networks.

The Journal of Networks reflects the multidisciplinary nature of communications networks. It is committed to the timely publication of high-quality papers that advance the state-of-the-art and practical applications of communication networks. Both theoretical research contributions (presenting new techniques, concepts, or analyses) and applied contributions (reporting on experiences and experiments with actual systems) and tutorial expositions of permanent reference value are published. The topics covered by this journal include, but not limited to, the following topics:

- Network Technologies, Services and Applications, Network Operations and Management, Network Architecture and Design
- Next Generation Networks, Next Generation Mobile Networks
- Communication Protocols and Theory, Signal Processing for Communications, Formal Methods in Communication Protocols
- Multimedia Communications, Communications QoS
- Information, Communications and Network Security, Reliability and Performance Modeling
- Network Access, Error Recovery, Routing, Congestion, and Flow Control
- BAN, PAN, LAN, MAN, WAN, Internet, Network Interconnections, Broadband and Very High Rate Networks,
- Wireless Communications & Networking, Bluetooth, IrDA, RFID, WLAN, WMAX, 3G, Wireless Ad Hoc and Sensor Networks
- Data Networks and Telephone Networks, Optical Systems and Networks, Satellite and Space Communications

## Special Issue Guidelines

Special issues feature specifically aimed and targeted topics of interest contributed by authors responding to a particular Call for Papers or by invitation, edited by guest editor(s). We encourage you to submit proposals for creating special issues in areas that are of interest to the Journal. Preference will be given to proposals that cover some unique aspect of the technology and ones that include subjects that are timely and useful to the readers of the Journal. A Special Issue is typically made of 10 to 15 papers, with each paper 8 to 12 pages of length.

The following information should be included as part of the proposal:

- Proposed title for the Special Issue
- Description of the topic area to be focused upon and justification
- Review process for the selection and rejection of papers.
- Name, contact, position, affiliation, and biography of the Guest Editor(s)
- List of potential reviewers
- Potential authors to the issue
- Tentative time-table for the call for papers and reviews

If a proposal is accepted, the guest editor will be responsible for:

- Preparing the “Call for Papers” to be included on the Journal’s Web site.
- Distribution of the Call for Papers broadly to various mailing lists and sites.
- Getting submissions, arranging review process, making decisions, and carrying out all correspondence with the authors. Authors should be informed the Instructions for Authors.
- Providing us the completed and approved final versions of the papers formatted in the Journal’s style, together with all authors’ contact information.
- Writing a one- or two-page introductory editorial to be published in the Special Issue.

## Special Issue for a Conference/Workshop

A special issue for a Conference/Workshop is usually released in association with the committee members of the Conference/Workshop like general chairs and/or program chairs who are appointed as the Guest Editors of the Special Issue. Special Issue for a Conference/Workshop is typically made of 10 to 15 papers, with each paper 8 to 12 pages of length.

Guest Editors are involved in the following steps in guest-editing a Special Issue based on a Conference/Workshop:

- Selecting a Title for the Special Issue, e.g. “Special Issue: Selected Best Papers of XYZ Conference”.
- Sending us a formal “Letter of Intent” for the Special Issue.
- Creating a “Call for Papers” for the Special Issue, posting it on the conference web site, and publicizing it to the conference attendees. Information about the Journal and Academy Publisher can be included in the Call for Papers.
- Establishing criteria for paper selection/rejections. The papers can be nominated based on multiple criteria, e.g. rank in review process plus the evaluation from the Session Chairs and the feedback from the Conference attendees.
- Selecting and inviting submissions, arranging review process, making decisions, and carrying out all correspondence with the authors. Authors should be informed the Author Instructions. Usually, the Proceedings manuscripts should be expanded and enhanced.
- Providing us the completed and approved final versions of the papers formatted in the Journal’s style, together with all authors’ contact information.
- Writing a one- or two-page introductory editorial to be published in the Special Issue.

More information is available on the web site at http://www.academypublisher.com/jnw/.

*(Contents Continued from Back Cover)*