Contents

Special Issue: Performance Evaluation of Communication Networks and Systems
Guest Editors: Mohammad S. Obaidat and José Luis (Sevi) Sevillano

Guest Editorial
Mohammad S. Obaidat and José Luis (Sevi) Sevillano

SPECIAL ISSUE PAPERS

Analysis of a Buffered Optical Switch with General Interarrival Times
Conor McArdle, Daniele Tafani, and Liam P. Barry

Comparison of the Design Characteristics of MMI Wavelength Demultiplexers Using Different Approaches by Computing the Effective Index
M. Najjar, R. Rejeb, H. Rezig, and M. S. Obaidat

Design and Performance Evaluation of Service Overlay Networks Topologies
Davide Adami, Christian Callegari, Stefano Giordano, Gianfranco Nencioni, and Michele Pagano

An Empirical Evaluation of Multi-Step Prediction Performance
Mohamed Faten Zhani, Halima Elbiaze, and Farouk Kamoun

A Visualization Tool for Exploring Multi-scale Network Traffic Anomalies
Romain Fontugne, Toshio Hirotsu, and Kensuke Fukuda

Interference Reduction in Overlaid WCDMA and TDMA Systems
Maan A. S. Al-Advany and Amin M. Abbosh

GCAD: A Novel Call Admission Control Algorithm in IEEE 802.16 based Wireless Mesh Networks
Floriano De Rango, Andrea Malfitano, and Salvatore Marano

REGULAR PAPERS

A Secure Protocol for Sharing Trust Data in Hybrid P2P Network
Huaiqing Lin and Yonghong Zhou

Mobile Platform and Secure Access Approach of UMTS Terminal Based on Trusted Computing
Yingyou Wen, Xuena Peng, Shuyi Chen, and Hong Zhao

A Network Architecture for Load Balancing of Heterogeneous Wireless Networks
Wenxiao Shi, Bin Li, Na Li, and Chuanjun Xia

An Identity Based Aggregate Signature from Pairings
Yike Yu, Xuefeng Zheng, and Hua Sun
<table>
<thead>
<tr>
<th>Title</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Detecting Malware Variants by Byte Frequency</td>
<td>638</td>
</tr>
<tr>
<td>Sheng Yu, Shijie Zhou, Leyuan Liu, Rui Yang, and Jiaqing Luo</td>
<td></td>
</tr>
<tr>
<td>A Novel Timing and Frequency Synchronization Technology for OFDM System</td>
<td>646</td>
</tr>
<tr>
<td>Yi Sha, Mingming Li, and Jiafu Chu</td>
<td></td>
</tr>
<tr>
<td>Analysis on Stability of a Network Based on RED Scheme</td>
<td>654</td>
</tr>
<tr>
<td>Shengbo Hu, Heng Shu, and Tiancheng Lin</td>
<td></td>
</tr>
<tr>
<td>Analysis and Improvement of the BAN Modified Andrew Secure RPC Protocol</td>
<td>662</td>
</tr>
<tr>
<td>Weibo Liu, Wenping Ma, and Yuanyuan Yang</td>
<td></td>
</tr>
<tr>
<td>Proactive Location Service in Mobility Management</td>
<td>670</td>
</tr>
<tr>
<td>Yu Liu, Qiang Shen, ZhiJun Zhao, and Hui Tang</td>
<td></td>
</tr>
<tr>
<td>Soft-Handoff in WLAN Realized by Dual Link</td>
<td>678</td>
</tr>
<tr>
<td>Hong-yan Qian, Wei-xin Cai, and Xiao-lin Qin</td>
<td></td>
</tr>
</tbody>
</table>
This special issue of the Journal of Networks includes mainly extended versions of selected papers accepted and presented at the 2009 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS 2009). Only papers with excellent review scores were invited to submit extended versions to this Special Issue, which have undergone a second review process. The selected papers address a variety of topics related to the performance evaluation of communication networks and systems.

In the first paper, “Analysis of a Buffered Optical Switch with General Interarrival Times”, by Conor McArdle, Daniele Tafani and Liam P. Barry, the authors developed a relatively simple model for the analysis of an OBS (Optical Burst Switching) node with FDLs (Fiber Delay Lines) by applying circuit switching analysis methods. The model may be used to evaluate performance under any general independent (GI) traffic stream that may be expressed in terms of the LST (Laplace-Stieltjes transform) of the interarrival distribution. The analysis is formulated in terms of virtual traffic flows within the optical switch from which expressions for burst blocking probability, fiber delay line occupancy and mean delay are derived.

The second paper, “Comparison of the Design Characteristics of MMI Wavelength Demultiplexers Using Different Approaches by Computing the Effective Index”, authored by M. Najjar, R. Rejeb, H. Rezig and M. S. Obaidat, investigates the design characteristics of a wavelength demultiplexer using the effective index method in order to determine the propagation constants and field distributions. In order to analyze the impact of computation method on the design characteristics of the demultiplexer, the effective index is computed by an analytic approach. A new algorithm for developing an analytic model is proposed.

In the third paper, “Design and Performance Evaluation of Service Overlay Networks Topologies”, authored by Davide Adami, Christian Callegari, Stefano Giordano, Gianfranco Nencioni and Michele Pagano, the topology design problem of a Service Overlay Network is addressed from a performance point of view, taking into account both the traffic demand and the overhead. The performance of a limited set of well-known topologies is investigated. Additionally, three new traffic demand-aware overlay topologies are proposed based on heuristics, which are also evaluated through extensive simulations.

In the fourth paper, “An empirical Evaluation of Multi-Step Prediction Performance”, by Mohamed Faten Zhani, Halima Elbiaze and Farouk Kamoun. In this paper, the authors perform an analysis of Multi-step Internet traffic prediction performance of the ARIMA (AutoRegressive Integrated Moving Average) and the LMMSE (linear minimum mean square error) models. Two multi-step prediction techniques are compared: the Iterating Multi-Step Technique (IMS) and the Direct Multi-Step Technique (DMS). The analysis is based on two sets of real Internet measurements.

In the fifth paper, “A Visualization Tool for Exploring Multi-scale Network Traffic Anomalies”, by Romain Fontugne, Toshio Hirotsu and Kensuke Fukuda, an interactive tool is presented that takes advantage of several graphical representations highlighting the different aspects of network traffic and anomalies. The proposed tool, available at http://www.fukuda-lab.org/~romain/mulot/, allows for exploration of network traffic at any temporal and spatial (address and port) scales. The usefulness of this tool is verified by evaluating it using several traffic traces.

In the sixth paper, “Interference Reduction in Overlaid WCDMA and TDMA Systems”, by Maan A. S. Al-Adwany and Amin M. Abbosh, evaluates the performance of WCDMA uplink system for UMTS mobile communications and investigate the possibility of increasing mobile communication cell capacity through merging WCDMA and TDMA systems in one cell. An interference canceller is proposed to reduce, or even completely cancel, the interference between WCDMA and TDMA, hence enabling them to work together. This coexistence is proven to be possible via computer simulations.
Finally, the seventh paper is: “GCAD: A novel Call Admission Control algorithm in IEEE 802.16 based Wireless Mesh Networks”, by Floriano De Rango, Andrea Malfitano and Salvatore Marano. In this paper, a new call admission control algorithm for 802.16 distributed mesh networks named GCAD-CAC (Greedy Choice with Bandwidth Availability aware Defragmentation) is proposed. The algorithm is characterized by an initial greedy choice, by a preemption and a defragmentation process. The performance of the proposed GCAD algorithm is evaluated in terms of throughput, average end-to-end delay, average delay jitter, number of refused requests and packet loss percentage. GCAD is also compared with another two CAC algorithms in literature, giving the best performance due to the presence of a defragmentation process.

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 evaluation of communication networks and systems.

GUEST EDITORS’ BIOGRAPHY

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 GLOBECOM 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 SCS Modeling & Simulation Newsletter. 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: ACS/IEEE AICCSA 2009, DCNET 2010, SPECTS-2009 and SPECTS-2010 (as Program Co-Chair), SPECTS 2008 (as Program Vice-Chair) as well as in the TPC of many conferences like ACS/IEEE AICCSA 2007, IEEE ICC 2008, IEEE Globecom 2008 and 2010, DSAI 2009 and 2010, etc. 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.
Analysis of a Buffered Optical Switch with General Interarrival Times

Conor McArdle, Daniele Tafani and Liam P. Barry
Research Institute for Networks and Communications Engineering, School of Electronic Engineering, Dublin City University, Ireland.
Email: {mcardlec, tafanid, barryl}@eeng.dcu.ie

Abstract—Optical buffering is known to significantly improve the performance of optical packet and burst switched networks and a number of useful analytic models for the case of Poisson traffic have been proposed previously. In this paper, we propose an approximate analytic model for generally distributed arrivals, specifically treating Gamma-distributed interarrival times, and show that the variance of the traffic has a significant impact on performance. The analysis is formulated in terms of virtual traffic flows within the optical switch from which we derive expressions for burst blocking probability, fibre delay line occupancy and mean delay. Emphasis is on approximations, by way of moment-matching techniques, that give good numerical efficiency so that the method can be useful for formulating dimensioning problems for large-scale networks. Numerical solution values from the proposed analysis method are compared with results from a discrete-event simulation of an optical burst switch.

Index Terms—optical switching, fibre delay lines, performance analysis, equivalent random method, gamma arrivals

I. INTRODUCTION

In recent years, considerable research effort has been focused on developing efficient Optical Burst Switching (OBS) and Optical Packet Switching (OPS) architectures and on performance improvements by way of contention-resolution schemes [1], [2] and optimised burst aggregation algorithms [3], [4]. Although the technologies are maturing to the extent that test-beds have been built [5]–[7] and it seems likely that OBS and OPS may be deployed in the medium-term, there remains a need to resolve pertinent network design, dimensioning and cost-optimisation challenges to enable network deployment. To this end, efficient analysis methods for OBS/OPS node and network performance evaluation are desirable and considerable attention is now focused there [8]. In particular, the analysis of wavelength conversion schemes and fibre delay lines (FDLs), as two of the main contention-resolution components of the switch, is receiving attention.

The addition of wavelength converters to optical switches reduces contention at output ports by enabling a packet arriving on one wavelength channel to be directed to an alternative free wavelength channel at the output. In performance evaluation studies, there may be assumed restrictions on the number of available wavelength converters and on the sharing strategy. Additionally, there may be restrictions on the range of conversion between one wavelength and another, due to limiting physical properties of the conversion devices [9], [10].

The addition of FDLs to the switch has also been shown to achieve a substantial reduction in packet loss (by orders of magnitude in some cases [11]) by selectively delaying packets in order to reduce contention for outgoing channels. Our focus in this paper is on the analysis of burst/packet loss and delay in OBS/OPS nodes with FDLs and unrestricted wavelength conversion. We develop an approximate model of switch performance, for general offered traffic, by applying circuit-switching analysis methods to model the switch output port and associated FDLs. Our goal is an efficient model that can accurately account for likely traffic characteristics within an OBS/OPS network so that the node model may be applied to modelling/dimensioning of large networks of optical switches.

A. Relation to Previous Work

There are several existing approaches to performance evaluation of optical nodes with buffering functionality implemented with FDLs. In [12], Callegati presents a framework for evaluating the blocking probability for asynchronous variable length bursts and models a single FDL as a queue with balking. A similar approach has been adopted by Lu & Mark in [11], where the overall system behaviour is characterised as a multi-dimensional continuous-time Markov chain. They develop an asymptotic approximation based on the $M/M/k$ queue with balking, when arriving bursts are long, and an $M/M/k/m$ queue for short bursts. An exact Markov chain analysis is also provided by Rogiest et al. [13] and an analysis for correlated arrivals is considered in [14]. In [15], Fan et al. model buffers as $M/M/k/m$ queues and provide bounds on the loss probabilities for classless and prioritised bursts. Gauger [16] investigates the influence of the...
combination of wavelength converters and FDL buffers in OBS, through simulation. The performance of several scenarios of feed-back and feed-forward FDL schemes are evaluated. The present authors approximately modelled feed-back FDLs, for Poisson offered traffic, in [17].

Previous work on performance evaluation of FDLs has largely assumed that burst interarrival times are exponentially distributed. Recently, Mountoukieu and Perros have studied burst aggregation algorithms at ingress nodes and propose that this assumption may not be accurate [18]. Burst interarrival times are shown to be, for example, general-Erlang distributed, depending on the burst aggregation method and the packet arrival process at the aggregator. As for burst length distribution, Gauger [19] has found from simulation that performance is relatively insensitive to burst length distribution. Rostami and Wolisz [20], through analysis, also show that burst length distribution has little impact on performance, concluding that assuming exponentially-distributed burst lengths is appropriate in analysis.

This previous work leads us to consider a modelling framework for generally-distributed arrivals and exponentially-distributed burst lengths and we base our analysis on the $GI/M/N/N$ loss system. We have chosen gamma-distributed interarrivals as a concrete case of a general independent ($GI$) traffic arrival process, as it allows a full range of interarrival-time variance to be explored in a single model. We note that other distributions could be handled in the framework, however most commonly used arrival models such as the Interrupted Poisson Process (IPP) or Erlang-$k$ can be parameterised for only a limited range of interarrival time variance.

The approach to modelling in the current paper is to identify virtual traffic flows, between the output channels and FDLs, modelling the switching node as a network of relatively simple queuing systems. This differs from previous work, as outlined above, which has focused mainly on direct evaluation of more complex single-queue systems. We make use of existing results for calculating overflow and carried traffic characteristics in loss systems, by way of Equivalent Random Theory (ERT) [21] and Brandt and Brandt’s work on the $GI/M/N/N$ system [22]. Our overall approach most closely relates to Reviriego et al. [10], where overflow analysis is applied to evaluate blocking, for Poisson arrivals, for a limited number of shared wavelength converters in an OBS node without FDLs. We do not consider the added complexity of converter sharing in the present work.

We note that the current paper is an extended version of [23], with additional results comparing the proposed model to simpler Markovian models and evaluation of the method for a wider range of FDL configurations.

B. Outline of Paper

The remainder of this paper is structured as follows. Section II gives an overview of Optical Burst/Packet Switching and common contention resolution techniques and then briefly explores the performance of different architectures of buffered optical switches. The characteristics of offered traffic are also briefly discussed. Section III defines the specifics of the buffered optical switch under study. Section IV introduces a model of traffic flows within the switch. Section V presents the analysis of the model, proposing methods for calculating blocking probability and mean delay. In Section VI our results are presented, with comparison to discrete-event simulations and to simpler analytic models for Poisson traffic. Our conclusions are given in Section VII.

II. OPTICAL PACKET/BURST SWITCHING

The deployment of cost-effective switching schemes that provide access to the large potential bandwidth offered by wavelength division multiplexed (WDM) optical transmission systems is the key to the successful accommodation of the growing volumes of heterogeneous Internet traffic in core networks. This argument serves as the main motivation for Optical Packet and Burst Switching schemes [24].

In Optical Packet Switching (OPS), data packets are processed and transmitted through the network in the optical domain without any conversion of the packet into electronic signals (Fig. 1). Each optical link carries multiple independent wavelength channels and high transmission capacities may be achieved, in the order of tens or hundreds of Gbps. The high speed of transmission creates challenges for packet header processing using current technologies as fast header processing in the optical domain is required [25].

In Optical Burst Switching (OBS), at edge nodes of the network, data packets destined for the same node are aggregated into larger optical packets, called bursts. This reduces the number of packet headers that require processing. Additionally, a control packet known as the Burst Header Packet (BHP) is associated with each burst and it is sent, on a dedicated out-of-band wavelength channel, to the core network prior to the burst’s transmission. The BHP sets-up the optical cross connects (OXC’s) in nodes along the burst’s path so that when the burst arrives at each node a transparent optical path through the switch and into an outgoing wavelength channel is provided. As the BHP is offset from the payload in time, it may be processed electronically. In this way, OBS circumvents
the disadvantages of OPS which requires faster optical switching and processing of packets in the optical domain.

The channel reservation mechanism in OBS can take different forms. The most often seen is the Just-Enough-Time (JET) protocol [26]. This is a one-way reservation scheme where, using knowledge of the offset time between the BHP and burst, the switches on a path are configured to route a burst just prior to its arrival (rather than immediately on receipt of the BHP). JET can achieve desirably low BHP/burst offset delays in wide area networks. As the channel reservation is not acknowledged, there is the possibility that a burst may find all output channels on a link to be busy. In this case the arriving burst is simply dropped (blocked) at the input port of the switch.

Both OPS and OBS share the desirable properties of optical transparency and the potential performance advantage of statistical multiplexing to achieve higher throughputs compared to optical circuit switching schemes with long-lived end-to-end connections. Both are also disadvantaged by the possibility of lost packets/bursts due to contention for outgoing channels at a switch (Fig. 2).

Packet/burst loss is the prevalent factor for performance of OPS/OBS networks and a number of contention resolution schemes have been proposed.

A. Contention Resolution Techniques

In OPS and OBS, contention occurs when two or more packets/bursts arrive from input ports on the same wavelength and are destined for the same output port (Fig. 2). Only one packet/burst may be carried on the corresponding outgoing wavelength. To alleviate this problem a contention resolution measure may be adopted in the switch. Two popular methods require additional hardware in the switch, Tuneable Wavelength Converters (TWCs) and Fiber Delay Lines (FDLs), which may be employed separately or in conjunction to significantly reduce packet/burst loss rates.

A Tuneable Wavelength Converter (TWC) is a device which can convert data from one incoming wavelength to one of the available outgoing wavelengths of the switch. This ability can avoid contention when two packets on the same incoming wavelength are destined for the same output port. An OPS/OBS switch can be equipped with full wavelength conversion or partial wavelength conversion. In the first case, each input (or output) channel is equipped with a dedicated wavelength converter; in the second case, a pool of converters are shared amongst channels, where the number of converters is less than the number of channels [1].

A Fiber Delay Line (FDL) is a fiber segment which introduces the possibility of providing a fixed delay time for the incoming optical packet/burst. This delay depends on the physical length of the FDL, which is normally limited by the large physical size and signal deterioration issues of a long fibre segment. In a switch equipped with an FDL, the contention between two packets at an output port channel may be resolved by directing one to the FDL to remove the overlap in time between the packets.

A switch may be equipped with multiple parallel FDLs, each with a different fixed delay time, offering a greater opportunity to remove overlaps. In this case, the set of FDLs normally have linearly increasing delays, where each delay is a multiple of a base delay time $C$, which is the delay of the shortest FDL. One factor in design of FDLs is the choice of $C$. If FDL time delays are significantly shorter than packet transmission times then the likelihood of successfully removing overlap between contending packets is low. If FDLs are very long, signal degradation issues may arise and average delay in the switch increases. An optimum value for $C$ may be decided by examining the effectiveness of the FDLs in reducing burst loss as $C$ varies (Fig. 3).

We note that an FDL may accommodate one or multiple independent wavelength channels. Therefore, the total

Figure 2. Contention in an Optical Packet/Burst Switch. Each link carries $N$ independent wavelength channels. Contention for a wavelength channel may arise when packets with the same input wavelength (at different input ports) are to be transmitted on the same output port. Additionally, ingress traffic at the node contends for outgoing wavelengths, adding to the congestion situation. Packet/burst dropping results from these contentions and it is left to higher network layers to retransmit the lost data, reducing network efficiency.

Figure 3. Example of blocking probability as a function of the FDL base delay, from discrete-event simulations. Packet arrivals are Poisson and packet lengths exponentially distributed of mean length 1ms. A base delay of two times the average burst length can be seen to be a safe compromise. Wider simulation studies of this topic have been conducted by Gauger [19], with similar conclusions.
number of buffer lines offered by a bank of FDLs is given by the product of the number of FDLs in the buffer and the number of wavelength channels accommodated in each FDL.

### B. Architecture of Optical Switches with FDLs

In proposed OBS switch designs it is common to find configurations of wavelength converters and FDLs in tandem, in order to more effectively resolve the output port contention problem. Optical switches can be variously configured with full, partial or no wavelength conversion and with or without a bank of single/multiple wavelength channel FDL buffers, arranged in feed-forward or feedback schemes (Fig. 4). Details of the design of OBS switch architectures are presented in [19].

Similar switch architectures have been proposed for OPS where FDLs are employed as a contention resolution mechanism, in [27] and [28] for example. Other OPS architectures, in [29] for example, employ FDLs to facilitate optical header processing (label switching). The model considered in this paper applies to the former use of FDLs only.

In a feed-forward architecture (Fig. 4(b)) each of \( P \) output ports has a dedicated FDL. The cross connect switch may redirect an incoming packet to one of \( N \) available channels in an FDL, in order to avoid a contention. With wavelength converters present at input ports, any available FDL channel may be selected. Having been delayed in the FDL, the packet is transmitted on the same wavelength at the output port. If it is not possible to select a wavelength that is available both in the FDL and the output port, at the required epochs, then the packet must be dropped. Although further efficiency could be gained by introducing additional wavelength converters between the FDL and the output port, this arrangement would incur significant addition hardware costs.

In a feed-back architecture (Fig. 4(c)) all output ports share a bank of \( K \) FDLs of different delay times \( k.C, k = 1 \ldots K \). With wavelength converters at input channels, a contending packet may be directed to any free channel in a chosen FDL. The number of channels in an FDL (denoted \( L \) in Fig. 4) may be less than the number of channels at the input port, allowing the FDL ports (and associated switch matrix) to be scaled according to cost/performance trade-offs. This flexibility is afforded by the additional \( L \) wavelength converters at each FDL return port, which allow re-conversion of packets to any of \( N \) output port channels. It is theoretically possible that a packet may recirculate multiple times through the switch and FDL bank, although signal degradation issues may limit the number of recirculations in practice and there are diminishing performance gains as FDL resource usage per packet increases with each recirculation.

Our focus in this paper is on the performance of feedback FDL switches but we make a brief performance comparison with feed-forward FDLs here, by way of simulation results from our OBS simulator. Fig. 5 presents a representative comparison of the loss rates (blocking probabilities) for switch architectures with no FDLs, with feed-forward FDLs (FF) and with different configurations of feed-back schemes (FB). Clearly the addition of FDLs, of any configuration, gives a significant improvement in blocking rates. The FB configurations generally outperform the FF configuration, even when the total number of FB FDL channels is less than the number of FF channels. This improvement is accounted for by the fact that two wavelength conversions may be performed in the FB case allowing independent wavelengths to be selected in the FDL and the output port. Broader simulation studies comparing different numbers of output port channels and channels per FDL have been conducted by Gauger [19], drawing similar conclusions.

We note that in the analysis that follows, we do not differentiate between blocking rates in multiple-channel and single-channel cases, only distinguishing performance based on the total number of FDL channels. For lower capacity FDL configurations however, estimates of blocking for either case are similar, as illustrated in Fig. 5 when comparing FB-1-4 with FF-4-1.

### C. Traffic Characteristics in OPS/OBS Networks

The brief simulation study of buffered optical switches in the previous subsection is based on the assumption of Poisson offered traffic. This may not be the most appropriate traffic model for either OPS or OBS networks.

![Figure 4. OBS switches with TWCs on all input channels and different FDL arrangements](image-url)
In the case of OPS, ingress traffic to the optical network would be expected to have the characteristics of data traffic in the edge-networks from which it emanates. It is generally accepted that many traffic processes in local and wide area networks do not follow a Poisson arrival process but instead exhibit a self-similar behavior.

In the case of OBS, Rostami and Wolisz [20] conclude from simulation studies that the nature of the burst assembly algorithm in OBS has a large impact on the traffic characteristics and resulting performance characteristics of the network. Burst assembly algorithms typically take one of two forms (i) timer-based where a burst is formed by collecting arriving packets over a fixed time period and (ii) size-based where packets are collected until some fixed burst size is reached. In either case, depending on the offered packet traffic feeding the aggregator, the interarrival distribution can vary from having a very low variance (consider case (i)) to that higher than Poisson traffic. Mountrouidou and Perros [18] have analysed such traffic characteristics of aggregated traffic for OBS networks, with similar conclusions.

In order to attempt to analyse the affect of wide variations in traffic characteristics expected in OPS/OBS, in the present paper a model for generally distributed inter-arrival times is proposed. The Gamma distribution is chosen due to the possibility of modelling arbitrary variance in a unified model. We note that Bhatnagar [30] has proposed a traffic model with Gamma-distributed inter-arrival times parameterised for high variances as an approximation for analysis of systems subjected to self-similar traffic.

In the case of OBS, Rostami and Wolisz [20] conclude from simulation studies that the nature of the burst assembly algorithm in OBS has a large impact on the traffic characteristics and resulting performance characteristics of the network. Burst assembly algorithms typically take one of two forms (i) timer-based where a burst is formed by collecting arriving packets over a fixed time period and (ii) size-based where packets are collected until some fixed burst size is reached. In either case, depending on the offered packet traffic feeding the aggregator, the interarrival distribution can vary from having a very low variance (consider case (i)) to that higher than Poisson traffic. Mountrouidou and Perros [18] have analysed such traffic characteristics of aggregated traffic for OBS networks, with similar conclusions.

In order to attempt to analyse the affect of wide variations in traffic characteristics expected in OPS/OBS, in the present paper a model for generally distributed inter-arrival times is proposed. The Gamma distribution is chosen due to the possibility of modelling arbitrary variance in a unified model. We note that Bhatnagar [30] has proposed a traffic model with Gamma-distributed inter-arrival times parameterised for high variances as an approximation for analysis of systems subjected to self-similar traffic.

### III. Optical Switch Under Study

In this section we formally describe the switch architecture and channel scheduling behaviour being modelled.

The system under study (Fig. 6) is an optical burst switching node with wavelength conversion and feed-back FDLs. It is assumed that the range of conversion from one wavelength to another is unrestricted and that there are as many converters at an input port as there are wavelength channels, that is, full wavelength conversion is available. We note that, although we deal with the case of burst switching, the model we develop may also be applied to an optical packet switch with feed-back buffers.

We focus on the analysis of blocking probability and mean delay at a single output port with \( N \) channels and a bank of FDLs containing \( K \) FDL units (\( K \) individual optical fibres). We will employ an aggregate model of the traffic offered to the output port, which consists of traffic from multiple input ports attempting to traverse the switch, ingress traffic that is being added to the network at this node and a portion of these traffic which are delayed in the bank of FDLs before reappearing at the output port. We note that a single output port model may quite easily be extended to a multi-port model by assuming independence of traffic streams from each input port, but we do not treat this case here.

Each FDL unit is a single fibre offering a constant delay of \( D_k \) seconds, \( k \in \{1, 2, \ldots, K\} \). Delay times of the units are each a multiple of a base delay time \( C \) so that \( D_k = kC \). We assume that the direct path from the switch to the output port has delay \( D_0 \approx 0 \). Additionally, each fibre may be wavelength division multiplexed carrying multiple wavelengths simultaneously with FDL unit \( k \) supporting \( L_k \) wavelength channels. The total number of wavelength channels provided by the bank of FDLs is denoted \( L = \sum_{k} L_k \).

A controller in the switch coordinates scheduling of the channels and FDLs. If none of the \( N \) output channels is available for the duration of a burst arriving at a time \( t \), an attempt is made to simultaneously schedule a free FDL (of delay length \( D_0 \)) and any output channel that will become free at time \( t + D_k \). The scheduler first attempts the procedure using FDL unit 1, offering delay \( D_1 \), and iterates in sequence through all \( K \) FDLs until a feasible schedule is found. If none of the available FDL delay times can resolve the schedule, then the
burst is blocked (lost). We assume that a burst may circulate through the FDL bank at most once, given signal degradation constraints. We next develop a traffic model which represents an approximate analogue of the switch resource scheduling behaviour just described.

IV. BUFFERED OUTPUT PORT TRAFFIC MODEL

We assume that the aggregate traffic arriving to the output port is of general renewal type (GI traffic) and burst lengths are taken to be exponentially distributed. Thus, the probability of blocking at the output port could be estimated, in the first instance, by analysing blocking in an GI/M/N/N system, where N is the number of output channels. A single GI/M/N/N model would, of course, not take into account the coordinated scheduling of output channels and FDLs in the actual system, which tends to correlate burst arrivals at the output channels in a manner that gives a reduction in blocking compared to that of a GI/M/N/N system. Our modelling aim is to approximate the improvement given by the FDLs without resorting to a detailed analysis of the traffic correlations involved. We model FDL behaviour as an additional GI/M/L/L blocking system and develop a model of virtual flows (Fig. 7) that approximates the overall output port scheduling behaviour.

We firstly make the observation that traffic which is potentially blocked by the output channels, before the scheduler attempts to resolve conflicts by delaying bursts in the FDLs, may be approximated as a (virtual) overflow traffic from an GI/M/N/N system representing the group of output channels. This overflow is indicated in Fig. 7 as flow \( \hat{F} \). We then consider this overflow traffic as forming offered traffic to an independent GI/M/L/L system representing the bank of FDLs.

We justify this lumped model of the FDL bank by observing that each FDL \( k \), consisting of a group of \( L_k \) channels, may be approximately modelled as an GI/M/L_k/L_k system. As traffic offered to the output channels is assumed renewal, then so is the overflow \( \hat{F} \) [21] and as the scheduler first attempts to resolve a conflict with FDL 1, we may consider FDL 1 as an independent loss system offered all overflow traffic from the group of \( N \) channels. FDL 1 is itself a group of \( L_1 \) channels and, when all \( L_1 \) channels are occupied, the scheduler cannot resolve a conflict using FDL 1 and instead attempts to resolve it with the delay offered by FDL 2. Thus we can view FDL 1 as generating its own renewal overflow traffic \( \hat{F}_1 \) which in turn is offered to FDL 2, and so on down the chain of \( K \) FDLs, with each FDL \( k \) producing overflow which is offered to FDL \( k+1 \). These virtual traffic flows within the FDL bank are depicted in Fig. 9. Overflow \( \hat{F}_K = F_B \) from the final FDL represents the actual overflow from the output port. This traffic flow, \( F_B \), is lost from the system (blocked). For the purposes of calculating overflow (blocking) from the FDL bank, we may combine this cascade of overflowing loss systems as a single GI/M/L/L system, where \( L \) is the aggregate number of channels in the bank. To calculate mean delay, we resolve the occupancy in each of the \( K \) FDLs.

To complete the flow model, we consider the combined traffic carried by all FDLs in the bank as a traffic flow that is offered again (notionally) to the output channels, at some time in the future. This total carried traffic flow from the FDLs, \( \hat{F} \), competes with the input traffic flow (\( F_I \)) for

---

**Figure 7. Virtual Flow Model of Output Port with FDLs**

**Figure 8. We associate three traffic flows with the scheduling behaviour of the switch, as illustrated here for the simple case of a single output channel and a single FDL channel. The offered traffic flow emanates from input ports and new ingress traffic. If an incoming burst is blocked we consider it as part of an overflow traffic \( \hat{F} \) which is offered to the FDL channel. If carried by the FDL it is again offered to the output channel as part of a flow \( \hat{F} \).**

**Figure 9. Virtual Cascading Overflows Within FDL Bank**

---

© 2011 ACADEMY PUBLISHER
the output channels at that future time. We neglect time correlations between these flows and identify an effective (virtual) flow $F$ that is the aggregation of the input flow $(F_I)$ and the FDL carried traffic $(\hat{F})$ that is fed back to the input of the channel model.

As the feedback traffic is not renewal, neither is the aggregated input traffic $(F)$. For the purpose of formulating an approximate model, we assume that the feedback flow $(\hat{F})$ is small in comparison with $(F_I)$ and so the renewal nature of $F$ is assumed to be undisturbed.

We characterise the various traffic flows in the model using the notion of an infinite server (or ‘infinite trunk group’) [31], whereby a traffic flow is described in terms of the moments of the channel occupancy distribution in a $GI/M/\infty$ system when offered an identical traffic flow. The channel occupancy distribution may be classed as being ‘peaked’, when the variance $V$ is greater than the mean $M$, or ‘smooth’ when the variance is less than the mean. The ‘peakness’ of the traffic is denoted as $Z = V/M$. The mean of the occupancy distribution is termed ‘traffic intensity’. We summarise the main flows in the model and identify the traffic moments of interest, below. We identify either the central or factorial moments of the flows depending on which representation is the most convenient in the analysis that follows (Section V).

- $F_I$ is the actual traffic flow offered to the output port. It is assumed to be renewal, that is, burst interarrival times are independent and identically distributed. The factorial moments of this traffic flow are denoted $M_{I,0}, j \in \mathbb{N}$.
- $F_O$ is the actual carried traffic from the node, with traffic intensity $M_O$.
- $F_B$ is the total actual blocked traffic from the node, with factorial moments denoted $M_{B,j}, j \in \mathbb{N}$.
- $\hat{F}$ is the virtual overflow traffic from the $GI/M/N/N$ system. This flow constitutes the traffic that must either be delayed and scheduled on output channels for transmission at a later time, or else blocked if there is no feasible schedule. The factorial moments of the flow are denoted $M_{(j)}, j \in \mathbb{N}$.
- $\hat{F}$ is the carried traffic from the $GI/M/L/L$ system. This flow represents the traffic that is successfully scheduled to be delayed in the FDL bank and subsequently carried by the output channels. The first and second factorial moments of the flow are denoted $M_{(1)}$ and $M_{(2)}$ respectively.
- $F$ is the effective total offered traffic at the output channels. This consists of the actual offered traffic to the port plus the traffic flow generated by previously delayed traffic from the FDL bank. It is assumed to be renewal with factorial moments $M_{(j)}, j \in \mathbb{N}$.
- With respect to flows within the FDL bank (Fig. 9), $\hat{F}_k$ is overflow traffic from FDL $k$, with mean and variance $M_k$ and $V_k$. The mean of the channel occupancy in FDL $k$ is denoted $\bar{M}_k$.

We next analyse the model of Fig. 7 to resolve the moments of the flows identified above. Having done so, we may estimate the burst blocking probability at the output port and then, by resolving the flows of Fig. 9, we may estimate the mean delay experienced by a packet transiting through the port.

V. MODEL ANALYSIS

To resolve blocking probability we require the mean of flow $F_B$. We first resolve the effective input traffic flow, $F$, from which calculation of the other flows follow. Although only the mean of flow $F_B$ is required, we include higher moments of the flows in calculations in order to achieve an accurate estimate.

A. Model of Offered Traffic

We model the offered traffic flow $F_I$ as having interarrival times distributed according to a gamma distribution. This characterisation enables performance for a full range of offered-traffic peakness to be examined. We note, however, that the methods that follow allow any independent interarrival time distribution to be represented.

In order to apply the gamma distribution in our analysis, we need to first derive the relationship between the parameters of the distribution and the moments of the traffic, that is, the moments of the occupancy distribution in an infinite trunk group with exponential holding times, when offered traffic with gamma-distributed interarrivals.

In the $GI/M/\infty$ queue, it is well known [22] that the factorial moments of the offered traffic, denoted $M_{(j)}$ here, may be expressed in terms of the interarrival time distribution for a renewal arrival process as

$$M_{(j)} = \frac{1}{\mu E[\tau]} \prod_{i=1}^{j-1} \frac{i F^\ast(i\mu)}{1 - F^\ast(i\mu)}, \quad j \in \mathbb{N}, \quad (1)$$

where $F^\ast(\cdot)$ denotes the Laplace-Stieltjes transform (LST) of the interarrival cdf, $\mu$ is the parameter of the exponentially distributed holding times in the infinite trunk group and $E[\tau]$ is the mean interarrival time. In our analysis, we will also require expressions for the first two moments of the traffic in terms of the interarrival time distribution, and we derive these as follows. Let $\tau$ be the random variable denoting the interarrival time where $\tau$ has a gamma distribution, that is, its probability density function $f_\tau(t)$ is given by

$$f_\tau(t) = \frac{\theta^{-k} t^{k-1} e^{-\theta t / \mu}}{\Gamma(k)} \quad t \geq 0 \quad (2)$$

where $k > 0$ is the shape parameter, $\theta > 0$ is the scale parameter and $\Gamma(k)$ is the gamma function. The LST $F^\ast(s)$ of the corresponding cumulative distribution function $F_\tau(t)$ is given as

$$F^\ast(s) = \int_0^\infty e^{-st} f_\tau(t) \, dt = (1 + \theta s)^{-k} \quad (3)$$

from which the first moment of the interarrival time $\tau$ is

$$E[\tau] = -\frac{dF^\ast(s)}{ds} \bigg|_{s=0} = \theta k. \quad (4)$$

We now wish to find values of the parameters $\theta$ and $k$ such that traffic with interarrival time $\tau$ arriving to an
infinite trunk group has a given mean intensity $M$ and peakedness $Z$. From (1) and (3) we may calculate the first two factorial moments of the traffic as

$$M(1) = \frac{1}{\mu E[\tau]} = M \quad (5)$$

$$M(2) = \frac{1}{\mu E[\tau]} \frac{(1 + \theta \mu)^{-k}}{1 - (1 + \theta \mu)^{-k}} = \frac{M}{(1 + \frac{1}{\Lambda_{\theta}})^{k} - 1} \quad (6)$$

The mean and peakedness expressed in terms of the factorial moments of the offered traffic are

$$M = M(1) \quad \text{and} \quad Z = 1 - M(1) + M(2)/M(1), \quad (7)$$

and so we may relate the mean and peakedness of the traffic to the gamma distribution parameters by the equations:

$$\theta = \frac{1}{M \mu k} \quad (8)$$

$$Z = 1 - M + \frac{1}{\left(1 + \frac{1}{\Lambda_{\theta}}\right)^{k} - 1} \quad (9)$$

Given desired values of mean $M$ and peakedness $Z$ of the offered traffic, we may solve (9) numerically to yield corresponding values of $k$ and $\theta$.

It is also useful to derive the bounds on traffic peakedness $Z$ for gamma interarrivals. From (9) we see that, as $k \rightarrow 0$, $Z \rightarrow \infty$, so there is no upper bound. To find the lower bound on $Z$, we compute the limit of $(1 + \frac{1}{\Lambda_{\theta}})^{k}$ as $k \rightarrow \infty$. This limit has the indeterminate form $1^\infty$ but we may transform to the form $0^0$ and apply l'Hôpital’s rule to find

$$\lim_{k \rightarrow \infty} \left(1 + \frac{1}{M \mu k}\right)^{k} = e^{1/M} \quad (10)$$

and so the lower bound on $Z$ is given as

$$Z_{\text{min}} = 1 - M + (e^{1/M} - 1)^{-1} \quad (11)$$

This limit is identical to the general result [22], so we may conclude that there is no restriction on the range of peakedness we may examine using gamma-distributed interarrivals.

B. Overflow and Carried Traffics

We wish to characterise the overflow traffic from the $G1/M/N/N$ system, representing the output port channels (Fig. 7). Let us assume initially that there is no feedback flow $\tilde{F}$ and so the effective offered flow $\tilde{F}$ is equal to the actual gamma-distributed offered flow $F_1$. We may calculate the factorial moments of the overflow $\tilde{F}$, from Potter’s formula [22], as

$$\frac{1}{M(k)} = \sum_{l=0}^{N} \frac{N!}{l!} \frac{(k + l - 1)!}{(k - 1)! M(k + l)} \quad , \quad k \in \mathbb{N} \quad (12)$$

where $M(j), j \in \mathbb{N}$ are the factorial moments of the offered traffic $F$, which may be computed from (1) given the LST of the gamma distribution from (3).

In a similar manner, we may compute the factorial moments of the overflow $F_B$ from the FDL bank, given the factorial moments of the offered traffic, which in this case is the flow $\tilde{F}$ with factorial moments $\tilde{M}(j), j \in \mathbb{N}$ computed by (12):

$$\frac{1}{M_B(k)} = \sum_{l=0}^{N} \frac{N!}{l!} \frac{(k + l - 1)!}{(k - 1)! M(l + k)} \quad , \quad k \in \mathbb{N} \quad (13)$$

We have calculated the overflow moments when the feedback traffic $\tilde{F}$ is neglected. To accurately estimate $\tilde{F}$ (and subsequently all other flows in the model) we account for the additional feedback traffic as follows. Given an estimate of the moments of $F$, we may calculate the first two moments of the carried traffic $\tilde{F}$ using Brandt’s calculation [22], where the offered traffic in this case is again the overflow $\tilde{F}$ with factorial moments $\tilde{M}(j)$ given by (12), that is,

$$\tilde{M}(2) = \frac{\tilde{M}(2)}{\tilde{M}(1)} - \tilde{M}(1) \tilde{M}(2) \sum_{l=1}^{L} \frac{L!}{M(l+1) \sum_{m=1}^{L} \left(\frac{m \tilde{M}(m)}{M(m+1)} + 1\right)}, \quad (14)$$

where $\tilde{M}(1) = \tilde{M}(1) - \tilde{M}(2)$, by the conservation principle, and $\tilde{M}(1)$ and $\tilde{M}(2)$ given by (13). We note that (14) gives the required moments of the “freed” carried traffic as distinct from the moments of channel occupancy, provided by the usual equivalent random methods [21].

Having calculated the moments of the feedback traffic, we now make the assumption that $\tilde{F}$ may be estimated as begin gamma-distributed traffic with moments determined as follows. The mean of $\tilde{F}$ may be calculated simply as the sum of the means of $\tilde{F}$ and the actual offered traffic $F_1$, that is,

$$\tilde{M}(1) = M(1) + M(I_{(1)}) \quad (15)$$

We make the assumption that $\tilde{F}$ and $\tilde{F}$ are independent traffic streams and so the variance of $\tilde{F}$ may similarly be estimated as the sum of the variances of $\tilde{F}$ and $F_1$ or, in terms of the factorial moments, we may derive

$$\tilde{M}(2) = 2M(I_{(1)}M(1)) + M(I_{(2)}), \quad (16)$$

Given the first two moments of $\tilde{F}$, which we have assumed remains gamma-distributed, we may calculate further moments by calculating the distribution parameters $k$, $\theta$ from (8) and (9), calculating the distribution’s LST from (3) and then calculating higher factorial moments from (1).

We now have a set of open-form equations relating the factorial moments of all flows from which we may form an iterative algorithm to resolve the blocking probability.

C. Resolving Blocking Probability

To resolve the factorial moments of the effective offered traffic $F$ we first calculate overflow and carried traffic moments assuming no feedback flow $\tilde{F}$. This yields an approximation for the moments of $\tilde{F}$ from which a new estimate for $\tilde{F}$ may be calculated. We then iterate this calculation until the first two moments of $\tilde{F}$ are within
a desired $\epsilon$ over two successive iterations. We note from [22] that the complexity of calculation of overflow and carried traffic moments in (12), (13) and (14) is $\mathcal{O}(C)$, where $C$ is the number of channels, and thus our simple iterative method has good efficiency. (We have found the algorithm to converge rapidly for a range of test cases, although we do not have a convergence proof.) Given the solution values for the moments of $\mathcal{F}$, we have a solution value for the first moment of the node overflow traffic $\hat{M}_{B,(1)}$ from (13) and so the burst blocking probability at the node may be calculated as

$$B = \frac{\hat{M}_{B,(1)}}{M_{L,(1)}} \quad (17)$$

D. Resolving Mean Delay

Delay in the system occurs when FDLs are employed by the scheduler to resolve contention at the output channels. To estimate the mean delay we first resolve the mean and variance of the offered traffic to each of the $K$ FDL units (Fig. 9). Having done so, we may then resolve the mean occupancy of each FDL, $\hat{M}_k$, from which, given a set of FDL delay times $\{D_k\}$, we may approximate the mean delay in the system.

We denote the mean and variance of the overflow from FDL $k$ as $\hat{M}_k$ and $\hat{V}_k$ respectively, as per Fig. 9. As the overflow from an FDL is the offered traffic to the next FDL in the chain, the offered traffic to FDL $k$ has mean and variance $\hat{M}_{k-1}$ and $\hat{V}_{k-1}$.

Having solved for the factorial moments $\hat{M}(k)$ of the overflow from the group of $N$ channels in the previous subsection, the mean and variance of the traffic offered to the first FDL in the FDL bank are given as

$$\hat{M} = \hat{M}_{(1)}$$
$$\hat{V} = \hat{M}_{(1)} - \hat{M}_{(1)}^2 + \hat{M}_{(2)} \quad (18)$$

We now wish to resolve the mean and variance of the overflow from FDL 1, $\hat{M}_1$ and $\hat{V}_1$ respectively, when it is offered traffic $\hat{M}, \hat{V}$. We employ Equivalent Random Theory (ERT) to resolve $\hat{M}_1, \hat{V}_1$ [21]. Having done so, $\hat{M}_1, \hat{V}_1$ becomes offered traffic to FDL 2 and, assuming independence between flows, reapplying ERT resolves $\hat{M}_2, \hat{V}_2$ and so on down the chain of $K$ FDLs.

We show the solution for an arbitrary FDL $k$ receiving traffic $\hat{M}_{k-1}, \hat{V}_{k-1}$ and producing overflow $\hat{M}_k, \hat{V}_k$. With this solution and $\hat{M}_0, \hat{V}_0$ given by $\hat{M}, \hat{V}$ respectively, we may iterate for all $K$ FDLs in the bank. The details of the method follow.

In Equivalent Random Theory, a virtual group of size $N^*$ is offered virtual Poisson traffic of intensity $A^*$ which produces an overflow mean and variance which may be matched, given appropriate values of $N^*$ and $A^*$, to the given (actual) mean and variance. This overflow traffic is the offered traffic to the actual group. The problem reduces to finding the $A^*$ and $N^*$ group whose overflow matches the required actual (peaked) offered traffic. Having resolved $A^*$ and $N^*$, the mean and variance of the overflow (and carried traffic) from the actual group may be resolved using the equivalent overflow model of Fig. 10.

From Fig. 10, for FDL $k$, expressions for the mean and variance of the actual overflow, in terms of the virtual group size $N^*_k$, the virtual offered intensity $A^*_k$ and the actual group size $L_k$ are given by the equivalent system as

$$\hat{M}_k = A^*_k \cdot E(A^*_k, L_k + N^*_k) \quad (20)$$
$$\hat{V}_k = \hat{M}_k \left(1 - \frac{A^*_k}{L_k + N^*_k + 1 - A^*_k + \hat{M}_k}\right) \quad (21)$$

$A^*_k$ and $N^*_k$ are given implicitly in terms of $\hat{M}_{k-1}$ and $\hat{V}_{k-1}$, the previously calculated mean and variance of the overflow from the virtual source, as

$$\hat{M}_{k-1} = A^*_k \cdot E(A^*_k, N^*_k) \quad (22)$$
$$\hat{V}_{k-1} = \hat{M}_{k-1} \left(1 - \frac{A^*_k}{N^*_k + 1 + \hat{M}_{k-1} - A^*_k}\right) \quad (23)$$

From (22) and (23), $N^*_k$ may be written in terms of $A^*_k$ and known constants $\hat{M}_{k-1}$ and $\hat{V}_{k-1}$ as

$$N^*_k = A^*_k \left(\frac{\hat{M}_{k-1} + \hat{V}_{k-1}/\hat{M}_{k-1}}{\hat{M}_{k-1} + \hat{V}_{k-1}/\hat{M}_{k-1} - 1} - \hat{M}_{k-1} - 1\right) \quad (24)$$

and so, from (22), we have a function of a single variable $A^*_k$, $f(A^*_k) = \hat{M}_{k-1} - A^*_k \cdot E(A^*_k, N^*_k) = 0$, (25) which may be solved for $A^*_k$ as a numerical root finding problem. We may choose an initial solution for the numerical solution from Rapp’s approximation [21] for an overflow system:

$$A^* \approx V + 3Z(Z - 1)$$

$$N^* \approx \frac{A^*(M + Z)}{M + Z - 1} - M - 1$$

where $Z$ is the peakedness.
We note that, in the numerical method, the values of $N^∗$ must be allowed to take non-integer values for a solution to be found. The usual recurrent evaluation method for the Erlang B formula

$$E(A, k + 1) = \frac{A \cdot E(A, k)}{k + 1 + A \cdot E(A, k)} \quad E(A, 0) = 1$$

is extended using Szypick’s approximation [32] which gives the blocking probability for real-valued $0 \leq N \leq 2$

$$E_s(A, n) \approx \frac{(2 - n)A + A^2}{n + 2A + A^2} \quad n \in \text{real interval } [0, 2].$$

For a given positive real-valued $N = \lfloor N \rfloor + (N - \lfloor N \rfloor)$, where $N$ may be $\geq 2$, we first evaluate

$$E(A, N - \lfloor N \rfloor) = E_s(A, N - \lfloor N \rfloor).$$

and then (from (26)) form the recursion

$$E(A, k + 1 + (N - \lfloor N \rfloor)) = \frac{A \cdot E(A, k + (N - \lfloor N \rfloor))}{k + 1 + (N - \lfloor N \rfloor) + A \cdot E(A, k + (N - \lfloor N \rfloor))}$$

where, for $k = 0$

$$E(A, 0 + N - \lfloor N \rfloor) = E_s(A, N - \lfloor N \rfloor).$$

Iterating for $k = 0, 1, \ldots, \lfloor N \rfloor - 1$ gives the final value of $E(A, N)$, for positive real-valued $N$.

We have solved (25) for $A^*_k$, and thus $N^*_k$ is given by equation (24). The mean and variance of the overflow traffic from FDL $k$ are then given by equations (20) and (21) respectively. We now have the mean of the carried traffic from FDL $k$ as

$$\bar{M}_k = \bar{M}_{k-1} - \bar{M}_k.$$

With this solution for FDL $k$, and $\bar{M}_0, \bar{V}_0$ given by $\bar{M}, \bar{V}$, we may solve for all $k \in \{1, 2, \ldots, K\}$ iteratively. The average burst delay $D$ at the output port is then given as

$$D = \sum_{k \in \{1, \ldots, K\}} \frac{\bar{M}_k}{M_O} D_k$$

where $M_O$ is the mean of the carried traffic from the port, which may be calculated, by the conservation principle, as:

$$M_O = M_{(1)} - \bar{M}_{(1)}.$$

VI. RESULTS AND ANALYSIS

We compare analytic results for blocking $B$ and mean delay $D$ with results from a discrete-event simulation of an OBS node implemented in Opnet Modeler™ [33]. Two different node configurations are considered, Scenario I: a node with 10 output channels and 2 FDLs and Scenario II: a node with 40 output channels and 5 FDLs. In both cases, each FDL carries a single wavelength ($L = K$).

Our discrete-event simulator models the full details of the output channel and FDL scheduling. The channel scheduler implements Latest Available Unscheduled Channel (LAUC) on both the output channels and the FDLs. When there is no output channel available for an arriving burst, coordination of output channel and FDL scheduling is of the “PreRes” type [16]. In this scheme a schedule is sought simultaneously for future availability of an output channel and FDL. Also, the simulator implements full wavelength conversion at the output port.

Burst interarrival times are gamma-distributed and burst lengths are exponentially distributed of mean length $1\text{ms}$. We note that our simulator packet generator is parameterised by the gamma-distribution parameters $(k, \theta)$ while our analytic model is parameterised by the factorial moments of gamma-distributed traffic offered to a virtual $GI/M/\infty$ group, however, we may match simulation setup with analytic model input values by evaluating our previously derived relations (8), (9) to give the appropriate $(k, \theta)$ to generate a desired mean and peakedness of offered traffic.

The FDL base delay time is chosen as $C = 2\text{ms}$. It has been illustrated in Section II.A that, when $C$ is shorter than the average burst length, the FDLs are less effective and blocking increases due to increased overlapping of bursts at the output channels. When $C$ is increased beyond the average burst length, burst blocking settles to a near constant value for a given load. $C$ should not be too large, as fibre lengths in FDLs become unfeasibly long and delay increases. We set $C$ to be twice the average burst length as a trade-off.

The simulations were executed such that the confidence interval for all points is better than $\pm 2\%$ at a confidence level of $98\%$. These intervals are small and for clarity are not marked on results plots below.

We compare simulation and analysis for blocking probability for Scenarios I and II in Fig. 11 and Fig. 12 respectively. Results are shown over a range of mean traffic intensities and for different traffic peakedness values, from smooth ($Z = 0.75$) to peaked ($Z = 2.5$), with $Z$ corresponding to Poisson traffic. Mean load values are shown normalised with respect to the number of output channels $N$. Results from our analytic model in both scenarios compare favourably with simulation over the range of traffic intensity and peakedness examined.
We similarly compare results for mean delay $D$ for Scenarios I and II in Fig. 13 and Fig. 14 respectively. With analytic results, Fig. 15 more clearly illustrates the effect of increasing peakedness on blocking probability, for a fixed mean traffic intensity $\rho$. For higher mean loads, it can be seen that blocking probability increases quite strongly with peakedness and thus the peakedness of the offered traffic (or more generally the interarrival-time distribution) is an important factor in determining system performance.

In addition to providing results for non-Poisson input traffic, we would hope that our model, for the case of Poisson traffic, is more accurate than simpler Markovian models. A number of previous studies, [11] and [34] for example, have proposed using an $M/M/N/N + K$ model to estimate blocking at an OBS output port with $N$ channels when equipped with $K$ FDL channels. A Birth-Death analysis of the $M/M/N/N + K$ system yields the probability of blocking. Given the memoryless property of the arrivals, the blocking probability is given by the probability of $N + K$ customers in the system, which is equal to

$$
\frac{(\lambda/\mu)^{N+K}}{N!N^K} \sum_{n=0}^{N} \frac{\lambda^n}{n!} + \frac{(\lambda/\mu)^N}{N!} \sum_{n=N+1}^{N+K} \rho^{n-N}
$$

where $\lambda$ is the mean arrival rate, $\mu$ the mean service rate and $\rho = \lambda/(\mu N)$.

We may also consider simpler Markovian models as an approximation. Simply ignoring the presence of the FDLs, an $M/M/N/N$ model (Erlang-B) could be considered. Alternatively, considering the FDL channels as additional output channels, an $M/M/N+N+K$ model could be considered. A comparison of all three Markovian models against simulation and our analytic model is given in Fig. 16, for the case of $N = 10$ channels and $K = 5$ FDL channels. Expectedly, overall the $M/M/N+N+K$ model fares best of the three Markovian models, although at low loads the $M/M/N+N+K$ model approaches a better approximation. We can conclude that at low loads one additional FDL channel will improve performance by approximately the same extent as one additional output channel. At very high loads the $M/M/N/N$ model approaches a reasonable approximation; as load increases, FDLs become less and less effective at resolving a contention. We see that our analytic model is a considerable improvement on the simpler Markovian models.

We have assumed, for the above results, that the overall capacity of the FDL bank is relatively small compared to the capacity of the output port. We reason that this would be a suitable configuration to limit the overall cost of the switch, however, we also compare results for the case of larger capacity FDL banks in Fig. 17. In this case, there...
are \( N = 10 \) output port channels and varying numbers of single-channel FDLs, up to \( K = N \). We note that for \( K > N \) the additional FDL channels beyond \( K = 10 \) channels cannot be used and there is no performance gain possible. Offered traffic is Poisson, allowing us to also compare with the \( M/M/N/N \) model.

We observe that, for some load values, our analytic method decreases in accuracy as the number of FDL channels increases, yet it provides a significantly better approximation than the \( M/M/N/N + K \) model, overall.

VII. CONCLUSION

We have developed a relatively simple approximate model for the analysis of an OBS node with FDLs by applying circuit switching analysis methods in a novel way, by allowing a feedback path between groups of channels (a situation not normally occurring in circuit-switched networks). Our overall aim is to produce a model of good accuracy and good numerical efficiency that may be extended to modelling and dimensioning of large networks of optical switches. We note the potential usefulness of modelling smooth, as well as peaked, offered traffic. As carried traffic from a group of channels is generally smoother than offered traffic, in network models smooth offered traffic may be encountered at some point on a transmission path, even if traffic is peaked at the ingress point. The traffic peakness will also vary with the burst aggregation mechanism in use in an OBS network. The model may be used to evaluate performance under any \( GI \) traffic stream that may be expressed in terms of the LST of the interarrival distribution. As future work, an extension to the model will be to include limited numbers of shared wavelength converters in the node model and to investigate dimensioning and optimisation based on this extended model.

REFERENCES


Conor McArdle received a B.Eng, in Electronic Engineering from Dublin City University in 1997. In the same year he joined Teltec Ireland in DCU as a researcher working on European-funded ACTS-framework projects and international standardisation in the Telecommunications Domain Taskforce of the Object Management Group (OMG), with a focus on modelling and performance optimisation of distributed systems for telecommunication services provisioning. He completed his PhD thesis on this topic in 2004 and then joined the School of Electronic Engineering in DCU as a contract Lecturer. In August 2007 he joined the Research Institute for Networks and Communications Engineering (RINCE) at Dublin City University as a Research Fellow and is currently working on optical network performance evaluation and dimensioning.

Daniele Tafani was born in Tivoli, Italy on December 12th 1981. He received the Laurea degree in Telecommunications Engineering from University “La Sapienza” of Rome in 2006. From 2006 to 2008 he was employed in IBM Tivoli Laboratories of Rome as a software developer. He joined the Research Institute for Networks and Communications Engineering (RINCE) at Dublin City University, in October 2008 where he is currently working toward the PhD degree in Electronic Engineering. His main interests are focused on modeling of optical burst switching networks and optimisation problems.

Liam P. Barry received his BE (Electronic Engineering) and MEngSc (Optical Communications) degrees from University College Dublin in 1991 and 1993 respectively. He was then employed as a Research Engineer in the Optical Systems Department of France Telecom’s Research Laboratories in Lannion, France, where he worked on the use of ultra short optical pulses in high capacity networks. As a result of this work he obtained his PhD Degree from the University of Rennes in France. In February 1996 he joined the Applied Optics Centre in Auckland University, New Zealand, as a Research Fellow, and in March 1998 he took up a lecturing position in the School of Electronic Engineering at Dublin City University, and established the Radio and Optical Communications Laboratory. His main research interests are; all-optical signal processing, optical pulse generation and characterization, hybrid radio/fibre communication systems, wavelength tunable lasers for packet/burst switched optical networks, and optical performance monitoring.
Comparison of the Design Characteristics of MMI Wavelength Demultiplexers Using Different Approaches by Computing the Effective Index

M. Najjar¹,², R. Rejeb³, H. Rezig¹, M. S. Obaidat⁴, Fellow of IEEE

¹ High school of engineering (ENIT), SysCom Laboratory, PB 37, Tunis 1002, Tunisia
² High institute of computer sciences (ISI), Elmanar, Ariana 2080, Tunisia
³ Institute for advanced engineering and research, Germany
⁴ Department of Computer Science & Software Engineering, Monmouth University, West Long Branch, USA

Corresponding Author: Prof. M. S. Obaidat at: obaidat@monmouth.edu

Abstract – Wavelength Division Multiplexing (WDM) is the key technique for high data rates communication systems. WDM systems offer many advantages for optical communications with several unique features and requirements in terms of their implementation. This is because the design characteristics and realization of WDM components are crucial and depend particularly on the type of materials used for fabrication.

In this article, we investigate the design characteristics of a wavelength demultiplexer using the effective index method in order to determine the propagation constants and field distributions of fundamental modes in dielectric waveguides. In order to analyze the impact of computation method on the design characteristics of the demultiplexer, the effective index is computed by an analytic approach. To this end, we propose a new algorithm for developing an analytic model. We can confirm that design characteristics of WDM components depend on the approach chosen for computing the effective indexes.

Index Terms– Wavelength division multiplexers, Multi-Mode Interference length, MMI width, effective index method, beam propagation scheme, analytic method, extinction loss, insertion loss.

I. INTRODUCTION

Rib waveguides have become of great importance in integrated optics due to their capability of laterally confining the optical field. A rib waveguide has the guiding layer made of a slab with a strip or strips superimposed onto it. It offers captivity of the wave in 2-dimensions. Although this characteristic makes rib waveguides extremely convenient as basic building blocks for integrated optical devices, it requires the quantitative description of the modal field in the waveguide. Thus, the accurate evaluation of propagation characteristics involves the resolution of the electromagnetic boundary value problem for the structures of dielectric guides. Since a simple and accurate analytical method does not exist to date, various approximate approaches based on numerical techniques have been proposed to solve this issue [1]. However, most of these numerical techniques including circular harmonic analysis, direct numerical integration of the field equations [2], finite element analysis [3], finite difference analysis [4], field expansion in orthogonal functions [5,6], beam propagating method-3D [7] and equivalent network technique [8] require a huge computational time and do not lead to simple analytical solutions for the modal fields.

The Effective Index method (EIM) [9-10] is a widely used technique to determine the propagation constants and field distributions of fundamental modes in dielectric waveguide. Specifically, it is very useful in combination with the 2D Beam Propagation Method (BPM) to transform the 3D to 2D structure and reduce the computation complexity.

In this article, we investigate the design characteristics of a wavelength demultiplexer which can be used for optical access networking to separate two wavelengths \(\lambda_1 = 1530\) nm and \(\lambda_2 = 1565\) nm. In the context of this work, we consider a 1x2 wavelength demultiplexer model based on the Multi-Mode Interference technique (MMI) and the rib waveguide. In order to analyze the impact of computation method on the design characteristics of demultiplexer, the effective index is computed by a numerical method (NM) and an analytic method (AM).

Section 2 presents the simulation model of the wavelength demultiplexer considered. Section 3 proposes a novel numerical approach for computing the effective index. The simulation results and design characteristics of the MMI wavelength demultiplexer are discussed in Section 4. Finally, we conclude this article and present open direction for future work.

II. SIMULATION MODEL

WDM demultiplexers are generally realized as a multilayer structures based on the different optical waveguides such as rib or ridge guides. Regardless of the waveguide used; they will be simulated in 2D structure as this simplifies the task. Therefore, the transformation
from 3D to 2D form can be ensured by using the effective index method as it is shown in Figure 1. Let $n_c$ be the core index, $n_g$ the cladding index, $n_s$ the substrate index, $(n_{	ext{eff}})_I$ the effective index of the core layer, $(n_{	ext{eff}})_I$ the effective index of the first cladding layer and $(n_{	ext{eff}})_I$ the effective index of the second cladding layer.

![Image 1](image1.png)

**Figure 1**: Two views of a rib waveguide: (a) 3D rib waveguide, and (b) 2D view of the rib waveguide

The rib waveguide based on SiO$_2$/polymer/BK7 is used to realize the wavelength demultiplexer, which is designed to separate two wavelengths of two different bands $\lambda_1=1530$ nm (band C) and $\lambda_2=1565$ nm (band L). These bands are used by the optical access networks where the C band is reserved for the upstream transmission and L band for the downstream transmission [11]. The different layers of the model are presented in Figure 1a. The 2D equivalent model is shown in Figure 1b; it is given by the effective index of the three regions: guide region and two cladding regions. In the case of symmetrical structure the effective indexes of the cladding regions are the same.

The principle of effective index method is the separation of variables technique of the propagation equations. Therefore, many schemes are developed to calculate the effective index of a rib waveguide. In this work, we study two methods to determine the effective index and we will compare the results of simulation analysis obtained by each of them.

The first method, the analytic scheme, is based on the analytic analysis of the following nonlinear equations (1, 2):

$$b_{1,II} = -\pi - \arctan\left(\frac{1-b_{1,II}}{b_{1,II}}\right) - \arctan\left(\frac{1-a}{b_{1,II} - a}\right) = 0 \quad (1)$$

$$b_{1,II} = \left(n_{	ext{eff}}_{I,II}^2 - n_s^2\right)\left(n_g^2 - n_s^2\right) \quad (2)$$

Where $b_{1,II}$ are the constants to be calculated corresponding region I or region II and $a$ is an asymmetry parameter which given by:

$$a = \left(n_g^2 - n_s^2\right)\left(n_s^2 - n_s^2\right) \quad (3)$$

Generally, the effective index of a multilayer structure depends on the height of each layer. Therefore, the variation effect of the rib height $d$ and the film thickness $h$ versus the guide and cladding effective indexes are studied for the two wavelengths ($\lambda_1$, $\lambda_2$) and presented by Figures 4 and 5.

### III. NUMERICAL METHOD

In the numerical method (NM), the effective index $(n_{	ext{eff}})_I, II$ of the lowest guided mode for respective slab structure is then determined by solving the eigenvalue equations of asymmetric slab waveguide which are given by [10]:

$$\tan(k_{2x}d - M\pi) = \frac{k_{2x}(\gamma_1 + \gamma_3)}{k_{2x}^2 - \gamma_1\gamma_3} \quad (4a)$$

$$k_{2x}d - \tan^{-1}\left[\frac{n_2^2}{n_1^2}\frac{\gamma_1}{k_{2x}}\right] - \tan^{-1}\left[\frac{n_2^2}{n_3^2}\frac{\gamma_3}{k_{2x}}\right] - N\pi = 0 \quad (4b)$$

where,

- $k_0$: Free space propagation constant
- $M$: An integer number
- $k_{2x}$: Propagation constant of the second layer in the $x$ direction
- $\gamma_1$: Transverse propagation constant for medium 1
- $\gamma_3$: Transverse propagation constant for medium 3

The formulation for $k_{2x}$, $\gamma_1$ and $\gamma_3$ and can be written as:

$$k_{2x} = \sqrt{(k_0n_2)^2 - (\beta)^2} \quad (5)$$

$$\gamma_1 = \sqrt{(\beta)^2 - (k_0n_1)^2} \quad (6)$$

$$\gamma_3 = \sqrt{(\beta)^2 - (k_0n_3)^2} \quad (7)$$

$$\beta = k_0n_2 \sin \theta \text{ or } \beta = k_0n_{	ext{eff}} \quad (8)$$

A calculation program is developed to solve numerically the above equations by using a discretization step. The last values of the counters $N_1(i)$ and $N_2(i)$ will be considered as the effective indexes $(n_{	ext{eff}}_I)$ and $(n_{	ext{eff}}_II)$. The last values of the counters $N_1(i)$ and $N_2(i)$ will be considered as the effective indexes $(n_{	ext{eff}}_I)$ and $(n_{	ext{eff}}_II)$. The flowchart of the proposed calculation is given in Figure 2.
IV. SIMULATION RESULTS

In this part, a simple 1×2 MMI demultiplexer is designed and modelled by using different effective indexes calculated by NM and AM methods. Therefore, the first element of characterization is the MMI width, which is calculated according to self imaging effect. Figure 3 presents a simple MMI demultiplexer modelled by using a rib waveguide of SIO2 as a cladding layer, polymer as a film layer and a BK7 as substrate layer.

The guide and cladding effective indexes as function of the film thickness and rib height are shown in Figures 4 and 5, respectively. It is seen that these parameters are very sensitive to the structure dimension. Thus, the MMI guide should be accurately controlled as its fabrication tolerances are in the scale of nanometres. It can be clearly observed from the curves of Figure 5 that the cladding effective index is not affected by the rib height variation. Furthermore, it can be observed that the difference between the effective indexes of the two wavelengths becomes bigger for higher values of rib height.

Depending on the results of the first part, the selected rib height is 4 µm and the polymer layer width is 2µm. The effective index of the different layers is determined for the given dimensions; see Table 1.

<table>
<thead>
<tr>
<th>layer</th>
<th>$\lambda = 1530$ nm</th>
<th>$\lambda = 1565$ nm</th>
</tr>
</thead>
<tbody>
<tr>
<td>BK7</td>
<td>1.50091</td>
<td>1.50047</td>
</tr>
<tr>
<td>polymer</td>
<td>1.5556</td>
<td>1.5554</td>
</tr>
<tr>
<td>SIO2</td>
<td>1.52799</td>
<td>1.52750</td>
</tr>
<tr>
<td>$(n_{eff})_{II}$</td>
<td>1.548 (AM)</td>
<td>1.5477 (AM)</td>
</tr>
<tr>
<td></td>
<td>1.5482 (NM)</td>
<td>1.5479 (NM)</td>
</tr>
<tr>
<td>$(n_{eff})_{I}$</td>
<td>1.5359 (AM)</td>
<td>1.5353 (AM)</td>
</tr>
<tr>
<td></td>
<td>1.5356 (NM)</td>
<td>1.5350 (NM)</td>
</tr>
</tbody>
</table>

Table 1: The indexes and effective indexes of the different layers
As it is suggested by [12], the principal operation of MMI structures is the self imaging property of a multimode waveguide (MW). However, the necessary condition to separate two wavelengths such that \( \lambda_1 \) and \( \lambda_2 \) under the condition \( \lambda_1 < \lambda_2 \) is:

\[
L_{MMI} = pL_{\lambda_1} = (p+q)L_{\lambda_2}
\]

Where \( L_{\lambda_1} \) is the beat length for wavelengths \( \lambda_1 \), \( p \) is a positive integer and \( q \) is an odd integer.

As it is known, \( L_\pi \) increases as wavelength decreases therefore, \( p \) and \( q \) must be chosen to be as small as possible to optimize the structure dimension. The optimum values of \( p \) and \( q \) can be determined by varying the beat ratio \( \frac{L_{\lambda_1}}{L_{\lambda_2}} = 1 + \frac{q}{p} \) on the function of MMI width. The width \( w = 10.2 \, \mu m \) corresponds to a beat ratio equal to 1.0178 which leads to \( p = 56 \), \( q = 1 \) and \( L_{MMI} \) (\( \mu m \)) = 12675; determined by the numerical method (NM). The evolution of the input power along the propagation distance in the centre of slot region is illustrated in Fig. 6 for the wavelengths 1565 and 1535nm, respectively.

It can be noted that the wavelength separation in the two outputs is realized by taking the insertion loss and the extinction loss as the main parameters to characterise the model. Therefore, the model has an insertion loss = 0.97dB and 0.479dB, extinction loss = 4.703dB and 7.68dB for \( \lambda = 1565 \)nm and \( \lambda = 1530 \)nm, respectively.

The proposed model is designed by applying the analytic method, where \( (n_{eff})_{I}=1.5359 \), \( (n_{eff})_{II}=1.548 \) \( (\lambda=1530 \)nm) and \( (n_{eff})_{I}=1.5353 \), \( (n_{eff})_{II}=1.5477 \) \( (\lambda=1565 \)nm). The optimum values of \( p \) and \( q \) are found to be 58 and 1, respectively. The corresponding MMI width is \( W_{MMI}= 9.6 \, \mu m \) and coupling length is \( L_{MMI} = 11871 \)\( \mu m \). Therefore, the power distribution along the propagation distance is shown in Figure 7.

The difference between the effective indexes leads to different characterisation parameters, which confirm the dependence between the design characteristics of WDM components and the approach chosen for computing effective indexes. However, the wavelength separation demultiplexing is clearly noted in the ports with acceptable losses. The performance of the second model is very close to that designed by using the numerical method with insertion loss of 0.78dB, and 0.566dB and extinction loss of 7.432dB, and 9.05dB for \( \lambda = 1530 \)nm and \( \lambda = 1565 \)nm, respectively.

V. CONCLUSIONS

In this article, we investigated the design characteristics of a wavelength demultiplexer, which can be used for optical access networking. We considered a 1x2 wavelength demultiplexer model based on the multimode interference technique and the rib waveguide. In
order to analyze the impact of computation method on the design characteristics of this component, the effective index is computed by a numerical and analytic method. Consequently, we proposed a new algorithm for computing the numerical method. We can confirm that the design characteristics of WDM components depend on the approach chosen for computing effective indexes methods. In this work, we found out that the length and width of the model proposed are 12675 µm, 10.2µm and 11871µm and 9.6µm as determined by NM and AM, respectively. The values of the numerical method are close to those of the analytic scheme.

Figure 6: The power distribution vs. the propagation distance (NM): (a) For $\lambda=1565$nm, (b) for $\lambda=1530$nm
Figure 7: The power distribution vs. the propagation distance (AM): (a) \( \lambda = 1565 \text{nm} \), (b) \( \lambda = 1530 \text{nm} \)

REFERENCES


Design and Performance Evaluation of Service Overlay Networks Topologies

Davide Adami
CNIT Research Unit - Dept. Of Information Engineering,
University of Pisa, Italy
E-mail: davide.adami@cnit.it

Christian Callegari, Stefano Giordano,
Gianfranco Nencioni, Michele Pagano
Dept. Of Information Engineering,
University of Pisa, Italy
E-mail: {christian.callegari, stefano.giordano,
gianfranco.nencioni, michele.pagano}@iet.unipi.it

Abstract— Nowadays, Internet still lacks of an adequate support for QoS-sensitive applications, such as VoIP, Videoconference, and Video-on-Demand. To a large extent, this is due to the fact that Internet was originally designed to provide only a best-effort packet delivery service. In recent years, Service Overlay Networks (SONs) have emerged as a profitable way to address these issues without changing the underlying infrastructure. In this paper, we analyze the topology design problem of a SON from a performance point of view. Since the analytical solution of the problem is computationally too complex, we compare the performance of a limited set of well-known topologies. Based on heuristics, we also propose three new traffic demand-aware overlay topologies. Through extensive simulations, we investigate the performance of the candidate overlay topologies in different network scenarios, taking into account overhead and accepted traffic between the overlay nodes.

Keywords: SON, QoS, Topology Design.

I. INTRODUCTION

In the last decade, with the tremendous growth of multimedia technologies and the increasing popularity of real-time applications, the support of Quality of Service (QoS) in the Internet has been a great demand and several architectures to provide QoS (e.g., Integrated Services (IntServ) [1] and Differentiated Services (DiffServ) [2]) have been proposed by the IETF (Internet Engineering Task Force). Nevertheless, the deployment of these approaches is unlikely to be feasible in the long run, because IntServ has scalability problems, and DiffServ can only provide QoS with very coarse granularity. Moreover, appropriate business models are difficult to be introduced, because the Internet Service Providers (ISPs) are only concerned with providing QoS in their own administrative domains. As a result, the current Internet basically provides only a best-effort packet delivery service.

In recent years, Service Overlay Networks (SONs) have emerged as a profitable way to facilitate the deployment of QoS-sensitive applications, such as VoIP, videoconference, online gaming, etc. A SON is a virtual network built on top of an IP network, whose nodes, the overlay nodes, can be customized so as to build in complex features and to cooperate with each other in order to provide new services without any change to the routers of the underlying IP network. Pairs of overlay nodes are connected by means of logical links, also known as overlay links, which really are IP-layer paths, usually composed of one or more physical links. Moreover, an overlay path consists of one or more overlay links. A distinguishing characteristic of SONs is that the overlay links could be overlapped at the physical layer even though they are completely separated at the overlay layer. Non-overlay traffic or other overlay links may pass through a part or a whole group of IP-layer links. This means that the overlay links capacities are not fixed and cannot be controlled by the overlay nodes. To obtain satisfactory performance, the overlay nodes need continuously probing the network, so as to obtain updated information on the status and performance of the overlay links.

SONs can be classified as End-user Overlay Networks and Backbone Overlay Networks. The first type of SON [3][4] is constructed among the end hosts without any support from the intermediate network nodes. Though highly flexible, End-user Overlay Networks cannot guarantee end-to-end QoS, since the overlay links typically cross many intermediate Autonomous Systems (ASs), and the “uncontrolled” peering structure is unlikely to provide direct QoS support to the end-users. Due to these difficulties, the second type of overlay networks was introduced in several research works [5][6][7][8][9]. Usually managed by third parties, Backbone QoS Overlays may be classified in two different categories, based on the overlay nodes location: either only at the edge (see [5][6][9]), or also in the core of different domains (see [7][8]).

To optimize the performance and profitability of a SON, the first step is the selection of the best topology connecting the overlay nodes. In the following, we formalize the topology design problem taking into account the traffic demand among the overlay nodes as performance metric to be optimized. Since this problem is computationally too complex, we investigate the performance and overhead of several candidate SON topologies by varying the number of overlay nodes and the IP network size. More specifically, in this work we take into account three different models for the topology at IP-layer (flat,
hierarchical, real) and we compare the performance of seven overlay topologies built on top of those IP networks: four are well-known since they have been already introduced in the literature, whereas the other three (K-Shortest-Path-Tree, Pruned Adjacent-Connection and Demand-aware Adjacent-Connection) are completely new.

In the performance evaluation, we have considered as the “best” overlay topology the one that, on equal terms of accepted traffic and performance, has the lowest overhead, due to the overlay network maintenance traffic.

This paper extends the preliminary results presented in [10], by the introduction of two novel overlay topologies (Pruned Adjacent-Connection and Demand-aware Adjacent-Connection) and a new set of experimental results.

The paper is structured as follows. Section II discusses the related works. Section III introduces the topology design scenario and formalizes the topology design problem. Section IV presents some existing SON topologies as well as the new overlay topologies. Section V describes the simulation scenario and discusses the results of some simulations carried out to investigate the impact of the overlay topologies on performance and overhead. Finally, conclusions are presented in Section VI.

II. RELATED WORKS

Overlay topology design has been one of the most challenging research areas over the past few years. When considering different topologies, it is necessary to understand how they affect the overlay routing performance and how to efficiently build overlay topologies connecting all the overlay nodes. Several works have focused on the selection of the best overlay links (e.g. [11]), but other issues, such as binding end systems to overlay access nodes [11], positioning the overlay nodes [12][13] or choosing the right number of overlay nodes [13], have also been faced. In these studies, the overlay topology is usually represented as a graph and the topology design problem is expressed as an optimization problem. The general approach relies on the use of heuristic algorithms that allow finding a near-optimal solution.

Most works [11][12][14][15][16] analyze the topology design problem from an economic point of view with the aim to minimize the cost for the deployment of the SON. Only a few works [17][18] deal with the SON topology design problem from a network performance perspective. In [17], the authors aim at finding the overlay topology minimizing a cost function which takes into account the overlay link creation cost and the routing cost. They also highlight how the traffic demand affects the creation of new overlay links. In [18], instead, the authors compare several existing and some new overlay topologies in terms of resilience.

Our approach is similar to the one used in [18]. However, apart from introducing new overlay topologies, we compare several overlay topologies from a QoS (bandwidth, delay) performance point of view.

III. PROBLEM STATEMENT

This work is focused on Backbone QoS Overlays with intra-domain nodes (Figure 1), where the overlay nodes can be placed either at the edge or in the core of each domain.

In the following, we do not provide a more detailed description of this architecture, because it is out of scope. The chosen architecture, indeed, is only an instance used to evaluate the overlay routing performance when varying the overlay topology.

The experimental results obtained in this work are also valid in the case of Backbone QoS Overlays (see Figure 2) with Inter-domain nodes.

In the following, we suppose that:

- The location of the overlay nodes is pre-determined.
- The metric associated to each overlay link is the delay and the overlay path between a pair of overlay nodes is selected by using the Dijkstra algorithm.
- Each overlay path is composed of IP-layer links. At IP layer, the cost of each link is assigned in inverse proportion to the link bandwidth and the shortest path between a pair of IP nodes is computed by using the Dijkstra algorithm.
Therefore, the overlay topology construction problem can be formulated as follows.

Let us consider:
- The IP-layer topology $G_r(V_r, E_r)$ where $V_r$ is the set of nodes.
- $E_r$ is the set of IP layer links.
- $N$ is the total number of $V_r$ nodes.
- A set of overlay nodes, $V_o$, which is a subset of $V_r$.
- $M$ is the total number of $V_o$ nodes.
- The IP-layer path-link indicator function $P_{ij}^{mn}$, where $P_{ij}^{mn} = 1$ if the IP-layer path between $m$ and $n$ includes the IP-layer link $ij$.
- The overlay path-link indicator function $Q_{mn}^{xy}$, where $Q_{mn}^{xy} = 1$ if there is an overlay path from $x$ to $y$ that includes the overlay link $mn$.
- The delay of the IP-layer link $ij$, $D_{ij}$.
- The traffic demand between the overlay nodes $x$ and $y$, $T_{xy}$.

The goal is to find the topology $G_o(V_o, E_o)$ (i.e., a sub-set of overlay links, $E_o$) that minimizes the cost function:

$$\sum_{ij} T_{ij} \sum_{mn} Q_{mn}^{xy} \sum_{ij} D_{ij} P_{ij}^{mn}$$

in which the end-to-end delay is weighted by the traffic demand, with the constraints:

- $\sum e_{mn} \leq d$, where $d$ is the maximal node degree and $e_{mn} = 1$ if an overlay link between $m$ and $n$ exists, $e_{mn} = 0$ otherwise.
- $\sum_{mn} Q_{mn}^{xy} \sum_{ij} D_{ij} P_{ij}^{mn} \leq \delta$, where $\delta$ is the overlay paths delay constraint.
- $\sum_{mn} T_{xy} Q_{mn}^{xy} \leq \tau_{mn}$, where $\tau_{mn}$ is the available bandwidth of the overlay link $mn$.

It is worth highlighting that if popular or greedy nodes exist, the topology layout does not change, but such nodes are located at the centre, so as to minimize their latency with respect to the other nodes (see [19] for further details).

The degree constraint attends to limit the overhead associated to the overlay link monitoring (e.g., delay, available bandwidth, overlay traffic demand). Thus, if the node degree is higher, the overhead is greater.

IV. OVERLAY NETWORK TOPOLOGIES

The problem formulated in the previous section is NP-hard and therefore it is too complex from a computational point of view. The basic idea behind our approach is to interconnect the overlay nodes by means of the most common network topologies and to choose the best topology with respect to a set of performance metrics.

This section lists the main features of several candidate topologies that appeared in literature. Moreover, three new overlay topologies are proposed.

A. Full-Mesh (FM)

In the FM overlay network topology, each node is adjacent to all the other nodes at overlay layer. Therefore, every node has the same number of neighbours, i.e. the same node degree ($d$). As said previously, to retrieve information on the status and performance of the overlay links, every node has to periodically send probing packets to all the neighbours at overlay layer. The FM topology is therefore characterized by the highest overhead, due to the overlay monitoring traffic. An example of FM topology is shown in Figure 3.

![Figure 3 Full-mesh overlay topology](image)

B. K-Minimum Spanning Tree (KMST)

A minimum spanning tree (MST) is the lowest cost tree among all the candidate trees connecting a given set of nodes. A KMST overlay topology is composed of $K$ MSTs, where the $k$-th MST of the composite graph is the MST of the initial graph excluding the edges of the previously computed MSTs. Therefore, the overlay links of the $K$ trees are not overlapped. The $K$ value can be chosen as a trade-off among cost, performance and node degree constraint.

This approach has been proposed in [20] so as to minimize the overhead due to the amount of information exchanged for link monitoring.

Figure 4 shows a 2MST, where the two trees are depicted with dashed and solid lines, respectively.
C. Mesh-Tree (MT)

This topology (called Mesh-Tree in [18]) is obtained by joining the MST connecting all the overlay nodes with the set of overlay links connecting the overlay nodes that have a grandchild-grandparent or uncle-nephew relationship in the MST. Figure 5 shows an MT overlay topology: the MST is represented with solid lines, whereas links joining nodes with a grandchild-grandparent or uncle-nephew relationship are dashed.

D. Adjacent-Connection (AC)

The Adjacent Connection (AC) topology relies on the knowledge of the IP-layer topology. The construction of the AC topology, proposed in [8] and [21], is based on the following condition: if no overlay node is directly connected to the nodes belonging to the IP-layer path between any pair of overlay nodes, an overlay link connecting this pair of overlay nodes is created. Figure 6 shows an example of AC overlay topology.

E. K-Shortest Path Tree (KSPT)

The K-Shortest Path Tree (KSPT) is the first novel topology we introduce in this paper. It is constructed taking into account the traffic demand among the overlay nodes. A SPT is made up of minimum cost paths from a node (root) towards all the other nodes. In our case, the metric is the overlay link delay. The choice of the K value, like in KMST, results from a trade-off among cost, performance and node degree constraints.

A KSPT topology can be constructed as follows:

1. Initialize $K_{th} = \text{ceiling}[M/3]$, $K = 0$
2. Create a SPT for each overlay node.
3. Select the SPT whose root corresponds to the overlay node with the greatest traffic demand.
4. Add the SPT to the topology layout. Increment the value of $K$ ($K = K + 1$).
5. Calculate the average node degree. If it is greater than the node degree constraint or $K_{th}$ is reached, end. Otherwise, continue.
6. Select the SPT with minimum overlap with the created topology. Go to step 4.

Figure 7 shows a 2SPT, where the two trees, with roots G and C, are depicted by using dashed and solid lines, respectively.
F. Pruned Adjacent-Connection (PAC)

The Pruned Adjacent-Connection (PAC), is a novel topology we propose and it is constructed exploiting the results presented in [10]. PAC is a modified version of the AC topology, where some overlay links are deleted so as to reduce the node degree. In more detail, an overlay link is deleted if it connects two nodes with the highest degree. The number of overlay links to be deleted is determined by a fixed degree threshold, $T_h$, which is proportional to $M$.

A PAC topology can be constructed as follows:

1. Create an AC topology and determine the degree of each overlay node.
2. Select the overlay node, $i$, with the highest degree.
3. Select the neighbour of $i$, $j$, with the highest degree.
4. Delete the overlay link between $i$ and $j$, and update the node degree.
5. Calculate the average node degree. If it is greater than the threshold $T_h$, go to step 2. Otherwise, end.

Figure 8 shows an example of PAC overlay topology.

![Figure 8 Pruned adjacent-connection overlay topology](image)

G. Demand-aware Adjacent-Connection (DAC)

As the previous one, the Demand-aware Adjacent-Connection (DAC), is a new topology constructed based on the results obtained in [10]. However, DAC allows to take also into account the traffic demand among the overlay nodes. DAC, like PAC, is a modified version of the AC topology, where some overlay links are deleted to reduce the node degree. In more detail, an overlay link is deleted if it interconnects two nodes with the highest traffic demand. Also in this case, the number of overlay links to be deleted is determined by a fixed degree threshold, $T_h$, proportional to $M$.

A DAC topology can be constructed as follows:

1. Create an AC topology, determine the degree of each overlay node, and calculate the aggregate traffic demand of each overlay node, as the sum of the incoming and outgoing traffic.
2. Select the overlay node, $i$, with the lowest aggregate traffic demand and whose degree is greater than the threshold $T_h$.
3. Select the neighbour of $i$, $j$, with the lowest aggregate traffic demand and with a node degree greater than the threshold $T_h$.
4. Delete the overlay link between $i$ and $j$, and update the node degree.
5. Calculate the average node degree. If it is greater than the threshold $T_h$ and $i$ and $j$ exist, go to step 2. Otherwise, end.

Figure 9 shows an example of DAC overlay topology.

![Figure 9 Demand-aware adjacent-connection overlay topology](image)

V. PERFORMANCE ANALYSIS

A. Simulations settings

To compare the performance of the topologies described in the previous section, we considered flat and hierarchical IP-layer topologies generated by means of BRITE [22].

In the flat topology, the generation model to interconnect the nodes is based on the Waxman's probability [23]:

$$ P (u, v) = \alpha e^{-\frac{d}{\beta L}} $$

where $P (u, v)$ is the probability that a link between the nodes $u$ and $v$ is created, $\alpha (0 < \alpha \leq 1)$ and $\beta (0 < \beta \leq 1)$ are Waxman specific parameters (in our simulations $\alpha=0.03$, $\beta=0.03$), $d$ is the Euclidean distance between the nodes, and $L$ is the maximum distance between any two nodes. The link bandwidth has been assigned according to a uniform distribution in the range [10, 1000] Mb/s. It is worth highlighting that this value does not represent the IP-layer link capacity, but the available bandwidth, since non-overlay traffic may pass through the same links. The link delay has been assigned in proportion to the link length.

To create a hierarchical random topology, a top-down approach that consists of the following three steps has been adopted:
1. Generate an AS-level topology according to the Waxman’s model (for redundancy reasons, we set \(\alpha = 1\)).

2. For each AS, generate a router-level topology by using the Waxman’s model with the same parameters as for the flat topology.

3. Finally, use the random method to interconnect ASs as dictated by step 1: if \((i, j)\) is a link in the AS-level topology, then two nodes \((u, v)\), randomly chosen from ASs \((i,j)\), are interconnected at 512 Mb/s.

Simulations have been carried out randomly associating overlay nodes with the generated IP-layer nodes and interconnecting them through one of the topologies described in Section IV.

For a fair comparison of the candidate topologies and to also meet the node degree constraint \(d\), other research works modify the construction of the topology as follows: if an overlay node degree exceeds \(d\) in the resulting overlay topology, only the \(d\) closest neighbours overlay nodes are maintained, while the others are pruned. Since this approach dramatically affects the intrinsic nature of each topology, our choice was to consider the overlay node degree as a primary performance metric without changing the resulting overlay topology.

Moreover, the traffic demand is generated according to a uniform distribution normalized with respect to \(M\), so that the overall traffic demand is statistically the same when the number of overlay nodes changes.

A customized simulation environment has been developed by using C programming language.

Three simulation scenarios have been considered:
1. The IP-layer topology is flat \([N=(100, 150, 200, 250, 300)]\) and the number of overlay nodes changes \([M=(10, 20, 40, 60, 80\%)*N]\).
2. The IP-layer topology is hierarchical \([N=(100, 200, 300, 400, 600)]\), Number of ASs=4, Number of nodes per-AS=\([25, 50, 75, 100, 150]\) and the number of overlay nodes changes \([M=(10, 20, 40, 60, 80\%)*N]\).
3. A case study, corresponding to a “real network scenario” with a large hierarchical IP-layer topology \([N=600, \text{Number of ASs}=4, \text{Number of nodes per AS}=150]\) and a small number of overlay node \([M=(1, 2, 4, 6, 8\%)*N]\).

The average value and the 99% confidence interval of each performance metric have been evaluated in the three scenarios by performing a set of 50 independent simulations for each pair \((N, M)\).

**B. Performance Metrics**

To compare the performance of different overlay topologies, we introduce four parameters, each of them highlighting a specific topology feature.

- **Bandwidth Rejection Ratio (BRR)**
  This performance parameter represents the probability that an overlay path with bandwidth and delay requirements cannot be created due to bandwidth unavailability:
  \[
  BRR = \frac{\text{Number of bandwidth rejected end-to-end overlay paths}}{\text{Total number of end-to-end overlay paths}}
  \]

- **Delay Rejection Ratio (DRR)**
  This performance parameter represents the probability that an end-to-end overlay path with bandwidth and delay requirements cannot be created due to delay limitations although the bandwidth constraint could be satisfied:
  \[
  DRR = \frac{\text{Number of delay rejected end-to-end overlay paths}}{\text{Number of bandwidth accepted overlay paths}}
  \]

- **Average Node degree (AND)**
  This performance parameter provides information on the topology overhead. If \(d_i\) is the neighbours number of the \(i\)-th overlay node, AND can be defined as follows:
  \[
  \text{AND} = \frac{\sum_{i=1}^{M} d_i}{M}
  \]

- **Accepted Traffic Weighted Delay (ATWD)**
  This performance parameter describes the capability of the topology to associate low delays to the highest traffic demands:
  \[
  \text{ATWD} = \frac{\sum \{\text{Accepted traffic demand between the nodes } i, j \times \text{Delay}_{ij}\}}{\text{Total accepted traffic}}
  \]

It is worth emphasizing that the ATWD is strictly related to the cost function defined in Section III. Indeed, the only difference is that the ATWD is normalized with respect to the accepted traffic. This normalization is necessary for a fair comparison, because the accepted traffic varies with the overlay topology.

**C. Simulations Results**

A different graph has been worked out for each performance metric in case of flat and hierarchical topologies. In the following, we report the simulation results concerning AND, BRR and DRR only for the largest size of the IP-layer network, whereas as regards ATWD the results obtained for the smallest size of the IP-layer network are also reported, since the ATWD performance significantly varies with the number of IP-layer nodes.

Since the experimental results obtained in the flat topology cases (see the following sub-section), show that KSPT outperforms the other topologies, the simulations with PAC...
and DAC topologies are only performed when the IP-layer network is hierarchical and in the “real network scenario”. The behaviour of both topologies is analyzed with two values of the degree threshold: $T_h(M/3, M/2)$ in the hierarchical IP-layer network case, $T_h(M/2, 2M/3)$ in the “real network scenario” case, respectively.

- Flat IP-layer Network Model

Figure 10 shows the AND trend in a flat IP-layer network (N=300) when M ranges from 0.10*N to 0.80*N. It is worth highlighting that the behaviour of this metric is not affected by the size of the IP-layer network, so these results may also be extended to IP-layer networks with a different number of nodes.

The graph outlines that MT always has the minimum node degree that remains constant independently of M.

Moreover, the AND of KMST and KSPT increases almost linearly with M, but KSPT outperforms KMST due to the different way K is selected and the mechanism used to limit the value of K when the AND is too high.

Concerning the AC overlay topology, at the beginning, the AND is the highest one and increases up to reaching a maximum value when M=0.20*N. Afterwards, it starts decreasing and, when M ≥ 0.60*N, AC outperforms not only KMST, but also KSPT. Indeed, when M increases, it is more likely that another overlay node exists along the path between whichever pair of overlay nodes and therefore the number of overlay links decreases. The AND of FM (not reported in the graph) is always equal to M−1.

Figure 11 reports the BRR average value and confidence interval for N=300. As outlined in the graph, FM, KMST, AC and KSPT show a similar behaviour. MT, instead, always has the highest BRR. This result is in accordance with the node degree trend: since MT has the lowest node degree, also the overall available bandwidth is the lowest.

Figure 12 shows the DRR trend. Also in this case, FM, KMST, KSPT and AC show a similar trend that decreases with M. Instead, MT has the worst performance, since its DRR values are always significantly higher than the ones obtained for the other topologies when M ≥ 0.20*N. This is due to the lower node degree of the MT topology.
Finally, let us consider the ATWD parameter. By definition, this metric has low values when the overlay topology favours the creation of overlay paths with the lowest delay between the overlay nodes that exchange the largest amount of traffic. In a small size flat IP-layer network (see Figure 13), KSPT and MT perform better than the other topologies. On the contrary, as shown in Figure 14, in a large size flat IP-layer network MT sharply outperforms FM, KMST, KSPT and AC.

From the simulation results, we can infer that MT performs worse than the other overlay topologies because, although it is characterized by the lowest AND and ATWD metrics, the BRR has the highest values (see Figure 11) and therefore most of the traffic demand is rejected. In flat IP-layer networks, the overlay topology should be chosen based on the number of overlay nodes. If the number of both overlay and IP nodes is small, KMST is the best overlay topology. Instead, if the number of overlay nodes is small, but the IP-layer network size is large, the best overlay topology is KSPT. When the number of overlay nodes is large, AC is always the best overlay topology, because it performs in the same way as the other overlay topologies, but its AND is the lowest.

**Hierarchical IP-layer Network Model**

In a hierarchical IP-layer network, the AND behaviour (see Figure 15) is similar to that obtained for a flat IP-layer network. Nevertheless, the graph outlines that AC always has a lower AND than KMST. PAC and DAC allows reducing the AND only when $M=0.10\times N$ and the reduction is in inverse proportion to $T_\theta$.

Figure 16 and Figure 17 show the BRR and DRR behaviours for $N=600$, respectively. Also in this case, the trends are similar to those obtained for flat IP-layer topologies and MT performs worse than all the other candidate topologies. However, it is relevant to highlight that, apart from MT, KSPT has the highest BRR and DRR and the performance difference among KSPT and FM, KMST, AC is greater for small and middle size overlay networks.

The BRR of PAC and DAC is the same as FM, KMST, and AC, instead the DRR is higher. When $T_\theta=M/3$ PAC and DAC perform worse than KSPT. Basically, DAC performs better than PAC, while DAC ($T_\theta=M/2$) performs like FM, KMST, and AC.
performance improvement as it occurred in case of a flat topology.

PAC and DAC perform better in small size hierarchical IP-layer networks and with a higher degree threshold.

The graph outlines that the AND increases almost linearly with M for all the overlay topologies except for MT, which has the minimum AND that remains constant independently of M.

The AND of AC always have the maximum node degree. To be noted that PAC and DAC reduce the AND of AC. When the degree threshold is equal to 2+M/3, PAC and DAC have almost the same AND as KMST.

Regarding KSPT, it outperforms KMST due to the reason explained in the previous sections.

In summary, also in hierarchical IP-layer networks, MT underperforms the other overlay topologies for the same reasons as in case of a flat network. As far as the remaining topologies are concerned, AC outperforms all the other topologies, due to the “knowledge” of the underlay topology which allows AC to reduce the node degree and, as a result, the overhead. PAC and DAC are two valuable alternatives to AC to further reduce the node degree, but a performance worsening occurs. More specifically, DAC performs better than PAC, highlighting the advantages of building traffic demand-aware topologies. Finally, the results of the simulations outline that KSPT is not a suitable topology for hierarchical IP networks, since it always performs worse than the other overlay topologies (except MT).

- Case study: real network scenario

Figure 20 shows the AND trend in a “real network scenario when M ranges from 0.01*N to 0.08*N.

Figure 21 reports the BRR average value and confidence interval. As outlined in the graph, all the topologies show a similar decreasing behaviour. When M > 0.01*N, BRR is very low, because the overall traffic demand is low in this scenario. When M = 0.01*N the bandwidth request for each node is high due to the way used for generating the traffic matrix (see Section V.A) and since it overloads the links, BRR assumes a significant value.

Figure 22 shows the DRR behaviour. In this case, KMST and AC show a similar trend that decreases with M. Instead, MT has an increasing trend. To be noted that KSPT has the worst performance, while PAC and DAC have a worse DRR than AC, due to the introduction of the node degree threshold.
VI. CONCLUSIONS

In this paper, the topology design problem of SONs with performance requirements has been addressed taking into account both the traffic demand and the overhead. This problem has been theoretically formalized, but since it is computationally too complex, the performance of some reference overlay topologies has been compared and three new topologies, called KSPT, PAC, and DAC have been introduced. Through extensive simulations, the performance and overhead of each overlay topology have been investigated either when the IP-layer network model is flat or hierarchical and in a real network scenario.

The results presented in this paper show that MT always performs worse than the other overlay topologies. Moreover, when the size of the flat IP-layer network is large and the number of overlay nodes is small, KSPT is a valuable option.

Instead, when the IP-layer network is hierarchical, the AC topology outperforms all the other network topologies, because it takes advantage of the underlay topology knowledge. Moreover, PAC and particularly DAC are two possible alternatives to the AC topology to reduce the node degree.

Finally, in a real network scenario with a large amount of available bandwidth, KMST outperforms all the other overlay topologies and MT does not underperform the other overlay topologies as in the previous cases.

REFERENCES


Davide Adami received a degree in Electronic Engineering from the Department of Information Engineering at the University of Pisa, Italy, in 1992. From 1993 to August 1997 he worked at Consorzio Pisa Ricerche in the field of computer networks. In September 1997 he joined CNIT (National Consortium for Telecommunications), where he is a senior researcher in the field of telecommunications networks. His research interests mainly concern the provisioning of Quality of Service in IP networks and the TCP/IP enhancements for satellite networks. He is one of technical responsible of the CNIT satellite network. Currently, he is also an Assistant Professor at the Faculty of Engineering of the University of Pisa in the courses of “Planning and Simulation of Telecommunications Networks” and “Systems and Services of Telecommunications Networks”.

Christian Callegari received his Laurea degree in Telecommunication Engineering “cum laude” on October 2004 from the University of Pisa, discussing a thesis titled “Simulative analysis of RSVP-TE, and valutation of end-to-end rerouting techniques in MPLS networks”. In 2005 he obtained the qualification to practise the profession of Engineer and he is a member of the IEEE Communication Society. On January 2005 he joined the department of Information Engineering at the University of Pisa as a Ph.D. student, and in 2008 he obtained his PhD in Information Engineering. He is actually a Postdoctoral fellow in the Telecommunication Network research group at the department of Information Engineering of the University of Pisa.

Stefano Giordano received the Laurea degree “cum laude” in Electronics Engineering and the PhD degree in Information Engineering from the University of Pisa in 1990 and 1994 respectively. He worked with Consorzio Pisa Ricerche since 1990 in the field of Telecommunication Networks. Since the end of 2001 he is associate professor at the Department of Information Engineering of the University of Pisa (Telecommunication Networks Group) where he give lectures on “Telecommunication Networks”, and “Design and Simulation of Telecommunication Networks”. He is responsible of the Telecommunication Networks Group at Consorzio Pisa Ricerche. Stefano Giordano is member of the Scientific and Technical Committee of the Center for the ATM Campus Network of the University of Pisa (SERRA), of the IEEE Communication Society since 1989, of the Internet Society since its foundation, of the IFIP Working Group 6.3.His research and professional areas of interest are Broadband Communications, Telecommunication Networks Analysis and Design, Simulation of Communication Networks and Systems, Multimedia communications.

Gianfranco Nencioni received his Laurea degree in Telecommunication Engineering “cum laude” in September 2008 from the University of Pisa, discussing a thesis titled “Design and simulative analysis of topologies for Service Overlay Networks”. In 2009 he obtained the qualification to practise the profession of Engineer. In December 2008 he overcame a Ph.D. contest in Information Engineering at the University of Pisa. He is actually a Ph.D. student at the Telecommunication Networks Research Group of the Department of Information Engineering at the University of Pisa. His research activity regards “Green Networking” techniques, in particular the energy-aware routing and network design.

Michele Pagano received laurea (cum laude) in Electronics Engineering in 1994 and a Ph.D. in Electronics Engineering in 1998, both from the University of Pisa. Since 2007, he is an associate professor at the Dipartimento di Ingegneria dell’Informazione of the University of Pisa, where he is the official instructor of the courses of “Telematics”, “Performance of Multimedia Networks” and “Network Security”. His research interests are related to statistical characterization of traffic flows and to network performance analysis, mainly in the framework of architectures able to support Quality of Service. Performance evaluation has been carried through analytical approaches as well as by means of discrete event simulation. Finally, a new research field is represented by network security issues, mainly in the framework of Intrusion Detection Systems. He has co-authored around 100 papers published in international journals and presented in leading international conferences.

© 2011 ACADEMY PUBLISHER
An Empirical Evaluation of Multi-Step Prediction Performance

Mohamed Faten Zhani
Department of Computer Science
University of Québec in Montréal
zhani.mohamed_faten@courrier.uqam.ca

Halima Elbiaze
Department of Computer Science
University of Québec in Montréal
elbiaze.halima@uqam.ca

Farouk Kamoun
National School of Computer Sciences
Manouba 4010, Tunisia
frk.kamoun@planet.tn

Abstract—Traffic prediction constitutes a hot research topic of network metrology. MultiStep ahead prediction allows to predict more values in the future. Then, the result can be used to act proactively in many prediction applications. In this work, the AutoRegressive Integrated Moving Average (ARIMA) model and the linear minimum mean square error (LMMSE) are used for multiStep predicting. Via experimentation on real network traffic, we study the effect of some parameters on the prediction performance in terms of error such as the number of last observations of the throughput (i.e. lag) needed as inputs for the model, the data granularity, variance and packet size distribution. We also compared two multi-step prediction techniques: the Iterating Multi-Step technique (IMS) and the Direct Multi-Step technique (DMS). Besides, we performed a set of predictions based on packets size. Unexpectedly, we find that using more than two lags as inputs for the prediction model increases the prediction error. Using the last observation as the predicted value provides the same 1-step prediction performance as ARIMA or LMMSE model. The ARIMA model provides an acceptable multi-step prediction performance. Experimental results show that there is a granularity value at which the multi-step prediction is more accurate. They also show that the IMS technique provide more accurate traffic prediction than the DMS technique. We also find that the prediction of classified packets based on their size is possible. Especially, throughput of 1,500-byte packets is the less predictable.

Index Terms—Traffic analysis, traffic modeling and prediction, multi-step prediction, IMS, DMS, ARIMA, LMMSE

I. INTRODUCTION

The predictability of network traffic is of significant interest in many domains. We can distinguish two categories of prediction: long and short period predictions. Traffic prediction for long periods provides a detailed forecasting of the workload and traffic patterns to assess future capacity requirements, and therefore allows for more accurate planning and better decisions [1], [2]. Short period prediction (milli-seconds to minutes) is relevant for dynamic resource allocation. It can be used to improve the Quality of Service (QoS) mechanisms as well as congestion and resource control by adapting the network parameters to traffic characteristics [3]–[5]. It can also be used for routing packets and active queue management schemes [6], [7].

Traffic prediction has been extensively investigated since the discovery of the self-similar and the long-range dependence nature of networks traffic [8]–[11]. While these characteristics cause dramatic effects on network performance in terms of loss and delay, several studies have shown that they can be exploited to predict the traffic in order to control the network resources assignment [2], [7], [12]–[16].

Inspite of all works, model selection is still an uncertain procedure. In this paper, two linear models are investigated, namely the AutoRegressive Integrated Moving Average Model (ARIMA) and the Linear minimum mean square error model (LMMSE).

Even though they are linear models, they are usually used in literature [1], [2], [7], [15]–[21]. Other models like fractional models or non-linear models which capture long-range dependence are more complicated in terms of complexity and do not provide a high improvement on the prediction accuracy [18], [20]. Besides, linear models are easier to implement for online systems.

Many prediction issues are still not resolved such as the effect of the Internet traffic characteristics (granularity, correlation, packet size distribution...) and the input parameters of the model (like the number of lags). Multi-step prediction is also an important issue which was not treated in the case of Internet traffic. Multi-step prediction consists in predict the next values many steps in the future. There are two techniques of multi-step predictions. The first is the iterating multi-step technique (IMS) and the second is the direct multi-step technique (DMS).

This paper focuses on resolving these issues. It is organized as follows. Section II introduces selected related work on traffic prediction. Section III presents the prediction methodology, the one-step ahead and the multi-step prediction techniques. It also provide detailed descriptions of the ARMA and the LMMSE prediction models. Section IV describes the used network traces. Section V discusses the experimental results. The conclusions and future work are presented in Section VI.

II. RELATED WORK

Since the discovery of the self-similarity characteristic of network traffic, many researchers have investigated Internet traffic prediction. He et al. have shown that the correlation structure present in self-similar traffic can be detected on-line and used to predict future traffic [16]. Hence, they define a scheme, called TCP with traffic prediction (TCP/TP) that uses the prediction results to infer the optimal point at which a TCP connection should operate.
Sang et al. have proposed an approach to make predictability analysis of network traffic [17]. The approach assesses the predictability of network traffic by considering two metrics: (1) how far into the future a traffic rate process can be predicted with bounded error; (2) what is the minimum prediction error over a specified prediction time interval. The authors have used two stationary traffic models: the ARMA model and the Markov-Modulated Poisson Process. They have argued that the two models, though both short-range dependent, can capture statistics of self-similar traffic quite accurately.

Yang et al. have attempted to improve the least-mean square (LMS) predictor so-called Error-adjusted LMS (EaLMS) [22]. The main idea of EaLMS is using previous prediction errors to adjust the LMS prediction value, so that the prediction delay could be decreased. The authors have used traffic obtained by smoothing real traffic assuming that it preserves the main characteristic of original traffic.

He et al. have focused on predictability of large transfer TCP throughput [23]. They classified TCP prediction techniques into two categories: Formula-based (FB) and History-Based (HB). FB prediction relies on mathematical models that express the TCP throughput as a function of the characteristics of the underlying network path (round trip time, number of flows etc.). HB techniques predict the throughput measurements on the same path, when such a history is available. It has been shown that HB predictors are quite accurate but are highly path-dependent; whereas, FB predictors are accurate only if the TCP transfer is not saturating the underlying path [23].

Zhani et al. have studied the effect of some parameters on the prediction performance in terms of error using a neurofuzzy model (α_SNF) and the ARIMA model as prediction models [24]. They have classified prediction techniques into two categories: Training-Based (TB) techniques and Non-Training-Based (NTB) techniques. Specifically, the TB techniques need a training phase. The training phase consists of identifying model parameters based the history of the throughput measurements called the training data set. The TB model is then fed by the last observations of the throughput called lags in order to predict the future value. They found that the complexity of the training phase is not crucial since it is performed once. Contrarily, NTB techniques do not need training phase and calculate the predicted value using only the last lags. NTB techniques does not require prior knowledge of the correlation structure of the time series. Authors also investigated the use of exogenous variables as inputs for the model. Exogenous variables are variables which are different from the lags such as the number of packets or sampled data. Experimental results show that the models, identified with small dataset and using only one lag, can provide accurate prediction. They also show that counts of packets and especially large packets can be used to efficiently predict the throughput.

Other work in the domain of Internet traffic forecasting addresses long period predictions that are important for IP network capacity planning [1], [2].

Growschwitz et al. used time series analysis to create detailed forecasts of future NSFNET backbone traffic [1]. The resulting integrated autoregressive moving average model (ARIMA) made quite accurate forecasts of traffic levels up to a year in advance.

Papagiannaki et al. introduced a methodology to predict when and where link additions/upgrade have to take place in an IP backbone network [2]. They show that IP backbone traffic exhibits visible long term trends, strong periodicities, and variability at multiple time scales. Their methodology relies on the wavelet multiresolution analysis and linear time series models (ARIMA). They show that forecasting the long-term trend and the fluctuations of the traffic at the 12 hour time scale yields accurate estimates for at least six months in the future.

Other work in other domain like econometrics investigated the multi-step prediction techniques. There are two multi-step prediction techniques: the Iterating Multi-Step technique (IMS) and the Direct Multi-Step technique (DMS) [25]–[27]. The results of these papers are very different when comparing these two techniques. In fact, the performance of these techniques vary depending on the nature of the predicted data. We note the Internet traffic data is very different from other data since it is very variable [16]. These techniques are detailed in Section III.

This work provides many empirical tests showing the effects of many parameters on the prediction models. It is an extension of our previous work done in [28]. It deals with the multistep prediction techniques namely IMS and DMS techniques and with the prediction based on the packet size. To the best of our knowledge, none has analyzed yet the performance of the IMS and DMS multi-step techniques applied to a real network traffic.

III. PREDICTION METHODOLOGY

In what follows, we introduce the prediction methodology considered in this work. We present the notation used in this paper as well as the definition of one-step and multi-step prediction. The prediction models are then presented.

A. Prediction

In this work, we investigate the one step prediction as well as the multi-step prediction.

- Aggregated traffic and granularity:

Let \( y(t) \) be the rate at the time \( t \) that arrives at a router in a sampling interval or granularity \( g \).

We first define \( y^m(t) \) as the aggregate series samples of order \( m \):

\[
y^m(t) = \frac{1}{m} \sum_{i=(t-1)m+1}^{tm} y(i)
\]

As \( m \) increases, the \( y^m(t) \) represents the throughput at the granularity \( m \times g \).

- One-step ahead prediction:
In this case, it is desired to predict 1 period ahead from an end-of-sample \( y^m(t-1) \) that is to predict \( y^m(t) \) noted \( y^m_1(t) \). We have:

\[
y^m_1(t) = f(y^m(t-n),...,y^m(t-i),...,y^m(t-1)) \tag{2}
\]

where \( f \) is the prediction model, \( n \) is the number of lags i.e. last observations of \( y^m(t) \) used as inputs for the model, and \( m \) is the aggregation level.

- **Multi-step ahead prediction**

In this case, it is desired to predict \( s \) periods ahead from an end-of-sample \( y^m(t-1) \). That is to predict \( y^m(t),...,y^m(t+i-1),...,y^m(t+s-1) \) noted by \( y^m_1(t),...,y^m_i(t+i-1),...,y^m_i(t+s-1) \) where \( i \) starts from 1 to \( s \). Let’s note \( y^m_s(t) \) the \( s \)th step prediction of \( y^m(t) \).

There are two multi-step prediction techniques: the **Iterating Multi-Step Technique (IMS)** and the **Direct Multi-Step Technique (DMS)**.

The IMS technique consists in iterating one-step ahead prediction. Thus, we have:

\[
y^m_s(t) = f(y^m_{(s,n)}(t-n),...,y^m_{(s,i)}(t-i),...,y^m_{(s,1)}(t-1)) \tag{3}
\]

where

\[
\sigma(s, i) = \begin{cases} 
0 & \text{if } s \leq i \\
-s + i & \text{if } s > i 
\end{cases}
\]

where \( f \) is the prediction model, \( n \) is the number of lags used by the model and \( s \) is the prediction step ahead. Note that \( y^m_0(t) \) (i.e. when \( s = 0 \)) is the real value \( y^m(t) \). Thus, the one-step-ahead prediction formula defined in Eq.2 is valid.

<table>
<thead>
<tr>
<th>Prediction step</th>
<th>Inputs of the model</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( y^m_0(t-3), y^m_0(t-2), y^m_0(t-1) )</td>
<td>( y^m_1(t) )</td>
</tr>
<tr>
<td>2</td>
<td>( y^m_0(t-2), y^m_0(t-1), y^m_0(t) )</td>
<td>( y^m_2(t+1) )</td>
</tr>
<tr>
<td>3</td>
<td>( y^m_0(t-1), y^m_0(t), y^m_0(t+1) )</td>
<td>( y^m_3(t+2) )</td>
</tr>
<tr>
<td>4</td>
<td>( y^m_0(t), y^m_0(t+1), y^m_0(t+2) )</td>
<td>( y^m_4(t+3) )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Prediction step</th>
<th>Inputs of the model</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>( y^m(t-9), y^m(t-5), y^m(t-1) )</td>
<td>( y^m_4(t+3) )</td>
</tr>
</tbody>
</table>

The DMS technique consists in directly modeling the relation between observations separated by an \( s \)-period interval and using it for the prediction. Thus, we have:

\[
y^m_s(t) = f(y^m(t-n \times s),...,y^m(t-i \times s),...,y^m(t-s)) \tag{4}
\]

where \( f \) is the prediction model, \( n \) is the number of lags used by the model and \( s \) is the prediction step ahead. Note that \( y^m_0(t) \) (i.e. when \( s = 0 \)) is the real value \( y^m(t) \). Thus, the one-step-ahead prediction formula defined in Eq.2 is valid.

In order to understand how \( y^m_s(t) \) is computed for both of these techniques, we provide an example on how \( y^m_1(t+3) \), the 4-step prediction at time \( t + 3 \), is computed from an end-of-sample \( y^m_1(t-1) \). The model uses the last 3 lags as inputs. Table I and Table II show the evaluation steps for the DMS and IMS technique respectively. We can note that the IMS technique calculates \( y^m_s(t+3) \) by applying the model 4 times (which corresponds to the number of predicted steps \( s \)). In this case, the last 3 lags at time \( t-1 \), \( t-2 \) and \( t-3 \) must be available (they are used for the step 1). However, the DMS technique calculates \( y^m_s(t+3) \) by applying the model once using the 3 lags at time \( t-9 \), \( t-5 \) and \( t-1 \). Thus, for the DMS technique, prior lags must be available.

The DMS technique is supposed to capture traffic patterns and to ignore fast variations of the traffic while IMS is more sensitive for these small variations. These two techniques were largely investigated in econometrics [25]–[27]. The previous results are very different when comparing them. In fact, the performances of these techniques vary depending on the nature of the predicted data. We note the Internet traffic data are very different from other data since due to their high variability and self-similarity [16].

- **The identification of the model**

Available data (e.g. the throughput values) are divided into two sets. The first set is called the **training data set** constitutes \( p\% \) (usually \( p = 50 \)) of the available data. It is used to identify the model parameters. The second set is the **validation data set** used to compare the prediction results with the real data in order to evaluate the performance of the predictor. We note that unlike the ARIMA model, the LMMSE model does not need a training data set. It evaluates its parameters at each step using the last \( n \) lags. The Box-Jenkins methodology is used to identify the ARIMA model parameters [29].

- **Validation of the prediction**

The performance criterion used to evaluate the accuracy of the prediction is the **Root Mean Square Error (RMSE)**:

\[
RMSE^m_s = \sqrt{\frac{\sum_{i=1}^{n} [y^m_i(t) - y^m_i(t)]^2}{n}} \tag{5}
\]

where \( y^m(t) \) is the real value of the throughput, \( y^m(t) \) the \( s \)-step prediction of \( y^m(t) \) and \( n \) is the number of the input data. Thus, the \( RMSE^m_s \) measures the error between the \( s \)-step predicted values and the real values of the aggregated throughput. In this work, the prediction model can be either the ARMA model or the LMMSE model. In what follows, we present these two models. For simplicity, we omit the superscript \( m \) i.e. we write \( y(t) \) instead of \( y^m(t) \).

- **AutoRegressive Moving Average Model**

The most well-known linear forecasting models are the **AutoRegressive (AR)**, **Moving Average (MA)** and the **AutoRegressive Moving Average (ARMA)**. A time series \( y(t) \) is an ARMA(p,q) process if it is stationary and if for every \( t \):

\[
y(t) = \phi_1 y(t-1) + ... + \phi_p y(t-p) + \theta_1 z(t-1) + ... + \theta_q z(t-q)
\]
\[ +\epsilon(t) + \theta_1\epsilon(t-1) + \ldots + \theta_q\epsilon(t-q), \]  

where the \( \phi_i \) and \( \theta_j \) are constants that are estimated from data using least squares regression [30]. The \( \epsilon(t) \) are error terms which are assumed to be independent, identically distributed sampled from a normal distribution with zero mean and finite variance \( \sigma^2 \). The parameter \( p \) is the number of lags used by the model and \( q \) is the number of error terms.

The equation can also be written in a concise form as:

\[ y(t) = \sum_{i=1}^{p} \phi_i L^i y(t) + (1 + \sum_{i=1}^{q} \theta_i L^i)\epsilon(t), \]  

where \( L \) is the backward shift operator defined as follows:

\[ L^i y(t) = y(t - i). \]  

We notice that AR and MA are special cases when \( q = 0 \) or \( p = 0 \).

The ARMA fitting procedure assumes that the data are stationary. If the time series exhibits variations that violate the stationary assumption, then there are specific approaches to make the time series stationary. The most common one is called the “differencing operation”. It is defined by:

\[ (1 - L)y(t) = y(t) - y(t - 1) \]  

It can be shown that a polynomial trend of degree \( k \) is reduced to a constant by differencing \( k \) times, that is, by applying the operator \((1 - L)^k y(t)\). An ARIMA(p,d,q) model is an ARMA(p,d,q) model that has been differenced \( d \) times. Thus, the ARIMA(p,d,q) can be given by:

\[ (1 - \sum_{i=1}^{p} \phi_i L^i)(1 - L)^d y(t) = (1 + \sum_{i=1}^{q} \theta_i L^i)\epsilon(t). \]  

C. Linear minimum mean square error

The LMMSE is a linear model [7], [15], [16], [20], [21]. The model predicts the series sample, \( y(n+1) \), in the next interval as a weighted sum of the past \( n \) average samples:

\[ y(n+1) = [a_1 \ a_2 \ \ldots \ a_n] \begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(n) \end{bmatrix} \]  

where \( a_1 a_2 \ldots a_n \) are the LMMSE coefficients. Those coefficients can be expressed as:

\[ \begin{bmatrix} a_1 \ a_2 \ \ldots \ a_n \end{bmatrix} = \begin{bmatrix} R(n) & R(n-1) & \ldots & R(1) \\ R(0) & R(1) & \ldots & R(n-1) \\ R(1) & R(0) & \ldots & R(n-2) \\ \vdots & \vdots & \ddots & \vdots \\ R(n-1) & R(n-2) & \ldots & R(0) \end{bmatrix}^{-1} \]  

where \( R(i) \) is the covariance function of the time series, it can be estimated as:

\[ R(i) = \frac{1}{n} \sum_{t=i+1}^{n} y(t)y(t-i), \quad 0 \leq i \leq n-1. \]  

D. Trivial model

Trivial model predicts the series sample, \( y(t) \), in the next interval as \( y(t-1) \).

IV. ANALYSIS OF THE DATA

This section presents the used data sets and the preprocessing performed before prediction experiments.

The first set of data is the “Auckland-VIII data set”\(^1\). It is a two-week GPS-synchronized IP header trace captured with an Endace DAG3.5E tap Ethernet network measurement card in a link of 100 Mbps at the University of Auckland Internet.

The second set of data is the “CESCA-I data set”. It is a three-hour GPS-synchronized IP header trace captured in a 1Gbps link with an Endace DAG4.2GE dual Gigabit Ethernet network measurement card in February 2004 at the Anella Cientifica (Scientific Ring), the Catalan R&D network.

In this paper, we present the results found using 60 minutes of data from both traces measured at 10 am in December 2, 2003. We used 50% of the data (30 min) as the training data set and 50% (30 min) as the validation data set.

We analyzed the traces with the libtrace tools\(^2\). Table III shows a comparison between Auckland and CESCA traces. It shows that for the same duration (1 hour), the 100-Mbps link (Auckland) has much less load than the 1-Gbps link (CESCA) in terms of the data size and the throughput mean.

**TABLE III**

**TRAFFIC CHARACTERISTICS: AUCKLAND-VIII VS. CESCA-I.**

<table>
<thead>
<tr>
<th>Statistics</th>
<th>Auckland-VIII (Mbps)</th>
<th>CESCA-I (Mbps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Link Capacity</td>
<td>100</td>
<td>1</td>
</tr>
<tr>
<td>Duration</td>
<td>1 hour</td>
<td>1 hour</td>
</tr>
<tr>
<td>Trace Size</td>
<td>60 Mbyte</td>
<td>20.5 Gbyte</td>
</tr>
<tr>
<td>Throughput Mean</td>
<td>2.83</td>
<td>487.16</td>
</tr>
<tr>
<td>(Mbps)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Throughput Variance</td>
<td>1.280</td>
<td>797.781</td>
</tr>
<tr>
<td>(Mbps)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Throughput Mean</td>
<td>972.73pps</td>
<td>100 036.14 pps</td>
</tr>
<tr>
<td>(packets per second)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average Connection</td>
<td>8.23sec</td>
<td>8.14sec</td>
</tr>
<tr>
<td>Duration</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 1(a) illustrates the packet size distribution for the Auckland traffic. The smallest packets, 40 bytes in length, represent 27% of packets number. They are mainly TCP packets with ACK, SYN, FIN, or RST flags. They are many 1, 500-byte packets (17.67%) which result from the maximum packet size when using Ethernet. They are also many 1, 420-byte packets (2.10%). The large presence of 576-byte packets (5.14%) reflects TCP implementations without "path MTU discovery", which use packets of 536 bytes (plus 40-byte header) as the default Maximum Segment Size [31].

Fig. 1(b) shows the packet size distribution considering the generated traffic. The generated traffic is the amount of data

\(^1\)Data are available from the National Science Foundation (NSF) and the NLANR Measurement and Network Analysis Group (http://pmn.nlanr.net/Special/).

in MegaBit generated by a particular size. The figure shows that 1,500-byte packets constitutes 57% of the traffic but only 17.67% of the total number of packets. Besides, although more than 27% of the packets are from small packets, they constitute less than 5% of the generated traffic. This observation concurs with the findings of Shao et Al. [19] that the traffic pattern is bimodal: most traffic is carried by a small number of packets and most packets carry smaller number of bytes. The same observation is done for the CESCA traffic (Fig. 2).

V. EMPIRICAL ANALYSIS OF THE PREDICTION MODELS

A. Number of Lags

In order to choose the number of lags that should be used as inputs the prediction model, the correlation coefficient is used as a metric. In general statistical usage, correlation refers to the departure of two variables from independence. In this broad sense there are several coefficients, measuring the degree of correlation, adapted to the nature of data. A number of different coefficients are used for different situations.

The best known correlation coefficient is the Pearson product-moment correlation coefficient, which is obtained by dividing the covariance of the two variables by the product of their standard deviations [32]. The correlation $\rho_{X,Y}$ between two random variables $X$ and $Y$ with expected values (mean) $\mu_X$ and $\mu_Y$ and standard deviations $\sigma_X$ and $\sigma_Y$ is defined as:

$$
\rho_{X,Y} = \frac{\text{cov}(X,Y)}{\sigma_X \sigma_Y} = \frac{(X - \mu_X)(Y - \mu_Y)}{\sigma_X \sigma_Y}.
$$

We note that $\mu_X = E(X)$, $\sigma_X^2 = E(X^2) - E^2(X)$ and likewise for $Y$. Since $\mu_X = E(X)$, $\sigma_X^2 = E(X^2) - E^2(X)$ and likewise for $Y$, we may also write

$$
\rho_{X,Y} = \frac{E(XY) - E(X)E(Y)}{\sqrt{E(X^2) - E^2(X)}\sqrt{E(Y^2) - E^2(Y)}}
$$

Fig. 1. Packet Size Distribution (a) considering the number of packets (b) considering the generated traffic (the Auckland traffic).

Fig. 2. Packet Size Distribution (a) considering the number of packets (b) considering the generated traffic (the CESCA traffic).

Fig. 3. Correlation coefficient for different granularities.

(a) Auckland data.

(b) Cesca data.
In our case, we calculate the correlation coefficient for variable \(y(t-i)\) (e.g. the lag \(i\)) and the variable \(y(t)\). Fig. 3 shows the correlation coefficient calculated for 25 lags for various granularities. The correlation coefficient decreases for higher lags. That is pertinence of lags \(y(t-i)\) to predict \(y(t)\) is decreasing when \(i\) increases. Thus, \(y(t-1)\) and \(y(t-2)\) are the most correlated to \(y(t)\). This suggests that they are the most relevant to predict \(y(t)\). Fig. 3 also shows that the correlation between \(y(t-i)\) and \(y(t)\) increases slowly as the granularity increases. This suggests a better prediction performance as the granularity increases. These observations are confirmed by the results of the one-step and the multi-step predictions using different lags as it will be discussed in subsection V-B and V-C. We also note that the correlation coefficient of the 100-Mbps link (Auckland) decreases towards zero as the number of lags increases; Whereas it decreases slowly but still high for the 1-Gbps link (Cesca). Thus, the CESCA data is much more correlated than the Auckland data.

B. Traffic Granularity

In this paragraph, we discuss the effect of the traffic granularity on the prediction performance. The traffic granularity is the interval of time separating two consecutive measures of the traffic. Ideally, the traffic granularity should be chosen based on the application of the prediction. We performed predictions with granularities varying from 10 ms to 20 sec.

Fig. 4 and Fig. 5 depicts the obtained RMSE for different granularities. It show that using more than one or two lags as inputs for the models does not improve prediction performance; but on the contrary, it increases the prediction error. We think that it is due to the high variability and the bursty nature the traffic. For instance, when predicting \(y_1(t+1)\), the last two lags \(y_0(t)\) and \(y_0(t-1)\) have values close to the real \(y_0(t+1)\). Thus, the linear models (like Eq.(6) and Eq.(10)) provide good prediction \(y_1(t+1)\) of \(y_0(t+1)\). However, when using more lags \(y_0(t-10)\), \(y_0(t-1)\), linear models provide high error because these inputs hover around different averages and have high variance. The linear combination of the inputs will be far from the real value of \(y_0(t+1)\).

We also note that when using a small number of lags, the prediction error (RMSE) decreases as the granularity increases. Thus, the predictability is improved for high granularities.

Fig. 6 shows the standard deviation of the data (the square root of the variance) and the 1-step prediction error for different granularities. We notice that when the granularity increases, the standard deviation and the prediction error are decreasing. We can infer that the traffic becomes less predictable when the variance is high. We also note that for the 1-step prediction, the trivial model provides the same performance of the ARIMA or the LMMSE model especially for the 100-Mbps link. For more than 1-step prediction, the
ARIMA and the LMMSE models provide better performance than the trivial model (Fig. 7).

C. MultiStep Prediction

In this paragraph, our interest lies in assessing the performance of the models for multistep prediction. We compare then IMS and DMS techniques.

Figure 8 shows the IMS multistep prediction error using different number of lags as inputs for the ARIMA model. We only consider the granularity 1 sec. The first observation is that using a high number of lags decreases the prediction performance in terms of error. Second, multi-step prediction using ARIMA model could be done using only a small number of lags for the model (1 or 2 lags). We note that, compared to ARIMA model, the LMMSE model provides high prediction error that is why we omitted the figures representing its results.

We investigate now the effect of the granularity on multistep prediction performance for the model ARIMA(2,1,2). Fig. 9 shows the IMS multistep prediction error for different granularities. As expected, the prediction error increases when the prediction step is high for any granularity. The interesting feature is that the graph shows concavity around the granu-
This optimal granularity allows better multi-step prediction. Thus, increasing the granularity does not necessarily improve the performance as shown in Fig. 9(a).

Figure 10 compares the IMS and the DMS techniques for different granularities. We considered the ARMA model for both cases. The figure shows that 4-step prediction error found with the IMS technique is always lower than the one found with the DMS technique for all granularities. The difference between the performances is more important when the data variance is more important. Thus, the prediction error for the DMS technique is more important for the CESCA traffic since it is more variable. This result is explained by the fact that the DMS technique does not use successive lags but lags that are spaced in time (Table II). This implies high variability between lags. Thus, it is difficult to the model to be accurate. We can conclude that the IMS technique is more suitable for multistep prediction of the Internet traffic due to its high variability over time.

D. Prediction based on packet Size

The packet size distribution shows that there are special sizes that are more present than the others in the traffic. In this paragraph, we perform 1-step ahead prediction based on the packets size. That is, we predict the throughput generated by each packet size (the rate in Mbps). We consider only the granularity 1 sec and only 2 lags for the ARIMA model since 2 lags are quite sufficient for prediction (section V-A). We note \( y_0(t, \delta) \), the throughput in Mbps for the packets of size \( \delta \). Thus, the throughput \( y_0(t) \) can be written as:

\[
y_0(t) = \sum_{\delta = 20}^{1500} y_0(t, \delta)
\]

Then \( y_1(t, \delta) \) is the 1-step prediction of \( y_0(t, \delta) \).

Fig. 11 shows the prediction error as well as the standard deviation for the throughput of each packet size. This shows that, for both traces, the prediction error comes from the highest packet size namely 1,500 and 1,420 bytes.

Table IV shows the prediction error of the throughput \( y_0(t) \) using Eq.(2) or as the sum of the \( y_1(t, \delta) \), the throughput for each packet size. It shows that almost the same prediction performance is obtained. This result is important as it reflects the ability of predicting the throughput of each packet size apart. For instance, the result could be used for active queue management to reject packets based on the prediction of the throughput of each size.

Fig. 11. Standard deviation and prediction error (RMSE) for different packet size (granularity 1 sec).

<table>
<thead>
<tr>
<th>Packet Size (bytes)</th>
<th>ARIMA(2,1,2) Standard deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(a) Auckland-VIII data set.</td>
</tr>
<tr>
<td></td>
<td>(b) CESCA-I data set.</td>
</tr>
</tbody>
</table>

Table IV: Prediction using different packet sizes vs. direct prediction

<table>
<thead>
<tr>
<th></th>
<th>Auckland</th>
<th>Cesca</th>
</tr>
</thead>
<tbody>
<tr>
<td>( y_3(t) )</td>
<td>2.58</td>
<td>12.78</td>
</tr>
<tr>
<td>( \sum_{\delta=20}^{1500} y_1(t, \delta) )</td>
<td>2.57</td>
<td>12.26</td>
</tr>
</tbody>
</table>
VI. CONCLUSIONS AND FUTURE WORK

An analysis of Multi-step prediction performance of the ARIMA and the LMMSE models has been performed based on two sets of real Internet measurements. We find that one or two lags are sufficient to perform quite accurate prediction regardless of the used granularity. Unexpectedly, using more than two lags as inputs for the model increases the prediction error. We found that the ARIMA model performs a performance close of that of the ARIMA or the LMMSE models in the case of the one-step ahead prediction. Thus, there is no need to complicated models. For multi-step prediction, the ARIMA outperforms the LMMSE and the trivial models. For multi-step prediction, increasing the granularity does not improve the performance of the prediction. In fact, there is a granularity at which the multi-step prediction is more accurate. For the 1-step prediction case, the performance is improved as the granularity is increased. We also compared IMS and DMS multi-step techniques. We find that the IMS technique provides more accurate traffic prediction than the DMS technique because of the high variance of the data. It is possible to decompose traffic into several components based on packet’s size. Analysis shows that traffic behavior depends on large packets especially 1,500-byte packets. Their throughput is less predictable than the other packets. However, it is clear the effect of the traffic variance on its predictability. We found that high variance has led to a high prediction error.

Even though the Internet traffic varies in time and space, we are optimistic that the results are valid for any network traces since they have common characteristics (self-similarity, high correlation and variance, and even packet size distribution).

The paper analyzes only some aspects of Internet traffic prediction. Many issues require further study, including the effects of some other parameters such as the number of flows, packet loss, and cross traffic nature on traffic predictability.

REFERENCES


[31] “RFC 879 - TCP maximum segment size and related topics”.

A Visualization Tool for Exploring Multi-scale Network Traffic Anomalies

Romain Fontugne\textsuperscript{1}, Toshio Hirotsu\textsuperscript{2}, Kensuke Fukuda\textsuperscript{3}
\textsuperscript{1}The Graduate University for Advanced Studies, Tokyo, Japan
\textsuperscript{2}Hosei University, Tokyo, Japan
\textsuperscript{3}National Institute of Informatics / PRESTO, JST, Tokyo, Japan

Abstract—Since anomaly detection in Internet traffic is a crucial and unmet challenge, many anomaly detectors for backbone traffic have recently been proposed. However, evaluating anomaly detectors is a complicated task due to the lack of ground truth data. Our goal is to provide a good level of support for rapidly understanding traffic behaviors and assisting researchers in evaluating the effectiveness of anomaly detectors. This article presents an interactive tool that takes advantage of several graphical representations highlighting the different aspects of network traffic and anomalies. The proposed tool allows for exploration of network traffic at any temporal and/or spatial (address and port) scales. In addition, an accurate description of any sub-traffic is available in the form of textual packet information, enabling complete understanding of the monitored traffic. We exhibit the effectiveness of the proposed tool by analyzing darknet traffic, backbone traffic, and anomalies reported by an anomaly detector. We illustrate a manual validation of the anomalous traffic reported by anomaly detectors, and inspect a recent and sophisticated threat: the Conficker worm. We also state several typical patterns that stand for different kinds of anomalies.

I. INTRODUCTION

The Internet has become a common medium for communication and information exchange providing many attractive services for ordinary users. A victim of its own success, Internet traffic is still growing at a fast rate and contains an increasing amount of anomalies such as misconfigurations, failures, and attacks. These improper uses of network resources consume bandwidth and adversely affect the performances of networks. Thus, these anomalies penalize legitimate applications from using an optimal amount of network resources. Since the core of the Internet is particularly deteriorated by anomalous traffic, quick and accurate detection of anomalies in the backbone traffic has been a hot topic (e.g., [1], [2], [3], [4]). However, due to the lack of ground truth data for backbone traffic, evaluating an anomaly detector is quite challenging and tricky [5]. Therefore, researchers must validate their results from their anomaly detectors by manually inspecting the dump files or flow records. This is a baffling problem as it is laborious to identify a few thousand harmful packets from millions of innocuous ones.

Nevertheless, visualizing network traffic is a valuable tool for in investigating dump files. The main advantage of graphical representations is to highlight the significant features of the traffic, thus the main properties of the traffic are understood at a mere glance. Moreover, several degrees of information are retrieved by monitoring the various representations that depict different aggregations of the traffic. For example, a time series is useful for analyzing the time evolution of a single feature for a huge amount of flows. Whereas, a graphlet [6] depicts several features of only a few flows.

In this article, we propose a tool to visualize, explore, and understand network traffic at any temporal and spatial (address and port) scale. Our main contribution is to provide a tool that assists researchers, or network operators, in understanding and validating alarms reported by their anomaly detectors. The proposed tool provides six basic features to help researchers inspect network traffic and evaluate anomaly detectors:

- Network traffic is displayed at different resolutions, and the user is able to zoom in/out along the time axis or address/port space.
- The tool provides different types of scatter plots (corresponding to IP addresses, or port numbers) and time series (e.g., throughput and average packet size). Since these graphical representations are intuitive views, the tool simultaneously displays two views and provide an exhaustive description of the traffic.
- Understanding backbone traffic involves inspecting various sub-traffics, and therefore, the tool allows to easily move along the network traffic in time and space (i.e. address and port number space).
- The tool retrieves all the details concerning the monitored traffic in the form of accurate graphlet and textual data.
- Anomalies identified by anomaly detectors are displayed by this tool, and thus, researchers and network operators are able to easily validate the veracity of the detected anomalies.
- The current implementation runs on different platforms on a daily basis, it uses no intermediate database, and it directly reads dump files (pcap form [7]).

We evaluated the tool on several kinds of traffic: darknet traffic reveals shapes highlighting anomalous traffic, and similar patterns are also observed in the backbone traffic. Furthermore, we demonstrate the help provided by the tool in identifying recent and sophisticated attacks such as the Conficker worm. We also conduct a manual inspection of anomalous traffic reported by anomaly detector, and list several typical patterns highlighting anomaly detector (in accordance with those reported in [4]).
II. RELATED WORK

Various visualization tools assist researchers and network operators in monitoring network traffic. For example, Fischer et al. [8] and Goodall et al. [9] presented two interesting tools focusing on anomaly detection. The former [8] monitors traffic related to local hosts based on a TreeMap visualization. It is used to check alarms reported by intrusion detection system (IDS), and to identify large-scale attacks aiming at local hosts. The latter, Time-based Network traffic Visualizer (TNV) [9], highlights the connections between the hosts sorted within a matrix. The traffic between local and remote hosts is clearly displayed, and all the information about the packets is accessible. However, these two tools only display a limited number of hosts (e.g., about 100 hosts for TNV on a 1280x1024 display), and their home-centric view is not suitable for backbone traffic where the terms local and remote hosts are meaningless.

InetVis [10] is a visualization tool used to monitor the network traffic in three-dimensional scatter plots. Traffic is mapped into a cube [11] highlighting the specific patterns for particular anomalies. Although InetVis is adequate enough for monitoring small or extracted traffic (e.g., using IDS [12]), figures generated from heavy traffic (e.g. backbone traffic) are difficult to read and omit a lot of information. Moreover, textual information concerning plotted points cannot be obtained using this tool, whereas, information like port numbers, IP addresses, or TCP flags are usually required to validate anomalies. NVisionIP [13] is another visualization tool that cannot retrieve packet headers — essential to conduct thorough inspections of network traffic — although it is able to display traffic from large networks at several levels of aggregation, and provides detailed statistics on any hosts.

Similar to our work, IDGraphs [14] only displays two-dimensional views based on time. IDGraphs maps an original TCP-flag-based feature (SYN-SYN/ACK values of complete flows) on the vertical axis and emphasizes several patterns for different kind of attacks. However, due to routing policies, the backbone traffic is usually asymmetric and contains numerous incomplete flows, and therefore, the proposed feature based on the TCP flag is irrelevant for analyzing backbone traffic.

Our main contribution is to provide a visualization tool able to display high-volume-traffic from backbone link. Furthermore, global and detailed views of the traffic are available and no assumption on the traffic is required (e.g. complete flows or LAN traffic).

III. DESIGN AND FEATURES

A. Goals

Our main goal is to provide an interactive tool, to intuitively understand backbone traffic at different temporal or spatial resolutions, and to validate alarms reported by anomaly detectors. Manually validating results obtained from anomaly detectors is a challenging task because of the multi-dimensionality of network traffic and the large amount of data. Thus, we designed the proposed tool to include the following requirements: the tool has to focus on the significant traffic features to show a network traffic behavior and highlight anomalies in a way that is intelligible to users. It should enable the identification of diverse anomalies by exploring traffic at different scales and in various graphical representations, and permits a particular subset of the whole traffic to be analyzed by filtering the entire set of traffic. A precise understanding of the monitored traffic has to be gained by displaying the original header information and accurate graphs from selected plots. Since this tool is interactive, it has to display figures sufficiently fast, and provide them on different platforms. Script languages or interpreted languages have to be avoided for performance reasons. As the tool has to be quickly operational on several files, it needs to read data directly from the dump files and should not use an intermediate database.

B. Graphical representations

An asset of the proposed tool is its ability to display a large amount of data and highlight unusual behaviors in two-dimensional views that are easily readable. Obviously, three-dimensional views would provide additional information compared to those that are two-dimensional (hereafter respectively called 3-D and 2-D views). However, to observe such 3-D views we have to project them down onto a 2-D visual aid (e.g., a screen or paper). Two main issues are raised by this dimensionality reduction, namely disorientation and occlusion [15]. Disorientation means that the position of the plotted data is not clear and the values corresponding to the plots are difficult to retrieve. Occlusion occurs when plots hide one another, so information is omitted from view. These two problems are well-known in the field of computer vision, and a common solution is to display several 2-D projections instead of a single 3-D view.

Figure 1 shows an example of a 3-D scatter plot representing network traffic. The three dimensions correspond to the timestamp, source port, and destination port. The main advantage of this representation is to present two traffic features and the time in a single view. Nevertheless, the exact position of each point is difficult to determine and confusing. Also, we need to rotate the cube to verify that plots are not hidden in this
particular view. The occlusion issue is even more important when more data are displayed. However, by projecting data onto the faces of a cube surrounding traffic, we obtain an accurate 2-D view of the traffic. For example, the two scatter plots on the right-hand side of Fig. 1 represent the same traffic; the top one is drawn in the function of the source port and time, while the one at the bottom visualizes the traffic with regard to the destination port and time. These sub-figures are more readily understood than the 3-D representation and allow us to accurately identify the ports numbers corresponding to the plots.

The same type of 2-D scatter plot monitors traffic in the proposed tool, displaying understandable views of the traffic even though we have taken five dimensions into consideration (source port, destination port, source address, destination address, and time). In particular, the network traffic is represented in a five-dimensional space and projected onto several 2-D planes, where the horizontal axis always represents the time, but the vertical axis represents the different traffic features. The following constitutes a list of all the possible ways to represent network traffic using the tool; the first four scatter plots use a color convention where a plotted point is green when it stands for a few packets and becomes progressively redder as the number of packets it represents increases. On the other hand, the next three plots are a time series with the vertical axis representing the different traffic features.

The same type of 2-D scatter plot monitors traffic in the proposed tool, displaying understandable views of the traffic even though we have taken five dimensions into consideration (source port, destination port, source address, destination address, and time). In particular, the network traffic is represented in a five-dimensional space and projected onto several 2-D planes, where the horizontal axis always represents the time, but the vertical axis represents the different traffic features. The following constitutes a list of all the possible ways to represent network traffic using the tool; the first four scatter plots use a color convention where a plotted point is green when it stands for a few packets and becomes progressively redder as the number of packets it represents increases. On the other hand, the next three plots are a time series with the vertical axis representing the different traffic features.

1) **Destination IP address space:** This representation exposes anomalies through their targets. It highlights anomalies that aim at many hosts, or anomalies generating a lot of traffic to a single host/sub-network. The resulting scatter plots have vertical or oblique “lines” (consecutively aligned dots) for anomalies, such as remote exploit attacks, and horizontal “lines” for the targets of DoS attacks, or heavy hitters.

2) **Destination port number:** This representation emphasizes services targeted in the observed traffic. Obviously, busy services and port scans are highlighted and respectively occur as horizontal and oblique “lines”.

3) **Source IP address space:** This representation highlights the origins of the traffic. Anomalies generating heavy traffic from a single host appear as a horizontal line in the resulting scatter plots. Also, this representation emphasizes various traffics initialized at the same time as DDoS, botnet, or flash crowd.

4) **Source port number:** This representation reveals the port used by the hosts to communicate. Anomalies based on flooding create as many connections as possible using an increasing source port number. This is translated here as vertical or oblique “lines”. This graphical representation is helpful for exposing various kinds of DoSs and remote exploit attacks.

5) **Number of packets:** Here, the displayed figures are the time series of the number of packets transmitted for each protocol. A red time series is derived for TCP packets, a blue one for UDP, a green one for ICMP, and a black one for other protocols. This representation highlights the misuse of a protocol. For example, a flood generates a considerable number of packets using a particular protocol, easily identifiable as a significant variation in the time series.

6) **Number of bytes:** Several anomalies cause abnormal variations in the number of bytes. These processes that consume bandwidth are highlighted in this representation as significant variations in the time series.

7) **Average packet size:** As described by Bardford et al. [1], the average packet size can be taken into consideration to detect anomalies. This reveals the abuse of a particular application, as applications usually use the same packet size for all communications they carry out. This representation is a time series of the average packet size, where anomalies are emphasized by abnormal variations.

**C. Tool overview**

Figure 2 is an overview of our tool, which is composed of three panels, a small one (W0) with a menu bar and an overview of the traffic, and two larger ones (W1 and W2) displaying the traffic in detail. Since our tool displays only 2-D graphical representation based on a single traffic feature, the two detailed panels (W1 and W2 in Fig. 2) allow the monitoring of two traffic features simultaneously. Users choose which representation has to be displayed in each panel (available representations are listed in Section III-B). Thus, our tool avoids the confusion caused by irrelevant information and focuses on the anomalies as they are generally revealed through unusual uses of one or two traffic features [2]. For example, a network scan can easily be identified by analyzing only the destination address and destination port.

Sections III-D and III-E explain several operations for navigating in W1. Depending on these operations, W2 is automatically updated, providing more information about the traffic displayed in W1 as W2 displays only the packets shown in the W1 view. For example, W1 in Fig. 2 displays a scatter plot of the destination addresses, whereas W2 displays a scatter plot of the source ports. When W1 is zoomed to select a particular sub-network, W2 only presents packets for this sub-network. In W0, the blue rectangle (labeled “Navigation” in Fig. 2) helps us to figure out where the detailed view is located in the entire traffic. W0 also provides a packet header that corresponds to certain points selected by the user.

**D. Multi-scale**

Anomalies appear at different temporal and spatial scales. Namely, they can last for short or long periods (from an order of seconds to several hours), and they can aim at a single or multiple targets, on one or several ports. The proposed tool allows to zoom in/out independently on each axis. The length of time and feature space (e.g. address space) can be adjusted at any time. This is easily achieved with the mouse wheel, or corresponding buttons. Thus, when long and short-term anomalies are observed, their time duration and their impact in the feature space can easily be estimated.
E. Easy navigation

Inspecting network traffic and thoroughly investigating anomalous traffic requires movement along the traffic trace and a focus on a particular region. The proposed tool lets users conveniently navigate through the analyzed traffic. Only a click on a particular point is required to center the view on that zone.

F. Packet information

Characterizing anomalies is a complicated task, as some of them are only identifiable by inspecting the flags of the packet header. A combination of graphical and textual information is essential for identifying anomalies. Our tool helps users in their investigations by providing useful information about all the plotted pixels. A right click on a point in a figure brings up a zoomed view of the clicked zone, and a particular point can be selected to check the corresponding packets headers, and thus we can learn more about the displayed traffic. The tool also represents the selected data as a graphlet that is similar to those presented in BLINC [6]. These graphlets (or parallel coordinates [16]) allow us to simultaneously visualize more than two dimensions, and intuitively highlight communication patterns. The tool takes advantage of this graphical representation to display only small data sets pointed at by the user (graphlets representing large data sets are too confusing).

G. Input

The tool has to quickly display figures from several input files. Although it would be easier to access data, copying files into an intermediate database is too costly for analyzing daily backbone traffic. Instead of using a database, the tool reads directly from the dump files, like those produced by tcpdump. Also, the tool is able to directly read from compressed files (commonly used to save disk space). Moreover, several files can be given as inputs, and hence, the resulting figures are drawn as all the corresponding files are merged.

H. Anomaly description

Reports from anomaly detectors are passed on to the tool in the form of admd files\(^1\), which is a XML schema allowing the annotation of traffic in an easy and flexible way. Thus, anomalies reported by anomaly detectors are quickly identified and inspected as they are displayed in black in all the scatter plots.

I. Portability

Our tool is designed for users utilizing different platforms. We avoided script and interpreted languages for performance purposes, and implemented this application in C++ using only portable libraries to make it available to most users (e.g. views are displayed with the Clmg library [17]). Thus, the tool can currently be compiled and executed on different platforms: Unix (Linux and BSD), MacOS, and Windows.

J. Option

The tool is customizable through the command line interface to better fit the needs of the users. One important option from among the many options available permits to filter displayed traffic, thus, the tool monitors only certain sub-traffic from the entire traffic trace. Filters have the same syntax as pcap’s filters (the same as those used in tcpdump) and are based on any field of the packet header. They allow specific sub-traffic to be accurately selected. For example, this option helps investigations into anomalous traffic by displaying only traffic from a suspicious host on certain ports, or by only selecting SYN packets to highlight the probing processes and SYN flood.

K. Snapshot

Saving pictures of traces previously observed is essential for visually comparing or illustrating traffic behaviors. Users can save a snapshot of a particular figure at any time. The snapshots are in PNG format and the size can be specified by the user. The tool can also be used to generate a batch of visualizations from a set of files with the command line interface. For example, visualizations of daily figures from a year of traces can be generated and stored using only one command line\(^2\).

IV. RESULTS

A. Performance

The comfort of navigation and inspection of traffic with our tool is strongly related to its performance and reactivity

\[ \text{Table I. Gain in performance due to mechanism for seeking in PCAP files} \]

<table>
<thead>
<tr>
<th></th>
<th>User CPU time (clock ticks)</th>
<th>System CPU time (clock ticks)</th>
<th>Time elapsed (minutes:secs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>With 'seek structure'</td>
<td>6.00</td>
<td>0.64</td>
<td>00:23.28</td>
</tr>
<tr>
<td>Without 'seek structure'</td>
<td>10.25</td>
<td>1.43</td>
<td>00:58.42</td>
</tr>
</tbody>
</table>

\(^1\)Meta-data format and associated tools for the analysis of pcap data: [http://admd.sourceforge.net](http://admd.sourceforge.net)

\(^2\)An example resulting from this feature is the website MAWIViz illustrating all traffic traces of the MAWI archive [18]: [http://www.fukudalab.org/~romain/MAWIViz/](http://www.fukudalab.org/~romain/MAWIViz/)

© 2011 ACADEMY PUBLISHER
to one’s actions. Since the tool directly reads pcap files, some performance issues are addressed. The main problem is that libpcap does not offer the possibility to directly access a subset of packets corresponding to a given time interval. In practice, the whole traffic trace has to be scanned consuming substantial resources for large traffic traces. Therefore, we consider a dump file to be several parts of the same duration where the first packets of these time slices are called “key packets”.

Our implementation consists of a data structure that retains information on the “key packets”, such as their timestamps and their offsets in the trace file. This data structure helps us to directly access a “key packet” regarding its timestamp. Thus, “key packets” are used as indexes in order to quickly go through the traffic trace. For example, to read a packet at a particular time, \( t_0 \), the data structure helps us to jump to the “key packet” preceding \( t_0 \), thereby avoiding having to read numerous unwanted packets prior to this “key packet”.

Table I lists the gains in performance we obtained with this improvement. The numbers in this table represent the average results from five executions of the same scenario. The scenario consisted of five consecutive zooms in the time space on an uncompressed trace of about 800 MB. The measurements were done on a Linux system with the \texttt{time} command, using a computer with 2 GB of RAM and an Intel Core 2 Duo CPU operating at 2.6 GHz. This improvement makes for a comfortable multi-scale navigation through large traffic traces.

B. Darknet data

Figure 3 shows an example of the scatter plots generated from darknet traces taken from a /18 sub-network. As described by Pang et al. [19], darknet (or background radiation) is a type of nonproductive traffic sent to unused address spaces. Darknet data are usually analyzed to characterize anomalies and useful for demonstrating the efficiency of our tool. The vertical axis in the first panel of Fig. 3 stands for the destination addresses, whereas this axis represents the source port numbers in the second panel.

The vertical “lines” in the first panel represent the exploited attacks or any processes using network scans (e.g., (e)). The horizontal “lines” stand for the hosts or sub-networks under heavy attack. They could be the targets of any flood attacks or misconfigurations (e.g., (d) and (f) in the figure).

Other kinds of anomalies are observed in the second panel, and more information about those found in the previous scatter plot are available. Here the vertical “lines” or oblique “lines” represent any procedure using an increasing number of source ports. This is the case in most operating systems when a process opens as many connections as possible. The horizontal “lines” in this panel indicates the constant and heavy traffic from a single port, emphasizing floods, misconfigurations, or heavy-hitters. We can see two sets of consecutive vertical “lines” ((a) and (b) in Fig. 3) appearing at the same time as sudden heavy noise in the first panel. These two behaviors are interpreted as a process trying to access many of the computers of a sub-network within a short time duration (e.g. exploit or worm) as possible. Checking the headers information revealed that all these packets are directed to port 445. Windows has vulnerabilities in its protocols using this port and several worms have spread through these vulnerabilities. The vertical “line” (e) depicts the same behavior, but within a shorter time frame. Indeed, the packet header information emphasizes an exploit on ssh. We also analyzed the oblique curves (see (c) an (d) in Fig. 3) and detected attacks aimed at services sensitive to attacks. These attacks are not linear because of the variations in time processing or network delays (due to another activity (d) has some jumps in its source port numbers). Checking packet header data reveals that the ports concerned are 80 for (c) and 161 for (d). These services are the targets of well-known attacks driving DoS or buffer overflows. (d) aims at a small sub-network (see (d) in the first panel), whereas (c) is aimed at a single target easily identifiable by zooming in on (f).

C. Network traffic from trans-Pacific link

As an example of anomalies surrounded by legitimate traffic, we analyzed a traffic trace from the MAWI archive [18], which is a set of traffic traces that has been collected by the WIDE Project from 1999. This archive provides large-scale traces taken from trans-Pacific links. The traffic traces are in pcap form without any payload data with both addresses anonymized. Also, the time duration of each trace is fifteen
Figure 4 depicts views from ten consecutive files of the MAWI database. The total size of these ten files is about 7.6 GB, for a time of 2.5 h and more than 22 million packets. The vertical axis in the first panel stands for source ports. We can easily see that traffic is heavier than in the example presented in previous section. However, we can still distinguish several red "lines" highlighting some intensive uses of network resources. In the following, we focus on the right part of this figure. Consequently, the next scatter plot results from zooming in on the time axis.

The second panel has also been drawn in regard to source ports. Header information helps us to understand plotted pixels; the two oblique "lines" crossing the figure (see (a) in Fig. 4) represent a SYN flood. This is an attack from a single host to several targets, the attacker floods targets on port 443 (usually used for HTTP over SSL). This method is well known and results in buffer overflows in the Private Communications Transport (PCT) protocol implementation in the Microsoft SSL library. The other oblique "lines" represent the same kinds of attacks against other services and from different hosts. In particular, (b) stands for a DDoS attack against a few HTTP servers. The horizontal red "lines" are anomalies consuming bandwidth as in DoS attacks, misconfiguration or heavy-hitters from peer-to-peer networks.

The last panel in Fig. 4 shows the same traffic but in regard to the destination ports. Similar "lines" to those found in the previous panel (b) appear. They stand for the server’s reactions to the DDoS attacks previously observed. Also, two kinds of "lines" repeated several times (see (c) and (d)) are highlighted. Both of these are DoS attacks of ACK packets from two distinct hosts against different targets.
D. Manual inspection

1) Inspecting a specific anomaly: The tool helps in inspecting a particular sub-traffic by filtering the entire data before plotting it. The given filters are similar to those in tcpdump allowing for a powerful data extraction. Using filters, the tool is also useful for creating the visualizations of reported anomalies providing additional information in anomaly detector reports. Moreover, filters improve the global performance of the tool as less traffic is displayed.

For example, an anomaly detector [4] reported anomalous traffic on port 515. As this is not a typical target for attacks, we investigated the traffic related to this port. We monitored only the traffic for port 515 (Fig. 5) with the filtering option of our tool. The upper panel of Fig. 5 highlights the destination addresses of the traffic, and depicts two different traffic behaviors; the left-hand side of the scatter plot shows many short communications dispersed over numerous destination hosts, whereas, the right-hand side of the scatter plot displays longer communications concentrated on a few hosts. This can be interpreted as an attacker probing sub-networks to identify hosts with specific security holes, and a few connections are established to compromise detected victims. The bottom part of Fig. 5 represents the average packet size corresponding to the traffic displayed in the scatter plot. This time series also exhibits two different phases; it clearly indicates that the size of the packets during the first half of the analyzed traffic is abnormally constant while the second half is more typically fluctuating. The average size of packets in the first phase is particularly small due to the lack of packet payload used during the probing process. However, the following communications have packet payloads that considerably increase the average packet size.

The traffic behavior can intuitively be understood from Fig. 5, but actual information is still needed to confirm this. The tool supplies header information that corresponds to the displayed plots. Textual header information and a corresponding graphlet are obtained by pointing to a particular plot in the graph.

We retrieved information from several of the plots in Fig. 5 to clearly comprehend the displayed traffic. Figure 6 shows a graphlet corresponding to the header information from various plots selected from the first half of the analyzed traffic. The textual data reveals that all packets had a SYN flag set, and confirms that the plotted traffic corresponds to a probing process.

2) Inspecting outputs from anomaly detectors: The proposed tool provides valuable assistance to understand and evaluate anomaly detection methods by displaying their results at any temporal and spatial scales in various views. Indeed, by passing the anomaly detector results and original traffic to the tool, it monitors the reported anomalies and helps in rapidly validating them. Thus, researchers designing anomaly detectors are able to validate at a glance the traffic reported by their anomaly detectors and thoroughly inspect anomalies by retrieving anomalous packet header information.

Two examples of anomalies reported by two distinct
anomaly detectors are depicted in Figure 7, where the anomalous traffics are displayed in black. The two anomaly detectors analyzed a MAWI traffic trace in which the first quarter of the traffic is strongly altered by the spreading of the Sasser worm (see the main peak in Fig. 7(c)). The upper scatter plot (see Fig. 7(a)) depicts 337 anomalies reported by an anomaly detector based on image processing [4]. This view exhibits the inability of this anomaly detector (with the specified parameter set) to detect all Sasser activities during the main outbreak of the worm. This case emphasizes the valuable support provided by the tool as this fact could not be deduced by only inspecting the textual results outputted by the anomaly detector.

The middle scatter plot depicts 332 anomalies obtained with another anomaly detector based on multi-scale gamma modeling [3]. A quick visual comparison of the two views (Fig. 7(a) and Fig. 7(b)) indicates that these two anomaly detectors identified many distinct traffics — particularly during the peak identified in the first quarter of the trace — although they reported a similar amount of anomalies. This comparison is quickly derived from the two views provided by the tool, whereas, similar conclusions are usually deduced from a time-consuming manual analysis of the two anomaly detectors outputs.

E. Temporal-Spatial patterns in anomalous traffic

During our experiments we observed particular patterns that stood for certain kinds of anomalies. These patterns exhibit some important properties of the anomalies such as its range of targets and sources, its operational speed, and its time duration. It also provides certain information on the mechanisms used by the anomalies, particularly the uses of the source ports.

1) Coarse view: At large scales certain anomalies are easily identified as sudden changes in the main traffic behavior or in the usage of a particular protocol. For example, Figure 8 displays three months of darknet traffic recorded while the first two versions of the Conficker worm were released. This figure shows that first, a sharp increase in the number of source IP addresses and number of packets clearly signaling the start of the worm spread (labeled Conficker.A in Fig. 8). Second, another growth of these quantities depicts the release of the second version of the worm and its aggressive behavior in terms of the network resources consumption (labeled Conficker.B in Fig. 8). The scatter plot of the destination port (see middle scatter plot of Fig. 8) reveals that the first version of the worm is communicating with the other hosts using random port numbers ranging over [1024, 5120]. These types of communications disappear after the second release is unveiled, highlighting that different mechanisms are implemented in this new version.

2) Fine view: On smaller scales, we observe other kinds of patterns exhibiting anomalies through their abnormal uses of the traffic features. We emphasize that these patterns are in accordance with those identified by an anomaly detector based on pattern recognition [4]. For example, Figure 9 is composed of different anomalies observed on the same day (2004/10/14). The vertical axis represents the destination addresses for scatter plots at the top of the figure and source ports for those at the bottom. Three different anomalies are emphasized in this figure.

The two representations ((A) and (B)) on the left-hand side of Fig. 9 stand for an exploit against a Windows service operating on port 445. These were obtained by displaying only the traffic related to a specific IP address, X. The upper representation (A) shows long vertical lines meaning that X contacted numerous hosts within three short periods of time. The header information revealed that all the packets corresponding to these connections were directed to port 445 with the TCP SYN flag set. The representation of the source port (B) indicates that the traffic was initiated from a limited pool of high number ports (< 1024). This traffic is clearly malicious and corresponds to a probing process looking quickly for victims.

The two scatter plots labeled (C) and (D) in Fig. 9 stand for network traffics from a single host lasting for the entire traffic trace. The upper scatter plot displays long oblique lines, meaning that this traffic also correspond to a probing process. However, the inclination of the lines indicates a slower process than the one previously discussed. Moreover, the lower scatter plot (labeled (D)) shows a horizontal line representing only a couple of source ports.

The two representations, (E) and (F), on the right-hand side of Fig. 9 correspond to a spreading of the Sasser worm. Traffic from different hosts are displayed in these figures. The vertical structures in the upper scatter plot represent the probing proce-
dure done by the worm, and we noticed that different spreading are observed. The scatter plot representing the source ports (labeled (F)) indicates that this implementation of the Sasser worm generates traffic with only low source ports numbers that are linearly increasing. The shape and height of the observed “lines” provides a signature for this variant of the worm that can be easily identified in other traffic traces.

V. CONCLUSION AND FUTURE WORK

We outlined the need for understanding the network traffic behavior and evaluating anomaly detectors. To achieve these purposes, we designed and implemented a tool graphically representing the network traffic on any temporal and spatial scales. The main contribution of this tool is to display global and detailed views of the network traffic focusing on anomalies. Interesting traffic behaviors are uncovered by interactively exploring the traffic traces, and detailed information is also provided to enable data to be thoroughly investigated. Traffic from specific hosts or services is extracted by using a filtering mechanism. Thus, particular types of sub-traffics are displayed without surrounding noise and can easily be investigated. Furthermore, anomalies reported by anomaly detectors are highlighted in full view and their validation can then be facilitated. The tool runs on different platforms, licenced under the GNU General Public License (GPLv3), and is freely downloadable. We verified the usefulness of our tool by evaluating it on several traffic traces; darknet traces highlighting several patterns for different anomalies, and traces taken from a backbone link where anomalies surrounded by heavy noise were still identifiable. Observation of recent threats, such as the Conficker worm, can also be carried out.

We conducted manual inspections of the alarms reported by an anomaly detector and visually compared the outputs of two distinct approaches. Also, we listed several patterns standing for distinct anomalies and noticed that they are consistent with those found in [4].

One important project we intend to carry out in the future is to add a capability to process raw packets taken directly from a network interface.

Acknowledgments

We would like to thank G. Dewaele et al. for having provided us the source code of their anomaly detector. This work is partially supported by MIC SCOPE.

REFERENCES


Interference Reduction in Overlaid WCDMA and TDMA Systems

Maan A. S. Al-Adwany1 & Amin M. Abbosh2
1University of Mosul/ College of Electronics Eng., Mosul/IRAQ
2 School of ITEE, The University of Queensland, Australia

Abstract — In this paper, the performance of WCDMA uplink system for UMTS mobile communications is evaluated. Also, the possibility of increasing mobile communication cell capacity through merging WCDMA and TDMA systems in one cell is investigated. An interference canceller is proposed to reduce, or even completely cancel, the interference between WCDMA and TDMA, hence enabling them to work together. This results in a considerable increase in the cell capacity. The coexistence of WCDMA and TDMA systems in one cell is proven to be possible via computer simulations.

Index Terms—Cell capacity, WCDMA, TDMA, interference.

I. INTRODUCTION

Wideband CDMA (WCDMA) is a wideband Direct-Sequen Code Division Multiple Access (DS-CDMA) system, where user information bits are spread over a wide bandwidth by multiplying the user data with quasi-random bits (chips). In order to support very high bit rates, variable spreading factors and multic和平 code connections are used.

Increasing cell capacity in mobile communication systems is one of the important objectives for those systems. Given the constraints of the high cost and the limited availability of radio spectrum, efficient spectrum usage is fundamental to the economic success of new generations of cellular systems. One novel concept to use available resources more efficiently is the combination of existing radio systems into coordinated, hybrid systems, in order to utilize the strength and capabilities of the individual systems. The combination of systems adds a degree of freedom in the sense that at any time, the most spectrally efficient transport system can be chosen, depending for example on quality-of-service requirements and traffic characteristics.

It has been clearly implied by several authors that a need for future multiple access schemes employing an overlay is worth studying and simulating [1-3]. Simulation algorithms were used to study effect of overlaying different schemes on performance of the hybrid system.

The effect of using notch filters to enhance capacity of the hybrid system was studied in [4], whereas the effect of power control mechanism on the bit error rate of the combined schemes was investigated in [5]. A simplified model to investigate the capacity implications when both GSM and CDMA systems operate at the same band was used in [6].

In [7], a system was proposed to combine a TDMA based cell structure with CDMA using direct sequence spread spectrum communication to increase the capacity by time multiplexing cell operations between forward and reverse links. In another investigation [8], a new system for cellular mobile applications was proposed by utilizing TDMA to multiplex users within each cell and CDMA to control inter-cell interference. It was revealed, by comparative simulations, that the proposed system increased the total system capacity. The performance of two spread spectrum systems has also been compared in [9]; the first is a WCDMA system in the classical sense, while the second system employed spread-TDMA for the reverse link. It was suggested that TDMA and WCDMA should be used in indoor and outdoor cells, respectively. A theoretical investigation into the possibility of using a frequency overlay of a narrowband CDMA system and a TDMA system to provide a greater spectral efficiency was presented [10]. It was shown that under certain conditions the two systems can operate in the same frequency band and in the same area with a considerable improvement in the overall capacity of the whole system.

In this paper, a simulator for the uplink UMTS system is implemented using MATLAB, and the effect of dispersive and AWGN channels on system performance is investigated. The investigation is extended to consider the coexistence of TDMA and WCDMA and its impact on system Bit Error Rate (BER). The main aim for the proposed approach is to increase cell capacity through the joint operation of the two systems within the same cell. Also, we propose a simple interference canceller (IC) to reduce the interference between the TDMA and the WCDMA systems.
This paper is organized as follows: Section II gives a preliminary description of the WCDMA uplink simulator, and its BER performance under the effect of AWGN and dispersive channels. Section III presents the proposed joint WCDMA-TDMA approach, while section IV considers interference cancellation. Section V discusses the obtained results, and finally the conclusions are drawn in section VI.

II. DESCRIPTION OF WCDMA UPLINK SYSTEM

In the WCDMA (UMTS) uplink communication system; two physical channels are dedicated for every mobile station; the data channel and the control channel. The data channel is used to convey information data like voice and images, while the control channel is used to carry control signals. A simple schematic diagram for uplink WCDMA system is shown in Fig. 1.

First, the data and control channel bits are modulated using BPSK modulator, and then they are spread with Orthogonal Variable Spreading Factor (OVSF) short codes. After that the two channels are orthogonally related using the complex property. The resultant signal is then scrambled by multiplying with long Golden code. The Golden codes are complex long sequences with low cross correlation, and relatively high autocorrelation. The scrambled signal is then transmitted and propagated over channel.

At the base station receiver, a reciprocal process is achieved. After descrambling the received signal, the data and control channels are separated simply as shown in Fig. 1. Then a matched filter is used to estimate the transmitted BPSK symbols through using integrate and dump (matched filter).

A. Uplink System Simulator

We have built the baseband WCDMA system simulator using MATLAB/SIMULINK. The simulator consists of number of mobile stations (users), the dispersive channel and/or Additive White Gaussian Noise (AWGN) channel, and the base station receiver.

In this simulator, we consider some assumptions for the sake of simplicity. Firstly, a one cell model is used, and hence there is no co-channel interference; secondly a perfect power control is adopted so that all users have the same signal power at the base station receiver; thirdly, every user has its own Orthogonal Variable Spreading Factor (OVSF) and scrambling codes; and finally, full synchronization is assumed between the transmitter and receiver spreading codes.

In our simulation, we select the received Bit Error Rate (BER) as a criterion for system performance evaluation. Each result for BER is obtained through running the simulator for ~10^4 symbols.

B. Simulation Results for Uplink system

After running the simulator, the resulting signal waveforms are displayed for each system stage. Fig. 2 illustrates these waveforms and their locations according to diagram shown in Fig. 1.

Fig. 3 shows variation of the received BER with respect to number of users, when the transmitted signal is propagated over dispersive and dispersive AWGN channels. The spreading factor (SF) is taken to be equal to (8). Also, the information bit rate is 480 kbps. It is obvious that the BER increases with the increase of number of users, and it is also clear that for the same number of users, an addition of AWGN will slightly increase the BER because the effect of the dispersive channel dominates.
Fig. 2: Signal waveforms taken from selected test points.
The behavior of the system BER over dispersive channel when varying the spreading factor is depicted in Fig. 4. It is clear that the system BER is improved when the spreading factor is increased from 8 to 16 for the same number of users. This is because increasing SF reduces the noise power per Hz; which leads to an improvement in SNR at the receiver and hence reduces the BER. The effect of AWGN variance on the system performance is shown in Fig. 5. The increase in noise variance increases the probability of receiving error bits. Also, the system becomes more robust against noise when increasing the spreading factor.

III. EFFECT OF COEXISTENCE OF WCDMA AND TDMA SYSTEMS ON CELL CAPACITY

The aim of this work is to enable two different mobile communication systems to operate in the same cell, and thus increasing the cell capacity to make it ideally equal to the sum of the two systems’ users. The two candidate systems are WCDMA and TDMA, since the two systems are relatively robust to interference from each other.

A. Description Of The Hybrid System Operation

In this subsection, we explain how the CDMA and TDMA systems behave when they operate in the same cell. Fig. 6 shows the principle of the spreading and the de-spreading process. The useful signal is spread and then transmitted over a wideband channel and de-spreading in the
receiver. As can be seen in Fig. 6, the de-spreading process in the CDMA receiver operates as spreading process for a narrowband interfering signals. In terms of coexistence of TDMA and CDMA systems, the narrow band TDMA channels act as narrowband interferer for the CDMA system if they exist in the same frequency band. From the TDMA system’s point of view, the CDMA system acts like a noise-like interferer with a low power spectral density. The receiver filter in the TDMA receiver will ensure that the noise-like interfering signal will only be received in the narrow bandwidth of the TDMA system. Thus, only a small part of the CDMA interference power will be received [10], see Fig. 7.

**B. the Baseband WCDMA/TDMA System**

Time-division multiplexing (TDM) is the interleaving of several digital messages into one digital message with a higher bit rate; as an example, we consider the generation of TDMA wireless system which is similar to that of the European base group, called E1, at 2.048 Mbit/s, which is obtained by multiplexing 32 PCM coded speech signals that are belonging to 32 users at 64 kbit/s [11].

As an example of WCDMA system, we used the UMTS system [8], where the bit rate is chosen to be 240 kbps, and the spreading factor is 16. Hence, the chip rate would be 3.84 Mchip/s, which is the same chip rate for the scrambling codes. More detailed informations about UMTS system can be found elsewhere [12].

However, we propose a one cell model which has both TDMA and WCDMA mobile station transmitters, and a base station which contains two receivers; one for the WCDMA and the other for the TDMA. It is worth to mention that for the sake of simplicity, our simulation is concerned with the uplink system only.

Fig. 8 shows a simple schematic diagram for the proposed model.
IV. DESCRIPTION OF THE PROPOSED INTERFERENCE CANCELLER (IC)

Fig. 9 illustrates a schematic diagram of the proposed interference canceller. The idea is to extract the TDMA interference through the use of a low pass FIR filter which is matched to the TDMA bandwidth (2.048 MHz), and then subtract the extracted TDMA signal from the received composed signal (TDMA+WCDMA). Hence, it is possible to get a WCDMA signal with approximately no TDMA interference.

A threshold circuit is used to eliminate the residual WCDMA signal which passes with the extracted TDMA signal as shown in Fig. 9. The filter has a cutoff frequency of 2.048 MHz which is matched to the TDMA system bandwidth. Also, when designing the filter, the group delay of the filter is to be taken into consideration. It must not have a large group delay (group delay= \( \frac{\text{filter order} - 1}{2} \)). In our design, we select a reasonable filter order of seven; which gives a group delay of three sample periods. Table 1 shows the parameters of the designed FIR low pass filter.

| TABLE I  |
|---|---|
| **FIR FILTER PARAMETERS** |  |
| \( h(n) = a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3} + a_4 z^{-4} + a_5 z^{-5} + a_6 z^{-6} + a_7 \) |  |
| a1 | 0.154896 |
| a2 | 0.223566 |
| a3 | 0.25 |
| a4 | 0.223566 |
| a5 | 0.154896 |
| a6 | 0.070548 |
| a7 | 0.070548 |

Since the signal will suffer a group delay of three sample periods when it passes through the designed FIR filter, this delay must be compensated by adding a delay line (\( z^{-3} \)) for the purpose of signals alignment. Also, it is worthwhile to mention that all the delays and their compensations are taken into account throughout the simulation.

V. SIMULATION RESULTS AND DISCUSSION

The simulation is run under enough number of data symbols, and we neglected the effect of the additive white Gaussian noise in order to study the effect of TDMA interference more clearly.

Fig. 10 illustrates the variations of the BER at the WCDMA receiver with respect to the number of TDMA users (\( N_t \)) for two cases: one by using the proposed interference canceller (IC), and the other without using the IC. Also, the simulation is run for one WCDMA user (\( N_c=1 \)). It is clear that increasing \( N_t \) will increase the BER due to the increase in the number of TDMA interferers. Also, the use of the proposed IC has reduced the BER by about three times compared with the case of not using the IC. The increase in the BER by increasing \( N_t \) gives an indication that some of the TDMA signal power is still passing through the IC and causes an increase in the BER. This is considered as a disadvantage, since we expect a nearly constant BER by increasing \( N_t \) as an ideal case. It is possible to justify this by the non ideal characteristics of the FIR filter.

The relation between the BER at the WCDMA receiver and \( P_t/P_c \), is shown in Fig. 11, where \( P_t \) is the TDMA transmitted power and \( P_c \) is the WCDMA transmitted power from the mobile station. The simulation is run for one WCDMA user, and for 32 TDMA users. It is clear...
that, in general, increasing $P_t/P_c$ will increase the BER. But on the other hand, we can observe a reduction in the BER by a factor of one-fourth in average because of using the proposed IC.

Fig. 12 clarifies the variations of the BER with respect to $P_t/P_c$ when using the proposed IC for 32 TDMA users, and for different numbers of WCDMA users ($N_c$). It is obvious that for the same $P_t/P_c$, increasing $N_c$ will increase the BER due to increasing WCDMA interferes. Also, we can observe that, in general, increasing $P_t/P_c$ will increase the BER because of the non ideal characteristics of the FIR filter.

Fig. 11. Variation of BER with $P_t/P_c$ for one WCDMA user and for $N_t=32$.

Fig. 12. Variation of BER with $P_t/P_c$ for different numbers of WCDMA users and for $N_t=32$, by using IC.

Now, to get a clear vision about the performance of our proposed IC, we define an improvement factor as the ratio of the BER without using the proposed canceller to that when using the canceller. Fig. 13 shows the effect of $N_c$ and $P_t/P_c$ on the improvement factor, keeping the number of TDMA users constant (i.e $N_t=32$). It is possible to observe that the improvement factor increases with decreasing $N_c$. This can be interpreted as follows: decreasing the WCDMA interferers makes the TDMA interference the dominant interference, and hence suppressing the TDMA interference (using the proposed IC) will cause a considerable improvement in the BER. Also, it is observed that the maximum improvement factor is achieved at $P_t/P_c=2$.

Fig. 13. Variation of the improvement factor with $P_t/P_c$, for different numbers of WCDMA users and for $N_t=32$.

VI. CONCLUSIONS

It has been shown that the WCDMA and TDMA systems can work in the same cell and hence, it is possible to increase the cell capacity as the problem of cross interference between the two systems can be reduced using the proposed interference canceller. In WCDMA base station receiver, the BER can be considerably reduced by using the proposed interference canceller. On the other hand, in TDMA base station receiver, the effect of WCDMA interferers can be reduced by increasing the transmitted power of the TDMA mobile station.

ACKNOWLEDGEMENT

The authors acknowledge the assistance introduced by Mr. Sedqi B. Thanon from College of Electronics Engineering-Mosul University.

REFERENCES


GCAD: A Novel Call Admission Control Algorithm in IEEE 802.16 based Wireless Mesh Networks

Floriano De Rango, Andrea Malfitano, Salvatore Marano
D.E.I.S. Department, University of Calabria, Italy
Email: {derango, amalfitano, marano}@deis.unical.it

Abstract—In this paper, a GCAD-CAC (Greedy Choice with Bandwidth Availability aware Defragmentation) algorithm is proposed. It is able to guarantee respect for data flow delay constraints defined by three different traffic classes. To achieve good results, the algorithm tries to accept all the new requests, but when a higher priority request is received, a lower priority admitted request is preempted. This preemption can leave some small gaps which are not sufficient for new connection admission; these gaps can be collected by the GCAD algorithm by activating a bandwidth availability based defragmentation process. The quality of the algorithm is shown by a comparison with two other algorithms found in the literature.

Index Terms— Call admission control, mesh networks, WiMAX, IEEE 802.16.

I. INTRODUCTION

In distributed mesh mode, defined by IEEE 802.16 protocol [1], when a mesh node has an amount of new data to transfer to a destination node, it requires the instauration of a new connection in the neighbour node. This last node has to decide whether to admit the new call, and obviously, how much bandwidth to allocate to the new connection, for the service lifetime. The first is the admission decision, the second one involves the bandwidth to grant to the node for the admitted connection.

Both the decisions are inherent in bandwidth utilization in the network and influence the desired QoS (Quality of Service) level: the arrival of a new connection, can modify the allowed bandwidth to the existing connections, thus, all the QoS constraints must be reviewed. Therefore there is a “risk” in this choice, because, by admitting a new connection, we risk a deterioration in the QoS provided to old connections. The first of the previous listed process decisions is called call admission control, and this decision influences the network bandwidth utilization for a long time, i.e. it is a long term decision. The second one, instead, is a short term decision and defines the amount of bandwidth to grant to the requesting node.

In this paper we present a new CAC algorithm, referring to a mesh scenario which takes into account a set of three traffic classes with different priority levels. The focus is guaranteeing the respect of QoS constraints defined in term of end-to-end delay to higher priority flows. To do this the distributed CAC algorithm admits all the new calls in a greedy way, thus the lower priority flows can exploit the bandwidth availability until a set of higher priority calls claim bandwidth. The lower priority admitted calls are thus preempted to leave room for the new calls. The preemption process has a negative side, it can leave little gaps in the data subframe which are not useful. The solution is the presence of a defragmentation process started by the granter node.

In the rest of this paper our algorithm GCAD-CAC (Greedy Choice with Bandwidth Availability aware Defragmentation) and an evaluation of its performance is presented in detail.

II. IEEE 802.16 MESH MODE

The IEEE 802.16 protocol supports two operating modes: Point-to-Multipoint (PMP) and mesh mode. In the first mode, the base station (BS) has the most important role in coordinating the transmission, i.e. in downlink, the BS is the only station capable of broadcasting data and control messages. On the other hand in uplink, the subscriber stations (SSs) have to use the bandwidth as defined by BS in a centralized way. In mesh mode the BS loses the central role of coordinator, and the coordination can be made in a distributed manner, BS being distinguishable among mesh nodes, because it is the only gateway to reach the rest of the world. In this way, neighbour (a neighbour of a node is a node one step away from it), neighbourhood (the set of all neighbouring nodes) and extended neighbourhood (containing, in addition to the neighbourhood node, the neighbours of neighbours) are new terms unknown to PMP mode.

Given these definitions, we can enunciate the basic principle of transmission coordination in an 802.16 mesh network: no node can transmit on its own initiative, but it can transmit only when it is requested to do so by the node it is connected to.

A. Distributed Coordinated Scheduling

The mesh mode supports only TDD (Time Division Duplexing) mode; the frame is divided into two parts: the first one is the control subframe and the second one is the data subframe. The data subframe is divided into a fixed...
number of minislots, the control subframe can be of two different types:
- network control subframe: used to transport network control messages such as MSH-NENT and MSH-NCFG, used by nodes to acquire network synchronisation and network configuration properties.
- scheduling control subframe: used to collect bandwidth requests and to send grant messages. MSH-CSCF and MSH-CSCH are the messages related to the centralized scheduling mode, while MSH-DSCH is used in distributed mode.

The network control subframe is not present in each frame and it appears periodically. The Scheduling Frame parameter, broadcasted in the MSH-NCFG message, defines how many frames have a scheduling control subframe between two frames with network control subframes.

The dispatch of the previous messages (except MSH-NENT) occurs in a collision free manner and to guarantee it, each node inserts two fields in the control messages:
- \( xmt \) holdoff exponent
- \( next \) xmt mx.

Each node, during message forwarding (except MSH-NENT), calculates its next transmission instant and expresses it in the form of an interval using the two mentioned fields. In practice, the node does not inform the neighbours about its next transmission instant, but it sends an interval in which the next transmission falls in; this interval is defined by the following constraints:

\[
\text{next } xmt \text{ time} > 2^{xmt \text{ holdoff exponent} + \text{next xmt } mx} \quad (1)
\]
\[
\text{next } xmt \text{ time} \leq 2^{xmt \text{ holdoff exponent} + \text{(next xmt } mx + 1)} \quad (2)
\]

Between one transmission and the next, a node must wait for a time interval equal to:

\[
xmt \text{ holdoff time} = 2^{(xmt \text{ holdoff exponent} + 4)} \quad (3)
\]

Scheduling information will be issued in a request - grant format. While respecting the constraints of coordinated distributed scheduling, uncoordinated distributed scheduling can ensure fast setup communications. Uncoordinated scheduling is determined by direct requests and grants between two nodes and it must also take place in a way so as not to cause collisions with the messages and the coordinated scheduling traffic. Both modes of distributed scheduling, coordinated or not, use a three-way-handshake protocol.

A mesh node, that has data to transmit, sends a request in an MSH-DSCH message to a destination mesh node, indicating the requested number of minislots it needs. The destination node replies with a grant message which is acknowledged with a grant copy by the requester node.

B. QoS in Mesh Mode

The only claims made by the protocol for QoS issues, state that the quality of service must be guaranteed packet by packet, in the link context. It must be the node, within the constraints of the distributed bandwidth allocation algorithm, to ensure compliance with the constraints of the individual application quality. Thus, to realize and satisfy QoS constraints, the protocol defines specific fields within the PDU header. The generic header of a MAC PDU, contains a 16-bit CID field. In mesh mode, the CID field is split into two parts, the first portion of 8 bits is the logical network identifier and the second portion contains the link identifier.

C. Call Admission Control in WiMAX mesh networks

IEEE 802.16 technology is still an open question. There are few works that fill the gaps in the protocol. This is true for the mesh mode and even more for algorithms related to distributed mode. The same call admission topic in mesh mode is neglected by literature. Instead, some research into distributed scheduler performance is available, in particular [2] analyzes distributed scheduler performances and illustrates how to set the \( xmt \) holdoff exponent parameters dynamically. The optimization of mesh scheduling is described in [3] evaluating a combined centralized - distributed scheduling. The authors of [4] only propose an improvement in the distributed mesh scheduler. There is a proposal for a call admission control algorithm in distributed mode in [5]; in this last paper the concept of connection preemption with some limitations is presented; three traffic classes with assigned priorities are considered, the admission algorithm based on the concept that all the bandwidth can be divided among the three classes, but in this way in a steady state, the advent of new data flows with higher priority are refused because these classes have consumed the bandwidth reserved for it; instead, new data flows with lower priority can be admitted. Also the scenario used to validate the proposed CAC algorithm is very simple, the maximum path length in scenario is two hops. In [6] the authors propose an end-to-end bandwidth reservation scheme with a CAC algorithm which only refers to VoIP traffic. In [7] a simple CAC algorithm is proposed. The authors consider a traffic differentiation using the priority field of unicast CID. The CAC algorithm is based on a threshold mechanism. Requests with higher priority, if there are sufficient free minislots, are always admitted, whereas the low priority requests are refused in cases of congestion, which is verified with a bandwidth utilization computation. If bandwidth utilization is greater than a fixed threshold then low priority requests are refused. Also, the simulated scenario is too simple, each node is a neighbor of each other node in the network. The paper [8] describes a CAC algorithm related to PMP mode; it is very interesting for the connection preemption concept that is introduced and the admission decision is based on traffic class and bandwidth utilization of each traffic class. Each traffic class has a bandwidth portion reserved to it and can also preempt the lower priority admitted calls. Other interesting works focusing on call admission control in PMP mode are [9] - [13] and in particular, although it considers the PMP mode, the work [14] is to be taken into account to enrich our knowledge, in fact, the authors...
of [14] apply the Games Theory ([15], [16]) to the call admission issue.

The contribution of our work, can be considered important in the context of research into 802.16 mesh distributed architecture, because, to the best of our knowledge, in literature there are few works about CAC in mesh distributed mode.

Our intention is to present a distributed call admission control algorithm which takes into account three different traffic classes. The proposed GCAD (Greedy Choice with bandwidth aware Availability Defragmentation) algorithm has two interesting processes: preemption and defragmentation. Preemption occurs when there is a new call with higher priority, which can preempt a call with lower priority. The preemption process can cause a fragmentation in the data subframe, i.e. we can find, in the data subframe, some small unusable gaps of free minislots. The defragmentation process collects these gaps creating continuous availability.

IV. GCAD: A NEW CALL ADMISSION CONTROL ALGORITHM

We propose a CAC algorithm for an IEEE 802.16 distributed mesh network, each mesh node can support three different data traffic classes with “1”, “2” and “3” as priority values. The values “1” and “3” are the highest and the lowest priority values respectively. When a new source starts to transmit data, it has to identify a path to send data to the destination node. Subsequently, the mesh node can submit a bandwidth request to the next hop node. In this section, we describe our proposal for the source and next hop node behavior. The first one is described in terms of bandwidth estimation, or more correctly minislot number estimation, and the last one in terms of the call admission control process. In the following we indicate the source node as requester and the next hop node as granter. Obviously the next hop node in turn becomes a requester and so on.

A. Minislot Number Request Estimation

Each node has a data queue and when a packet appears in the queue, the node creates a bandwidth request. The node can classify the queued packets using the priority field present in the unicast CID. The three traffic classes can have QoS constraints expressed in terms of end-to-end delay, thus, the node has to estimate the amount of minislot requests. Each data subframe is divided into a fixed number of 256 minislots.

We define the MSNEA component; MSNEA is a Mini Slot Number Estimation Algorithm and its challenge is to determine the amount of bandwidth which a node needs. When a node, or for greater accuracy, the algorithm or agent designed with the task of observing the data queue, realizes that the data queue is not empty, it is necessary to request bandwidth to send the data packets. In particular in the IEEE 802.16 mesh scenario the frame is divided into two parts:

- control subframe
- data subframe.

The data subframe is the only part of the frame used to transmit data packets and it is divided into a well defined number of minislots. Consequently, when a node has to request bandwidth, the number of minislots needed has to be evaluated.

The evaluation of this quantity may seem simple but it is very important for two reasons:

- if the estimated number of minislots is smaller than the number which the node actually needs then it becomes difficult to ensure that there is not an accumulation of data packets in the queue and it is also very hard to try to guarantee respect for quality constraints;
- if the estimated number of minislots is greater than the number which the node actually needs then there is a waste of bandwidth.

The previous concepts illustrate the importance of the presence of an efficient estimation algorithm. The MSNEA is described by the flow chart in figure 1. In addition to the flow chart it is necessary to provide a set of “conditions” which are used in it. It is also necessary to define the following parameters:

- $MS$: OFDM symbol number for each minislot;
- $p_{size}$: packet size (bits);
- $eff$: efficiency of an OFDM symbol, expressed as number of data bits for each symbol;
- $dl$: delay constraint;
- $d_{sym}$: OFDM symbol duration (s);
- $f$: frame duration (s);
- $h$: path to destination hop count;
- $t_{a}$: arrival time of the first queued packet of BE traffic;
- $t_{b}$: arrival time of the last queued packet of BE traffic;
- $n_{BE}$: number of BE queued packets;
- $p_{mean}$: mean packet size of BE queued packets;
- $R$: estimated BE rate;

And with these parameters we can estimate the $nms$ request (number of minislots) for traffic with priority equal to 3 using the equations:

$$R = \frac{p_{mean} * \left(n_{BE} - 1\right)}{t_{f} - t_{i}}$$

(4)

$$nms = R * \frac{f}{MS * eff}$$

(5)

and the $nms$ request for traffic with priority equal to 1 or 2 resolving the following:

$$(nms * MS * d_{sym}) + \left(\frac{p_{size} * \left(n_{BE} - 1\right)}{n * MS * eff}\right) * f = \frac{dl}{h}$$

(6)

Now our intention is to explain the behavior of MSNEA following the flow chart depicted in figure 1 and also using equations (4) and (6) and the new equation:

$$\frac{dl}{h} - (t_{now} - t_{last}) \leq \frac{Total\Byte\ +\ Add\ Byte\ \times f}{nms * MS * eff}$$

(7)

which is defined using these parameters:

- $t_{now}$: time instant in which the calculation takes place;
The first term of equation (7) is indicated in the flow chart as \( t_{need} \) and the second term as \( t_{actual} \). Also this equation allows us to calculate the \( nms \) request for traffic with priority equal to 1 or 2. The use of this equation and of the other will be explained below. In the flow chart we indicate equation (5) with the term condition (1); equation (6) as condition (2) and equation (7) as condition (3).

The MSNEA is invoked by the node at the instant in which the node has a MSH-DSCH to send, in this way MSNEA can evaluate the possibility of making a new bandwidth request for an existing data flow or for a new data flow. The first step made by algorithm is the extraction of the first PDU from the data queue, we indicate this PDU as PDU1. The PDU1 belongs to a traffic class with a priority indicated as PDU1.priority; MSNEA verifies the presence of an existing pending request for bandwidth for data flow with this priority. If a pending request exists then the MSNEA searches for the presence of other PDU in the queue with a different priority value. If the MSNEA finds this PDU then it also verifies for this priority the presence of pending requests. If the algorithm finds a pending request then it repeats the process for the last priority value. When the MSNEA does not find a pending request, it scans the data queue searching for all PDUs with priority equal to PDU1.priority. During the queue scan, the algorithm evaluates a set of parameters and in particular :

- \( t_{last} \): time instant corresponding to the arrival of the last queued packet;
- \( \text{Total}_byte \): total bytes present in queue and referring to the same traffic class.

The first variable that the MSNEA considers is the following parameters:

- \( \text{Start}_time \): the arrival time instant of the first queued PDU with priority equal to PDU1.priority.
- \( \text{End}_time \): the arrival time instant of the last queued PDU with priority equal to PDU1.priority.
- \( \text{Total}_byte \): the total number of bytes related to all the PDUs queued with priority equal to PDU1.priority.
- \( \text{Card}_pack \): is the number of queued PDUs with priority equal to PDU1.priority;
- \( \text{Interval} \): is defined as

\[
\text{Interval} = \text{End}_time - \text{Start}_time. \quad (8)
\]

It represents the time interval which elapses between the two arrival time instants of the first and last queued PDUs with priority equal to PDU1.priority.

- \( \text{PS}_\text{mean} \): is defined as

\[
\text{PS}_\text{mean} = \frac{\text{Total}_byte}{\text{Card}_pack}; \quad (9)
\]

represents an estimation of the mean packet size.

- \( \text{Dt}_\text{mean} \): is defined as

\[
\text{Dt}_\text{mean} = \frac{\text{Interval}}{\text{Card}_pack}; \quad (10)
\]

represents an estimation of the time rate of queued PDU with priority equal to PDU1.priority.

At this point, if the MSNEA does not find already active grants for data flow with priority equal to PDU1.priority, it has to make the first request and it has to estimate the number of minislots \( (nms) \) on the basis of PDU1.priority. If the PDU1.priority is equal to 3, then the estimation is made by condition (1) i.e. by equation (5), instead if PDU1.priority is equal to 1 or 2, then the estimation uses condition (2), i.e. equation (6). Otherwise, if the node has an active grant for the same priority the MSNEA has to evaluate a set of conditions to decide if it is necessary to make a new request. To support the decision, the algorithm evaluates the state of three Boolean variables: First_alarm, Second_alarm and Constraint_alarm. To assign a value to these variables the algorithm elaborates the following parameters:

- \( \text{temp}_\text{nms} \): a first estimation of the number of slots that the node needs to deliver the queued PDU with priority equal to PDU1.priority is made by condition (1);
- \( \text{Total}_ms \): the number of minislots that the node already has;
- \( \text{t}_\text{need} \): represents the time slots that the node needs to deliver the queued PDU to its destination, with priority equal to PDU1.priority, respecting the delay constraint;
- \( \text{t}_\text{actual} \): the node, using the minislots previously granted to it for the PDUs with priority equal to PDU1.priority, has an available time interval to deliver the queued PDUs, this time interval is actual; \( \text{t}_\text{need} \) and \( \text{t}_\text{actual} \) are respectively the first and the second terms of equation (7);
- \( \text{Sampled}_\text{queue} \): to verify if there is the need to request more minislots, the MSNEA samples the data queue length, the sampling takes place when the node receives a new grant; this parameter indicates the last sampled value.

The first variable that the MSNEA considers is the First_alarm, this variable is set with “true” if the Total_ms is smaller than the temp_ms and this means that the number of minislots owned by the node is not sufficient to deliver all the queued PDU with priority equal to PDU1.priority. This condition represents a first alarm for the MSNEA. The second evaluated variable is the Constraint_alarm, it is considered only for PDU with priority equal to 1 or 2 and is set to true if the minislots previously granted to the node are not sufficient to guarantee compliance with the delay constraint.
The last variable is Second_alarm and it is set to true if in comparison with the actual value of the queue length it is greater than the sampled_queue parameter; this means that the previously granted minislots are not sufficient to guarantee the disposal of queued PDUs, i.e. if there is not a new grant then there will be a continuous accumulation of PDUs in the queue.

The last verification is useful to understand if it is necessary to make a new estimation for a new request. For MSNEA, if First_alarm and Second_alarm are both
true or the Constraint_alarm is true then it is necessary to make a new request and the nms is evaluated using condition (2) or condition (1) on the basis of PDU1.priority. It is interesting to note that if First_alarm and Second_alarm are both false then if Constraint_alarm is true then MSNEA effects a new request, this means that the number of minislots previously granted to the node are sufficient to dispose of the queued PDUs but not sufficient to guarantee the compliance with the delay constraint. The Constraint_alarm is the most important condition in deciding for a new request. It is necessary to clarify that for PDUs with priority equal to 1 or 2 the minislot number estimation is made using condition (2) and not condition (3). As can be seen condition (3) is only used to set the alarm variable, because the estimation made by condition (2) is smaller than the value obtained by condition (3) and thus we obtain a conservative estimation. By this we mean that it is better to make a new estimation that will probably be insufficient and not make a request that leads to a waste of bandwidth; to remedy insufficient bandwidth there is the possibility of subsequently resting the needs of a new request.

B. Call Admission Control Algorithm

The proposed GCAD-CAC algorithm is described by the flow diagram depicted in figure 2. The parameters expressed in the flow diagram are defined as the following:

- B_A: minislot number available at arrival instant of a new request;
- B_P: minislot number collected by preemption;
- B1_A: total minislot number obtained after a preemption to admit a new request with priority equal to “1”;
- B2_A: total minislot number obtained after a preemption to admit a new request with priority equal to “2”;
- B_D: total minislot number which can be obtained by defragmentation process.

When a mesh node receives a new request, expressed as a number of requested minislots: Rn, it admits all kinds of requests if there is sufficient available bandwidth. This explains why the algorithm is defined as greedy; many CAC algorithms define utilization constraints and refuse a new connection if the traffic class has achieved the utilization threshold. We instead, try to take advantage of the actual minislots available, also, trying to respect all QoS delay constraints.

If a new request arrives with higher priority than a previously admitted one and there are not sufficient available minislots, then admitted calls with lower priority can be preempted. Thus, before preemtping a connection, the granter evaluates the amount of minislots obtainable by preemption: B_P_A.

If the total available minislots (B1_A or B2_A for requests with priority equal to “1” and “2” respectively) is greater or equal to Rn, then the preemption of a previous admitted request ci with:

\[
\text{Priority}(c_i) < \text{Priority}(R_n)
\]  

then:

![Proposed algorithm for call admission control.](image)

An important condition to be considered is the following: B1_A and B2_A are evaluated considering only contiguous minislots.

For example, considering the allocation scheme depicted in figure 3, a new request, with priority equal to “1”, can preempt the “e” and not “d” allocation because only “e” is contiguous with the available minislots. After preemption the new request is admitted. The granter, in this case, advises connection “e”, that is preempted. In order to advise the preempted connection, we send a grant message with minislot range field equal to “0” to
the owner node and the preempted connection can make a new request to try to obtain free minislots.

Figure 3. Data subframe with minislot allocations.

After the preemption test, if a new request with priority equal to “1” or “2”, does not have sufficient available minislots, the granter node can activate defragmentation to collect fractioned available minislots in a whole availability.

Figure 4. Data subframe states: (a) before preemption; (b) after preemption and finally (c) after defragmentation process.

In figure 4, it is possible to note the case in which there is an advantage in defragmentation utilization. The rectangle without numbers represents free minislots. Case (a) represents the data subframe before a preemption due to the arrival of a new request with priority equal to “2”; the data subframe state after preemption is represented in case (b), the preemption causes the presence of a free minislot gap between two allocations; with defragmentation two gaps are unified and a new request can be admitted as in case (c). To effect defragmentation, the granter sends a grant message with a range equal to “0” to all the interested nodes. In this way the granter node advises the defragmented connection owners to bargain for the new grants. The granter, obviously, in the “bargaining” process, allocates minislots in a contiguous way, and admits a new request, in advanced minislots, only after re-allocation of the defragmented connections.

V. PERFORMANCE EVALUATION

To test the proposed algorithm, we designed a network simulator for IEEE 802.16-2004 [1] protocol in JAVA language. In the simulator we implemented our algorithm and also other two algorithms to make a performance comparison. In figure 5 the mesh simulated scenario is depicted. We consider a mesh network with 25 nodes, one of these (number 1) is the BS. The depicted lines represent the active links. The scenario is a square with area: 5 km x 5 km. All the traffic is from mesh nodes to BS node.

A. Simulation Scenario

Table I summarizes all the simulation settings. In each simulation, the source nodes and the packet generation start instants are randomly selected.

Figure 5. Simulated scenario.

The presented algorithm is compared with two other algorithms identified in literature. The first is extracted from paper [7] and in the following is indicated as THR algorithm (THR because it is based on a threshold mechanism). The second algorithm is a CAC algorithm for a 802.16 PMP scenario and is proposed in [8]. It is very promising and it has been adapted to a distributed mesh scenario; in the subsequent sections we refer to it as PMP algorithm. In the final part of this section, THR and PMP algorithms are briefly introduced.

THR is a call admission control algorithm for 802.16 distributed mesh mode. Calls are classified into three different classes and the admission decision is based on a few concepts:
- there is the presence of two checkpoints fixed along the available minislots: cp1 and cp2;
- there is a threshold value for bandwidth utilization;
- if the bandwidth utilization at checkpoint cp1 is less than threshold, all calls are admitted without distinguishing between priorities, otherwise, to admit a low priority call, the node searches for a frame from checkpoint cp2, and if there are sufficient availabilities, the request is admitted otherwise it is refused.

PMP is a call admission control algorithm referred to 802.16 Point to MultiPoint mode. In a call admission decision, the algorithm distinguishes between four different service classes: UGS (Unsolicited Grant Service), rtPS (real time Polling Service), nrtPS (not real time Polling Service) and BE (Best Effort). In our work, instead, three different traffic classes are considered and thus, to import the PMP call admission control proposed in [8] in mesh mode, UGS, rtPS and BE are mapped in traffic classes with priority “1”, “2” and “3” respectively.
With respect to the previous service mapping, the call admission decision is taken following these criteria:

**TABLE I. SIMULATION SETTINGS**

<table>
<thead>
<tr>
<th>SIMULATION SETTINGS</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PHY SETTINGS</strong></td>
<td></td>
</tr>
<tr>
<td>Modulation</td>
<td>OFDM, QPSK 1/2</td>
</tr>
<tr>
<td>BW (Channel Bandwidth)</td>
<td>25 MHz</td>
</tr>
<tr>
<td>NFFT</td>
<td>256</td>
</tr>
<tr>
<td>G</td>
<td>1/8</td>
</tr>
<tr>
<td>Frame length</td>
<td>20 ms</td>
</tr>
<tr>
<td>Symbol efficiency</td>
<td>184 bits</td>
</tr>
<tr>
<td>Coverage radius</td>
<td>500 m</td>
</tr>
<tr>
<td><strong>MAC SETTINGS</strong></td>
<td></td>
</tr>
<tr>
<td>msh-ctrl-len</td>
<td>4</td>
</tr>
<tr>
<td>msh-dsch-num</td>
<td>4</td>
</tr>
<tr>
<td>msh-csch-data-fraction</td>
<td>0</td>
</tr>
<tr>
<td>scheduling-frame</td>
<td>1</td>
</tr>
<tr>
<td>data queue size</td>
<td>50</td>
</tr>
<tr>
<td><strong>SOURCE SETTINGS</strong></td>
<td></td>
</tr>
<tr>
<td>Number of sources</td>
<td>3 - 24</td>
</tr>
<tr>
<td>QoS delay constraints</td>
<td></td>
</tr>
<tr>
<td>Priority: 1</td>
<td>40 ms</td>
</tr>
<tr>
<td>Priority: 2</td>
<td>80 ms</td>
</tr>
<tr>
<td>Priority: 3</td>
<td>no</td>
</tr>
<tr>
<td>Source rate CBR</td>
<td></td>
</tr>
<tr>
<td>Priority: 1</td>
<td>packet size: 64 byte, packets/s: 128</td>
</tr>
<tr>
<td>Priority: 2</td>
<td>packet size: 2500 byte, packets/s: 25</td>
</tr>
<tr>
<td>Priority: 3</td>
<td>packet size: 2500 byte, packets/s: 125</td>
</tr>
<tr>
<td>Simulation run duration</td>
<td>500 s</td>
</tr>
<tr>
<td>Num runs for each configuration</td>
<td>10</td>
</tr>
<tr>
<td>Confidential Interval</td>
<td>95%</td>
</tr>
</tbody>
</table>

- Advent of request with priority equal to “1”: $B_1$ is the bandwidth request with priority “1”. When a node receives the new request, it verifies if the remaining bandwidth is less than the $B_1$ request. If the condition is verified, then the request is admitted, otherwise the mesh node verifies if this condition can become true considering the preemption of previously admitted requests with a priority of less than “1”. If the condition, with the new bandwidth availabilities, becomes true then the request is admitted otherwise it is refused.

- The mesh node receives a request with priority equal to “2”: $B$ is the total bandwidth, $B_2$ is the request with priority “2” and $R_{e1}$ is the bandwidth reserved for calls with priority “1”. If the bandwidth admitted to the previous requests with priority “2” plus $B_2$ is less than $B - R_{e1}$ and if the remaining bandwidth is not less than $B_2$, the request is admitted; otherwise if the first condition is true, we can calculate the remaining bandwidth plus the amount of bandwidth released by preempted connections with priority “3”, if it is not less than $B_2$, the request is admitted otherwise it is refused.

- The request with priority “3” can use the remaining bandwidth, and can be preempted if necessary.

To make a comparison, we test the algorithms using an increasing source number: from 3 to 24. It is equally divided between the three traffic classes. In this way, with a source number equal to 24, we mean that the scenario contains 8 sources with priority “1”, 8 sources with priority “2” and 8 with priority “3”.

**B. Simulation Parameters**

To evaluate performance of the algorithms, we select a set of parameters and use it to make a comparison between the proposed GCAD, PMP and THR algorithms. The performance parameters are the following:

- Packet loss percentage: defined as the percentage of total packets generated by sources and not delivered to destinations. A packet can be lost because the data queue of a mesh node is full, or because a request is not admitted;
- Throughput: the percentage of sent packets received at destination;
- Average number of refused requests: takes into account the average number of requests which are not admitted;
- Average end-to-end delay: the average time interval required by a packet to complete the path from source to destination;
- Delay jitter: is a variability measure of packet delay. The delay jitter is very important for real time application.

**C. Simulation Results**

In figures 6, 7 and 8 the algorithms behaviour, in terms of packet loss percentage, is represented. Figure 6 considers the case of traffic class with a priority value equal to “1”. It is the highest priority traffic class. GCAD gives the best performance and always maintains the percentage under 5%.
The worst case is obtained by the PMP algorithm, as the number of sources increases, the percentage of packet loss, has a tendency to reach high values. Instead the GCAD trend grows slowly as network congestion increases. Also, by observing figures 7 and 8 GCAD shows the best trends. In figure 7 the worst case is related to the THR algorithm, while in figure 8 all the algorithms have a similar response to increasing congestion.

Figure 7. Packet loss percentage of sources with priority equal to “2”.

Figure 8. Packet loss percentage of sources with priority equal to “3”.

Considering the three cases, we can confirm the focus of the algorithms: THR tries to give more importance to priority “1”, neglecting priority “2” and “3”; PMP wants to put the two more important priority traffic classes on a par; GCAD has the same focus as PMP but it gives the best performance due to the presence of the defragmentation process.

The defragmentation process gives the algorithm the capability of accepting a higher number of requests and amount of bandwidth. This is visible in figures 9, 10 and 11. In figure 9 the only algorithm which has refused calls is THR, while figure 10 shows that GCAD is able to obtain a higher number of requests of priority “2” and this is confirmed by the packet loss depicted in figure 7. In figure 9, PMP did not refuse calls with higher priority, but it reached high packet loss values in a congested network, as PMP accepts all requests but gives them small amounts of bandwidth. Both PMP and GCAD behaviour depicted in figure 11 are similar.
In this way, evaluating the packet loss and the average number of refused calls, we can conclude that the introduction of a defragmentation process, allows the bandwidth to be managed in a more optimized way. The elimination of little availability gaps, gives the granter the possibility to create contiguous allocations, with the right size, to admit new calls.

Another way of viewing the capability of each algorithm to allow good results in terms of successfully transmitted packets, is to analyze the throughput performance. The throughput trends are depicted in figures 12, 13 and 14.

Figure 12 shows that our algorithm gives the best performance to sources with priority “1”. THR behaviour is also good, and this is because it preserves a bandwidth portion for sources with a higher priority. Instead, PMP performance, as depicted in figure 12, is characterized by a degradation due to bandwidth reserved for other kinds of traffic. Figures 13 and 14, relating to priority equal to “2” and “3” respectively, also confirm the quality of our proposal. Another point in favor of the GCAD algorithm is due to the greedy choice, in fact, if there is a sufficient number of minislots, it accepts each kind of request, and only in a second moment starts the preemption process if, and only if, necessary.

Figure 15. Average end-to-end delay: sources with priority equal to “1”.

Figure 16. Average end-to-end delay: sources with priority equal to “2”.

Figure 17. Average end-to-end delay: sources with priority equal to “3”.
In figures 15, 16 and 17 average end-to-end packet delays are depicted. Figure 15 considers the priority “1” case. Observing the depicted trends, it is possible to see how the only algorithm, which respects the delay constraint, in each network condition, is the GCAD algorithm (it is necessary to remember that the QoS constraint for data flow with priority value equal to “1” is an end-to-end delay value less than 0.04 s). PMP and THR do not respect the QoS delay constraint in a scenario with 18, 21 and 24 sources.

Also in the priority “2” case the THR algorithm overflows the delay threshold (in this case the end-to-end delay constraint is less than 0.08 s). Instead in the priority “3” case, there are no quality thresholds. The GCAD algorithm does not give the best performance in each case and this is due to the presence of the defragmentation process. On one hand it allows the optimization of bandwidth management, but on the other it pays for this with an imperfect delay behaviour.

Finally in figures 18 and 19 we depict the jitter trends related to priority “1” and “2” cases.

![Figure 18. Average delay jitter: sources with priority equal to “1”.

![Figure 19. Average delay jitter: sources with priority equal to “2”.

The GCAD algorithm, in traffic with priority equal to “1”, gives the best results. Its jitter trend is regular and the values are not great even in a congested network. This delay jitter characteristic is very important in a real-time application. Instead, by observing figure 19, we can see that the delay jitter trend is more irregular, this is surely due to the defragmentation process. In fact, it can introduce variable delays, as it causes a new bandwidth bargaining process of connections involved in defragmentation.

VII. CONCLUSIONS

In this paper we have presented a new call admission control algorithm for 802.16 distributed mesh networks. The algorithm is characterized by an initial greedy choice, by a preemption and a defragmentation process. The proposed algorithm has been tested in a scenario of 25 mesh nodes with a max number of 24 sources. The performance of the proposed GCAD algorithm is evaluated in terms of throughput, average end-to-end delay, average delay jitter, number of refused requests and packet loss percentage. The GCAD performances are compared with those of another two CAC algorithms in literature. The GCAD algorithm gives the best performance due to the presence of a defragmentation process. It allows an optimized management of minislot allocation.

REFERENCES


**Floriano De Rango** graduated in Computer Science Engineering in October 2000, and a Ph.D. in Electronics and Communications engineering in January 2005, both at the University of Calabria, Italy. From January 2000 to October 2000 he had a scholarship in the Telecom Research LAB C.S.E.L.T. in Turin. From March 2004 to November 2004 he was visiting Researcher at the University of California at Los Angeles (UCLA). From November 2004 until September 2007 he was a Research Fellow in the D.E.I.S. Department, University of Calabria where he is now Assistant Professor. He was recipient of the Young Researcher Award in 2007. He served as reviewer and TPC member for many International Conferences such as IEEE VTC, ICC, WCNC, Globecom, Med Hoc Net, SPECTS, WirelessCOM, WinSys and reviewer for many journals such as IEEE Communication Letters, JSAC, IEEE Trans. on Vehicular Technology, Computer Communication, Eurasip JWCN, WINET etc. His interests include Satellite networks, IP QoS architectures, Adaptive Wireless Networks, Ad Hoc Networks and Pervasive Computing. He has co-authored more than 130 papers among International Journal and Conferences Proceedings.

**Andrea Malfitano** was born in Neuchatel (Switzerland) in 1977 and graduated in Computer Science Engineering. He currently is a Ph.D student at University of Calabria and collaborates with the telecommunication group of the DEIS Dept. His main research interests are about topics related to physical and MAC (Medium Access Control) levels of IEEE 802.16 protocol stack. In particular he carries out research into signal impairment effects due to environment, transmission techniques of the IEEE 802.16 protocol, transmission channel modeling, scheduling issues, mechanisms to support QoS and call admission control topics.

**Salvatore Marano**, graduated in Electronics Engineering at the University of Rome in 1973. In 1974 he joined the Fondazione Ugo Bordoni. Between 1976 and 1977 he worked at the IIT Laboratory in Leeds, United Kingdom. Since 1979 he has been an Associate Professor at the University of Calabria, Italy. He was reviewer for many journals such as IEEE Communication Letters, JSAC, IEEE Trans. on Vehicular Technology, IEEE Transaction on Wireless Comm., European Transaction on Telecommunication Journal. His research interests include performance evaluation in mobile communication systems, satellite systems and 3g/4G networks. He has published more than 160 papers in international conferences and journals.
A Secure Protocol for Sharing Trust Data in Hybrid P2P Network

Huaiqing Lin
College of Electronic Engineering, Naval University of Engineering, Wuhan, China
Email: whoamisir@163.com

Yonghong Zhou
College of Electrical and Information Engineering, Naval University of Engineering, Wuhan, China
Email: sunnyzyh@sohu.com

Abstract—The trust data is critical to the trust model of P2P system. In this paper we present an efficient certificateless cryptography scheme and propose a protocol which provides the ability for sharing trust data securely. The protocol avoids the escrow problem identity-based cryptosystem and the secure delivery of private keys. The security of scheme is based on some underlying problems closely related to the Bilinear Diffie-Hellman Problem are computationally hard. It tolerates the Type I and Type II adversary. The proof of security is presented in the random oracle model. Through security discussion, we show that my secure protocol is extremely secure when encounter a variety of possible attacks.

Index Terms—P2P, trust, CL-PKC, bilinear pairings

I. INTRODUCTION

As an emerging model of communication and computation, peer-to-peer (P2P) networking has recently gained significant acceptance. Millions of users share huge amounts of resources by forming an abstract, logical network called an overlay network. Peer-to-peer networks have many benefits over standard client-server approaches to data distribution, including increased robustness, scalability, and diversity of available data. There are many good P2P applications, such as Napster, Gnutella, KaZaa, eDonkey, BitTorrent, Tapestry, Chord, CAN, Pastry, SETI@home, Avaki, Popular Power, JXTA, Magi, Groove, etc. However, the open and anonymous nature of these networks raises several issues. Firstly, the complete lack of accountability for the content a peer puts on the network opens the doors to abuse of these networks by malicious and irresponsible peers. The other is that P2P communities are often established dynamically by peers that are unrelated and unknown to one another. In order to complete the transaction quickly and securely, peers should select a secure, powerful and steady partner, which is however difficult to accomplish in P2P network, since peers cannot know the status of the others.

To resolve afore mentioned issues, one effective way is developing strategies for establishing community-based trust through reputation [1-10]. Reputation system provides a way for building trust through social control without trusted third parties. It can help a peer to select partner with higher reputation when he performs a transaction.

Reputation-based trust model can effectively improve performance of P2P networks. A fair amount of work has been done in the area of computing reputation-based trust ratings. However, only few papers discussed the issue of developing secure underlying protocols to distribute and access the trust ratings in the overlay network.

The purpose of designing an anonymous authentication protocol in P2P systems is motivated by the problem how to securely propagate the peers’ trust data? The accuracy and validity of trust model most rely on the quality of trust data, so the secure access of trust data is critical to the trust model. However the mechanisms of message forwarding in the P2P network are that a message reaches the desired peer after going through a number of other peers in the network. Since anybody can modify the information in the message, the security of trust data can not be insured.

PKI-based secure protocol may be employed to implement secure access of trust data. But the main difficulty in developing secure systems based on public key cryptography is the deployment and management of infrastructures to support the authenticity of public keys. In traditional public key infrastructure (PKI), this is achieved in the form of certificates issued by a trusted certification authority (CA), which induces the problem of certificates management. In [11, 12], Identity-based public key cryptography (IBPKC) was proposed to solve this problem, since the direct derivation of public keys in IB-PKC eliminates the need for certificates.

In 2003, Al-Riyami and Paterson [13] proposed a scheme called certificateless public key cryptography (CL-PKC) which is intermediate between identity-based [10, 11] and traditional PKI-supported cryptography. The concept was introduced to suppress the inherent key-escrow property of identity-based cryptosystems (ID-PKC) without losing their most attractive advantage which is the absence of digital certificates and their important management overhead.

In this paper we present a secure protocol for maintaining and accessing trust rating information. The protocol uses CL-PKC schemes to provide security and is resistant to various attacks. One of difficulties of
deploying CL-PKC in the P2P network is how to select KGC. But in the hybrid P2P network, we can let the server act as KGC. CL-PKC can avoid both the problem of certificates management and escrow problem. In the original construction of CL-PKE in [13, 14, 15], the KGC must ensure that the partial key is delivered securely to the correct entity, while this requirement is canceled in our scheme.

The rest of this paper is organized as follows. Section 2 introduces related works including secure protocol in the trust management schemes, CL-PKC and anonymous P2P protocols. We review the formal definition and adversarial model of CLE schemes in section 3. Our certificateless public key cryptography scheme is proposed in section 4. Section 5 presents secure protocol for sharing trust data. We give the security discuss in section 6, and conclude this work in Section 7.

II. RELATED WORK

A. The secure scheme in the trust model

Very few papers attend to the secure access of trust data. The following are the basic ideas of secure algorithm of EigenRep [10]:

- The current trust value of a peer must not be computed by and reside at the peer itself, where it can easily become subject to manipulation.
- It will be in the interest of malicious peers to return wrong results when they are supposed to compute any peer’s trust value. Therefore, the trust value of one peer in the network will be computed by more than one other peer.

In other words the security of EigenRep just relates to the security of trust computing.

TrustMe [9] is a secure and anonymous underlying protocol for trust management, which is based on the public key cryptography. It was the bootstrap server responsibility for selecting THA-peers (peers which hold the trust value for a particular peer). This kind of approach suffers from following shortcomings:

- How to choose the bootstrap server is a very difficult task to complete.
- The security of system relied on the bootstrap server.
- No discussing data synchronization between multiple THA-peers.

B. Certificateless public key cryptography

In order to avoid the authentication of the public key, [11, 12, 16, 17, 18] introduced the notion of identity-based cryptography (IBC), which was to simplify key management and avoided the use of digital certificates. The corresponding private key is generated for the user by a trusted third party called private key generator (PKG) and is given to the user through a secure channel.

In order to avoid the escrow problem, Al-Riyami and Paterson [13] invented a new paradigm called certificateless public key cryptography (CL-PKC). CL-PKC also uses a third party called Key Generation Center (KGC) to help a user to generate his secret key. However, the KGC only provides a partial private key for each user. The full private key is generated by the user who makes use of the partial private key obtained from the KGC and the secret information chosen by himself. Hence, CL-PKC removes the key escrow problem. The public key of the user is computed from the KGC's public parameters and his secret information, and is published by the user himself.

A more recent work [19] thoroughly investigated the connections between the CLE and CBE scheme. The paper [20] describes a somewhat similar scheme to [19], another work [21] that investigates identity-based and certificateless extensions of key encapsulation mechanisms. Huang et al. [22] pointed out a security drawback of the primal CLS scheme and defined the secuity model of CLS schemes. Zhang et al. [23] and Hu et al. [24] improved the security model of CLS schemes and presented a more efficient CLS scheme. A new kind of ‘Type II attack’-‘Malicious but Passive KGC attack’ is introduced in [25]. In the new attack, the KGC is assumed malicious at the very beginning of the Setup stage of the system.

III. PRELIMINARIES

A. Overview of pairings and bilinear problems

We briefly review the necessary facts about bilinear maps and bilinear map groups. We consider two groups $G_1$ (additive) and $G_2$ (multiplicative) of the same prime order $q$. $P$ is the generator of $G_1$. A bilinear map is a map $e : G_1 \times G_1 \rightarrow G_2$ satisfying the following properties:

1) Bilinearity: $e(aP,bQ) = e(P,Q)^{ab}, \forall P,Q \in G_1, \forall a,b \in \mathbb{Z}_q$

2) Non-degeneracy: $\exists P,Q \in G_1, e(P,Q) \neq 1.$

3) Computability: $\forall P,Q \in G_1, e(P,Q)$ can be computed efficiently.

B. Complexity assumptions

Let $e : G_1 \times G_1 \rightarrow G_2$ be an admissible bilinear map. Let $P$ be a generator of $G_1$, whose order is a large prime $q$.

**Bilinear Diffie-Hellman Problem:** Let $a$ and $b$ be elements of $\mathbb{Z}_q^*$. Given $<P, aP, bP, eP>$ with uniformly random choices of $a, b \in \mathbb{Z}_q^*$, compute $e(P,eP)^{abc} \in G_2$.

An algorithm $A$ has advantage $\epsilon$ in solving the $\text{BDH}$ problem in $<G_1, G_2, e>$ if

$\Pr[A(<P, aP, bP, eP>) = <Q, e(P,eP)^{abc} >] = \epsilon$

**Generalized Bilinear Diffie-Hellman Problem:** Given $<P, aP, bP, cP>$ with uniformly random choices of $a, b, c \in \mathbb{Z}_q^*$, output a pair $<Q \in G_1', e(P,Q)^{abc} \in G_2>'$.

An algorithm $A$ has advantage $\epsilon$ in solving the $\text{BDH}$ problem in $<G_1', G_2, e'>$ if

$\Pr[A(<P, aP, bP, cP>) = <Q, e(P,Q)^{abc} >] = \epsilon$

4. **Bilinear Diffie-Hellman Problem:** Given $<P, aP, bP, cP, dP>$ with uniformly random choices of $a, b, c, d \in \mathbb{Z}_q^*$, output $D = e(P,P)^{abcd} \in G_2$.
An algorithm $\mathcal{A}$ has advantage $\epsilon$ in solving the 4-GBDH problem in $\langle G_1, G_2, e \rangle$ if
\[
Pr[\mathcal{A}(<P,aP,bP,cP,abP,acP,bcP,dP>) = \langle Q, e(P,P)^{abcd} \rangle] = \epsilon
\]
Here the probability is measured over the random choices of $a,b,c \in Z_q^*$.

BDH Parameter Generator: A randomized algorithm $I\!G$ is a BDH parameter generator if $I\!G(1)$ takes security parameter $k \geq 1$, (2) runs in polynomial time in $k$, and (3) outputs the description of groups $G_1$, $G_2$ of prime order $q$ and a pairing $e : G_1 \times G_2 \rightarrow G_3$. Formally, the output of the algorithm $I\!G(1)$ is $\langle G_1, G_2, e \rangle$.

C. Binding technique

In the original construction of CL-PKE in [15], it is assumed that the KGC is trusted and issues only one copy of each partial key to each entity. Additionally, the partial key $D_A$ should be transmitted to entity A confidentially and authentically. An alternative technique called "Binding technique" is discussed in [26]. The main idea is that the entity A must first fix its secret value and its public key $P_A$. Then the KGC binds entity A's public key with entity A's identity $ID_A$ to generate A's partial key $D_A$. A very important benefit is, with the binding technique, the partial key needs not to be kept secret.

D. Framework of certificateless encryption scheme

- **Setup.** This algorithm is run by the KGC that accepts as input a security parameter $s$ to generate a master-key and a list of system parameters $\text{params}$.
- **SetSecretKey.** The secret value setup algorithm $\text{params}$, is a probabilistic algorithm that takes as input a parameter list $\text{params}$ and a user identity $id$. It returns the user id’s secret value $x_{id}$.
- **SetPublicKey.** The public key generation algorithm, is a deterministic algorithm that takes as input a parameter list $\text{params}$, a user identity $id$, and the user id’s secret value $x_{id}$. It returns the user id’s public key $O_{id}$.
- **PartialPrivateKeyExtract.** The partial private key issuance algorithm, is a deterministic algorithm that takes as input a user identity id, a parameter list $\text{params}$, a master key $s$, and public key $O_{id}$. It returns the user’s partial private key $D_{id}$.
- **SetPrivateKey.** The private key generation algorithm, is a deterministic algorithm that takes as input a parameter list $\text{params}$, the user id’s partial private key $D_{id}$, and the user id’s secret value $x_{id}$. It returns the user id’s private key $S_{id}$.
- **Encrypt.** The encryption algorithm, is a probabilistic algorithm that takes as input a message $M$, a user identity id, a parameter list $\text{params}$, and the user’s public key $O_{id}$. Encrypt returns a ciphertext $C$.
- **Decrypt.** The decryption algorithm, is a deterministic algorithm that takes as input a parameter list $\text{params}$, the decryption key $S_{id}$, and a ciphertext $C$. Decrypt returns a message $M$ or the special symbol $\bot$.

E. Security model

Here we provide a list of the actions that a general adversary against a CL-PKE scheme may carry out and a discussion of how each action should be handled by the challenger for that adversary.

- Extract partial private key of user A: Challenger C responds by running algorithm PartialPrivateKeyExtract to generate the partial private key $D_{id}$ for entity A.
- Extract private key for A: Adversary $\mathcal{A}$ is allowed to make requests for entities’ private keys. If A’s public key has not been replaced then C can respond by running algorithm SetPrivateKey to generate the private key $S_{id}$ for entity A.
- Secret Value Queries: Adversary $\mathcal{A}$ can request the secret value of a user whose identity is $ID$. In response, C outputs the secret value $x_{id}$ for identity $ID$ (It outputs $\bot$, if the user’s public key has been replaced).
- Request public key of A: Naturally, we assume that public keys are available to adversary $\mathcal{A}$. On receiving a first request for A’s public key, C responds by running algorithm SetPublicKy to generate the public key $O_{id}$ for entity A.
- Replace public key of A: Adversary $\mathcal{A}$ can repeatedly replace the public key $O_{id}$ for any entity A with any value $O'_{id}$ of its choice. Note that it is unreasonable for $\mathcal{A}$ to issue partial key query, complete decryption key query, corrupted decryption servers’ key share query, or decryption query, on an entity A whose public key has been replaced.

Hence, there are two types of adversaries namely Type I adversary and Type II adversary with different capabilities in CL-PKC.

A Type I adversary does not access the KGC’s master key but can replace public keys of arbitrary identities with other public keys of her choosing. Such an adversarial behavior seems natural as, in the absence of digital certificates, anyone can alter public directories by replacing public keys without being caught or detected. As attackers against IBE schemes, Type I adversaries can also obtain partial and full private keys of arbitrary identities.

A Type II adversary can get the KGC’s master key and may still obtain full private keys for arbitrary identities but is disallowed to replace public keys during the game.

Chosen ciphertext security for CL-PKE: We say that a CL-PKE scheme is semantically secure against an adaptive chosen ciphertext attack ("IND-CCA secure") if no polynomially bounded adversary $\mathcal{A}$ of Type I or Type II has a non-negligible advantage against the challenger in the following game:

- **Setup:** The challenger takes a security parameter $k$ and runs the Setup algorithm. It gives $\mathcal{A}$ the resulting system parameters $\text{params}$. If $\mathcal{A}$ is of Type I, then the challenger keeps master-key to itself, otherwise, it gives master-key to $\mathcal{A}$.
**Phase 1:** A issues a sequence of requests, each request being either a partial private key extraction, a private key extraction, a request for a public key, a replace public key command or a decryption query for a particular entity. These queries may be asked adaptively, but are subject to the rules on adversary behaviors defined above.

**Challenge Phase:** Once A decides that Phase 1 is over it outputs the challenge identity $ID^*$ and two equal length plaintexts $M_0, M_1 \in M$. Again, the adversarial constraints given above apply. In particular, $ID^*$ cannot be an identity for which the private key has been extracted. Moreover, if A is of Type I, then $ID^*$ cannot be an identity for which both the public key has been replaced and the partial private key extracted. The challenger now picks a random bit $b \in \{0, 1\}$ and computes $C^*$, the encryption of $M_b$ under the current public key $P^*$ for $ID^*$. If the output of the encryption is $\perp$, then A has immediately lost the game (it has replaced a public key with one not having the correct form). Otherwise, $C^*$ is delivered to A.

**Phase 2:** A issues a second sequence of requests as in Phase 1, again subject to the rules on adversary behaviors above. In particular, no private key extraction on $ID^*$ is allowed, and, if A is of Type I, then the partial private key for $ID^*$ cannot be extracted. The challenger now picks a random $sPP \in \{0, 1\}$, and computes $C^*_{PP}$, the encryption of $M_{PP}$ under the current public key $P^*$ for $ID^*$. Once $A$ knows $C^*$, it can make at most $2^{k/2}$ queries $I$ and public key $O_*$, the sender first checks whether $e(X_A, P_0) = e(Y_A, P)$. If not, it aborts. Else, it chooses $r \in Z_q^*$ uniformly at random, and subsequently computes $Q_1 = H_1(ID_A, O_*)$. Then the sender computes $U = r P^*; V = M \oplus H_2(e(Y_A, Q_1))$, $W = r H_3(U, V)$, and outputs a ciphertext $C = \langle U, V, W, >$. 

**Decrypt:** Given a ciphertext $C = \langle U, V, W, >$, receiver, peer A, first checks whether $e(P, W) = e(U, H_3(U, V))$. The peer aborts if the check fails. Otherwise, the peer computes $M = V \oplus H_3(e(S_0, U))$.

**Sign:** For any message $M$ to be signed, the peer computes: $\sigma = S_A H_1(M)$.

**Verify:** Given the message $M$, the peer's identity information $ID_A$ and public key $O_*$, and the signature $\sigma$. The verifier first checks whether $e(X_A, P_0) = e(Y_A, P)$. If not, it aborts. Else, the verifier verifies the equality: $e(P, \sigma) = e(Y_A, Q_1 H_1(M))$, where $Q_1 = H_1(ID_A, P_0)$. If the equality holds, he accepts. Otherwise, outputs failure.

### B. Security Proof

**Theorem 1:** Our CL-PKC scheme is secure against a type I adversary in the random oracle model assuming the GBDH and 4-BDH problem is intractable.

**Proof:** Let $C$ be a GBDH and 4-BDH attacker who receives a random instance $(P, aP, bP, cP)$ of the GBDH problem in $G_1$ and a random instance $(P, aP, bP, cP, abP, acP, bcP, dP)$ of the 4-BDH problem in $G_1$. $A$ is a type I adversary who interacts with $C$ as modeled in Game. We show how $C$ can use $A$ to solve the GBDH and 4-BDH problem. $C$ sets $P_0 = sP$, selects params $= (G_1, G_2, e, P, P_0, H_1, H_2, H_3)$ and sends params to $A$. $A$ computes hash functions $H_1, H_2, H_3$ as random oracles.

**H1 Queries:** Suppose $A_1$ can make at most $q_{H1}$ times $H_1$ queries, $C$ chooses $I \in [1, q_{H1}]$. $C$ maintains an initially empty list $List_{H1}$ of tuples $(ID_0, O_0, a_0, Q_0)$. The same answer from the list $H_1$ will be given if the request has been asked before. On receiving a new query $H_1(ID_0, O_0)$, $C$ simulates the random oracle $H_1$ as follows.

1) If $i = J$, set $Q_i = bP$, add $(ID_0, O_0, a_i, Q_i)$ to $List_{H1}$ and return $Q_i$ as answer.

2) Otherwise, pick $a_i \in Z_q^*$ at random, set $Q_i = a_i P$, add $(ID_0, O_0, a_i, Q_i)$ to $List_{H1}$ and return $Q_i$ as answer.
H2 Queries: C keeps an initially empty list ListH2 of tuples (Ri, ui). Whenever Ai issues a query (Ri) to H2, the same answer from the list ListH2 will be given if the request has been asked before. If the query (Ri) is new, C selects a random ui \in Z_q^* adds (Ri, ui) to ListH2 and returns u_i as answer.

H1 Queries: C keeps an initially empty list ListH1 of tuples (R_i, v_j, a_j, Q_j). Whenever Ai issues a query (Ri, v_j) to H1, the same answer from the list ListH1 will be given if the request has been asked before. For a new query (Ri, v_j), C selects a random a_j \in Z_q, set Q_j = a_jP, adds (R_i, v_j, a_j, Q_j) to ListH1 and returns Q_j as answer.

SetSecretKey Queries: On receiving a query SK (ID), if the public key of ID has been replaced, C returns \bot. Otherwise, if there’s a tuple (ID, x, D_i, O_i) on ListKey, C returns x_i as answer; else, C first selects random x_i \in Z_q, set D_i = O_i \bot then returns x_i as answer and add (ID_i, x_i, D_i, O_i) to ListKey.

SetPublicKey Queries: On receiving a query PK (ID), the current public key from the list ListKey will be given if the request has been asked before. Otherwise, C does as follows.

1) If there’s a tuple (ID_i, x_i, D_i, O_i) on ListKey, (In this case, the public key O_i of ID has been set), compute X_i = x_iP, Y_i = x_iP_0, return O_i = \langle X_i, Y_i \rangle as answer and update (ID_i, x_i, D_i, O_i).
2) Otherwise, choose random x_i \in Z_q, compute X_i = x_iP, Y_i = x_iP_0, return O_i = \langle X_i, Y_i \rangle as answer, set D_i = \bot and add (ID_i, x_i, D_i, O_i) to ListKey.

PartialPrivateKeyExtract Queries: C keeps an initially empty list ListKey of tuples (ID_i, x_i, D_i, O_i). Whenever Ai issues a query PartialKey (ID, D_i, O_i) to ListKey, C will be given if the request has been asked before. Otherwise, C does the following.

1) If ID_i = \bot, abort.
2) Else if there is a tuple (ID_i, x_i, D_i, O_i) on ListKey, a) If there is a tuple (ID_i, O_i, a_j, Q_j) on ListH1, set D_i = a_jP_i and return D_i as answer.
b) Otherwise, first make an H1 query on (ID_i, O_i) to generate (ID_i, O_i, a_j, Q_j), then set D_i = a_jP_i and return D_i as answer.
3) Otherwise, do the following.
a) If there’s a tuple (ID_i, O_i, a_j, Q_j) on ListH1, select random x_i \in Z_q*, computer O_i the same way as SetPublicKey query, compute D_i = a_jP_0, return D_i as answer and add (ID_i, x_i, D_i, P_0) to ListKey.
b) Else, generate the tuple (ID_i, O_i, x_i, Q_i, D_i) the same way as he simulates the random oracle H1. Select random x_i \in Z_q*, compute O_i the same way as SetPublicKey query, compute D_i = a_jP_0, return D_i as answer and add (ID_i, x_i, D_i, O_i) to ListH1.

PublicKeyReplacement Queries: Ai can choose a new public key for the user whose identity is ID_i. On receiving a query PKR (ID_i, O_i), C first finds the tuple (ID_i, x_i, D_i, O_i) on ListKey (if such a tuple does not exist on ListKey or O_i = \bot, C first makes PK (ID_i)), then C updates O_i to O_i'.

Encrypt Queries: On receive a Encrypt query E(M_i, ID_i, O_i), C generates the ciphertext as follows. (Note A_i need not supply the secret value which is used to generate O_i.) It chooses r \in Z_q uniformly at random, and subsequently computes Q_\alpha = H_r(ID_i, O_i). Then the sender computes U = rP; V = M \oplus H_r(Y_\alpha, Y_i) \oplus W = rH_r(U, V) \text{ , and outputs a ciphertext C}=<U, V, W>.

Decrypt: At last, A_i returns a successful decrypt M. Which means A_i can compute M = V \oplus H_r(e(S_i, U)).

If A_i does not replace public key O_i, then M = V \oplus H_r(e(Q_i, P)^\alpha) \text{ , namely, given X_i = x_iP_i, P_0 = sP_i and U = rP_i, A_i can computer e(Q_i, P)^\alpha \text{ , A_i can then successfully obtained the solution of the GBDH problem.}

Else A_i replaces public key with O_i', but cannot execute PartialPrivateKeyExtract Queries. Qi \in G_1, so we can let Q_i = \lambda P_i, then M = V \oplus H_r(e(P_i, P_i)^\alpha) \text{ , namely, given X_i = x_iP_i, P_0 = sP_i, U = rP_i, Y_i = x_iP_i, D_i = s\lambda P_i and Q_i = \lambda P_i, A_i then can successfully obtained the solution of the 4-BDH problem.}

Theorem 2: Our CL-PKC scheme is secure against a type II adversary in the random oracle model assuming the GBDH problem is intractable.

Proof: Let C be a CDH and GBDH attacker who receives a random instance (P, sP, bP) of the CDH problem in G_1 and a random instance (P, aP, bP, cP) of the GBDH problem in G_1. A_{12} is a type II adversary who interacts with C as modeled in Game. We show how C can use A_{12} to solve the CDH and GBDH problem. C sets P_0 = sP, selects params = (G_1, G_2, e, P, P_0, H_1, H_2, H_3). We consider hash functions H_3, H_2, and H_1 as random oracles. When the simulation is started, A_{12} is provided with params and the master-key s.

H1 Queries: Suppose A_{12} can make at most q_{H1} times H1 queries, C chooses J \in [1, q_{H1}]. C maintains an initially empty list ListH1 of tuples (ID_i, O_i, a_j, Q_j). The same answer from the list H1 will be given if the request has been asked before. On receiving a new query H1(ID_i, O_i), C simulates the random oracle H1 as follows.

1) If i = J, set Q_i = bP_i, add (ID_i, O_i, a_j, Q_j) to ListH1 and return Q_i as answer.
2) Otherwise, pick r_i \in Z_q at random, set Q_i = a_jP_i, add (ID_i, O_i, a_j, Q_j) to ListH1 and return Q_i as answer.

H2 Queries: C keeps an initially empty list ListH2 of tuples (R_i, u_i). Whenever A_{12} issues a query (R_i) to H2, the same answer from the list ListH2 will be given if the request has been asked before. If the query (R_i) is new, C selects a random u_i \in Z_q adds (R_i, u_i) to ListH2 and returns u_i as answer.

H3 Queries: C is given an initially empty list ListH3 of tuples (R_i, v_j, a_j, Q_j). Whenever A_{12} issues a query (R_i, v_j) to H3, the same answer from the list ListH3 will be given
if the request has been asked before. For a new query (Ri, v), C selects a random αi ∈ Z∗q, set Qi = αiP, adds (Ri, v, αi, Q) to Listg and returns Qi as answer.

SetSecretKey Queries: On receiving a query SK (IDr), if the public key of IDr has been replaced, C returns ⊥. Otherwise, if there's a tuple (IDr, xo, Do, Oo) on ListKey, C returns xo as answer; else, C first selects random xi ∈ Z∗q, set D0 = Oo = ⊥ then returns xo as answer and add (IDr, xo, Do, Oo) to ListKey.

SetPublicKey Queries: On receiving a query PK(IDc), the current public key from the list ListKey will be given if the request has been asked before. Otherwise, C does as follows.
1) If there’s a tuple (IDc, xc, Dc, Oc) on ListKey (In this case, the public key Oc of IDc has not been set), compute \( X_i = x_iP, \ Y_i = x_iP_0 \), return O′i = \(<X_i, Y_i>\) as answer and update (IDc, xc, Dc, Oc) to \((IDc, x_c, D_c, O_c)\).
2) Otherwise, choose random \( x_i \in Z^*_q \), compute \( X_i = x_iP, \ Y_i = x_iP_0 \), return O′i = \(<X_i, Y_i>\) as answer, set D0 = \( ⊥ \) and add (IDc, xc, Dc, Oc) to ListKey.

PartialPrivateKeyExtract Queries: C keeps an initially empty list ListKey of tuples (IDc, xc, Dc, Oc). When A receives a query PK(IDc), the same answer from the list ListKey will be given if the request has been asked before.
1) If IDc = IDi, abort.
2) Else if there is a tuple (IDc, xc, Dc, Oc) on ListKey
a) If there is a tuple (IDc, xc, Dc, Oc) on ListH1i, set D0 = α0P and return D0 as answer.
b) Otherwise, first make an H1 query on (IDc, Oc) to generate (IDc, Oc, α0, Qo), then set D0 = α0P0 and return D0 as answer.
3) Otherwise, Do the following.
   a) If there’s a tuple (IDc, Oc, α0, Qo) on ListH1i, select random \( x_i \in Z^*_q \), compute Oc by the same way as SetPublicKey, compute D0 = α0P and return D0 as answer.
b) Otherwise, generate the tuple (IDc, Oc, Oc, Qo) on ListH1i, select \( x_i \in Z^*_q \), compute Oc by the same way as SetPublicKey, compute D0 = α0P0, then return D0 as answer and add (IDc, xc, Dc, Oc) to ListKey.

Encrypt Queries: On receive a Encrypt query E(Mi, IDr, Oc), C generates the ciphertext as follows. (Note that \( α_i \) need not supply the secret value which is used to generate \( O_c \).)
It chooses \( r \in Z^*_q \) uniformly at random, and subsequently computes \( Q_i = H_i(ID_r, Oc) \).
Then the sender computes \( U = rP, \ V = M \otimes H_2(e(Y_i, Q_i)) \), \( W = rH_1(U, V) \), and outputs a ciphertext \( C = \langle U, V, W, \rangle \).

Decrpyt: At last, \( A_1; \) returns a successful decrypt M. Which means \( A_1; \) can compute \( M = V \otimes H_3(e(S_i, U)) \), \( Q_i = G_i \), so we can let \( Q_i = AP \); then \( M = V \otimes H_3(e(S_i, P)^{\alpha_i}) \) namely, \( X_i = x_iP \), \( Q_i = AP \) and \( U = rP \), \( A_1; \) can compute \( e(Q_i, P)^{\alpha_i} \).
A then can successfully obtained the solution of the GBDH problem.

V. SECURITY PROTOCOL FOR TRUST DATA SHARING

The following notations are used: THA-peer stands for a peer who holds the trust value for a particular peer. A private key is denoted by the \( S \) and a public key by the \( O \).
The trust value is denoted by the \( TV \). The time stamp is denoted \( TS \). The symbol \( \sqcup \) stands for concatenation.
Encryption of a message \( M \) by a key \( K \) is given by \( ENC_k(M) \), and signature of a message \( M \) by a key \( K \) is given by \( SIG_k(M) \).

A. Overview

Security protocol broadly functions in the following manner. The trust values of a peer (say Peer B) are randomly assigned to another peer (THA peer) in the network. This assignment is done by the KGC in a way that the trust holding responsibilities are equally distributed amongst the participating peers. Any peer (say Peer A) interested in querying for the trust value of Peer B can broadcast a trust query for Peer B. The THA peer replies with the trust value along with some other information. Depending upon the trust value, Peer A can decide to interact with Peer B or not. Also after an interaction, Peer A can securely file a report for Peer B, indicating Peer A's new trust value for Peer B. Then, the THA peer can modify the trust rating of Peer B.

B. Details

1) System setup: Firstly, according to our CL-PKC Scheme, KGC output a master-key \( S \in Z_q^* \) and system's public parameters \( params = \{G_1, G_2, e, q, l, P, P_0, H_1, H_2, H_{SA}, H_{SB}, H_{S}, H_{A}, H_{B}, \} \). The \( params \) is known to all peers in the network. This can be easily achieved by distributing it at the time of a peer join.

2) Peer join: When peer A joins the network, he firstly generates a secret value \( x_A \in Z_q^* \) and public key \( O_A = <x_A, Y_A> \), \( x_A \) is used for providing/receiving services. The KGC then select a set of THA peers \( t_0, t_1, ..., t_N \) for peer A, where \( N \) is the maximum number of THA peers. Each THA peer \( t_i \) will execute followings to generate THA keys:
   - KGC send message \( ID_t \) to peer \( t_i \).
   - The peer \( t_i \) picks a random \( x_i \in Z_q^* \) as its secret value, \( x_i \) is used for providing/receiving services.
   - The peer \( t_i \) outputs THA public key \( THA-O_{t_i} = <x_{t_i}, Y_{t_i}> \), \( x_{t_i} \) is used for providing/receiving services.
   - KGC computers \( D_{t_i} = SQ_{t_i} = SH_i(ID_t, |ID_t, |THA-O_{t_i}|) \). Then \( D_{t_i} \) can be transmitted through public channels.
• The peer $i$ checks the validity of $D_i^A$ as follows. It computes $e(D_i^A, P) = e(H_i(\text{ID}_i) | \text{ID}_A, \text{THA}_O^A)$. The peer aborts if the check fails. The peer computes THA private key $\text{THA}_S_{i} = x_i^A D_i^A$.

The pair of THA keys $\{\text{THA}_S^i, \text{THA}_O^i\}$ is used as an authentication mechanism and as secure transmission mechanism

3) Query trust value: Any peer, say Peer B, intending to query for the trust value of Peer A can just broadcast the trust query message containing $\text{ID}_A$. Typically, a peer will simultaneously query for a number of peers. In such a case the trust query would be the concatenated IDs of all such peers.

\[ \text{Query} = \text{ID}_A | \text{TS} | \text{ID}_1 | \ldots | \text{ID}_n \]

4) Reply: On receiving a trust query for Peer A, its THA peer, say Peer $i$, can generate a reply and send it back to the querying Peer B. The goal of this message is to ensure that the querying peer can identify it to be generated by a THA peer and that it has not been tampered with enroute. The reply message looks like:

\[ R = \text{ENC}_{\text{OA}_i}(M) | \text{SIG}_{\text{THA}_S}^i(M) \]

where $M = \text{ID}_A | \text{ID}_B | \text{TVB} | \text{ID}_A | \text{TVB}$. Any message older than a user-decided window is disregarded. The latter scenario can be also easily prevented by the Certificateless Public Key Cryptography scheme. The malicious peers can generate a valid THA public key. But he can not generate THA private key without the KGC.

5) Collecting Proof-of-Interaction: Whenever two peers (Peer A and B) interact, they exchange proof of interaction with each other, i.e. Peer A gets $\text{proof}_B = \text{ENC}_{\text{OA}_B}(\text{ID}_B | \text{TVB} | \text{TS})$ from Peer B and Peer B gets $\text{proof}_A = \text{ENC}_{\text{OA}_A}(\text{ID}_A | \text{TVB} | \text{TS})$ from Peer A. This value is used as a proof of an interaction. TS is used to prevent replaying of such a message. The use of $\text{ID}_B$ and $\text{TVB}$ is for added protection against somebody using a message from Peer B’ interaction with some other peer. Another important use of the interaction message is that if a group of co-operating peers are attempting to boost each other’s rating, they will need to exchange such messages every time (unless they compromise on each others private keys as well), thus making them pay for every malicious attempt.

6) File report: After having interacted with Peer B, Peer A can file a report indicating Peer B’s new trust value, $\text{TVB}$ for Peer B, to THA peers of peer B. For this, we need to ensure that only the THA peers can read the message and that only a peer which has actually interacted with Peer B can send the report. The Report message is of the form:

\[ \text{ENC}_{\text{THA} \_ \text{TS} \_ \text{оф}}(\text{M}) | \text{SIG}_{\text{THA} \_ \text{OA}}(\text{M}) \]

where $M = \text{ID}_A | \text{TVB} | \text{ENC}_{\text{OA}}(\text{TS} | \text{proof}_B)$.

VI. SECURITY DISCUSSION OF PROTOCOL

Now let us look at various possible attack scenarios and how our protocol prevents it:

A. Manipulating Reply Messages

This can be attempted by either a malicious THA peer or a non-THA peer.

THA peer: A THA peer can send a wrong trust value in the reply. To prevent this, the KGC assigns a number of THA peers for a single peer. Then the querying peer can take a majority vote amongst them and select that value. Since the assignment of THA peers is random, there is extremely small possibility of majority of peers being malicious and co-operative, thus making the above mechanism secure.

Non-THA peer: It can either attempt to replay a genuine reply message at a later time or try and use a fake pair of $\langle \text{THA}_S, \text{THA}_O \rangle$ THA keys. The former scenario is prevented by including TS in the reply message. Any message older than a user-decided window is disregarded. The latter scenario can be also easily prevented by the Certificateless Public Key Cryptography scheme. The malicious peers can generate a valid THA public key. But he can not generate THA private key without the KGC.

B. Manipulating Proof-of-Interaction Message

A proof-of-interaction message can be manipulated either by attempting to replay an old message or by using a fake pair of $\langle S_k, O_i \rangle$ keys. The replay is avoided by the use of TS. Any report message, containing the proof-of-interaction message outside a reasonable time frame, is discarded. The other possibility is that a peer uses a fake pair of $\langle S_k, O_i \rangle$ keys. Note that it is not possible for an offering peer to fake, since its THA peer would have included its true $O_i$ value in the reply message. To prevent the querying peer from faking, the offering peer can also get its actual public key from its THA peer.

C. Manipulating Report Message

There is very little that can be done to manipulate a report message. As mentioned earlier, no peer can fake a report message and also that it is secure in the sense that only the THA peer can read the message. Only possibility is that a peer can rate another peer in a wrong manner, e.g., giving a poor rating in despite of good performance. This is tackled on the trust model level. Any good trust model will not affect a peer’s trust rating because of a single peer.

VII. CONCLUSION

Trust data sharing is the important part of trust model. In this paper we have described the design of a secure underlying protocol for trust data sharing. Based on the Certificateless Public Key Cryptography Scheme, the protocol provides secure, reliable, and accountable
distribution and access of ratings of peers. We have also presented a thorough security analysis of the protocol and reported some initial experimental results, showing that the protocol has desirable features of reliability, accountability and is secure in the presence of a variety of possible attacks.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China (No. 60573120), and the National High-Tech Research and Development Program of China (863 Program) under Grant No. 2007AA01Z420.

REFERENCES


Huaqing Lin received the B.S. degree in Military communications from Naval University of Engineering, china, in 2003. He is currently lecturer of Naval University of Engineering. Since 2005, he was working toward his PhD degree and currently he is a PhD candidate in College of Computer Science & Technology at Huazhong University of Science & Technology, china. His research interests are P2P network, network security.

Yonghong Zhou received the M.Sc. degree in Power Electronics and Power Drives from Naval University of Engineering, china, in 2006. Her research interests are signal detection and electromagnetic field.
Mobile Platform and Secure Access Approach of UMTS Terminal Based on Trusted Computing

Yingyou Wen
College of Information Science and Engineering, Northeastern University, Shenyang, China
Email: wenyy@neusoft.com

Xuena Peng, Shuyi Chen, Hong Zhao
College of Information Science and Engineering, Northeastern University, Shenyang, China
Email: {pengxn, chensy, zhaoh}@neusoft.com

Abstract—On the basis of comprehensive study of trusted computing technology and the threats to 3G network, the architecture of trusted mobile platform based on mobile trusted module is proposed, and a DAA-based key management and a trusted computing based access mechanism for 3G network are designed. Furthermore, a predicate logic based formal analysis method is also proposed. With DAA mechanism, user privacy is protected and the bottleneck problem of centralized CA is avoided. The proposed MTM-based trusted access mechanism focuses on not only the authentication of mobile user, but also the healthy status of the mobile user. Therefore, the UMTS network can forbid the unsafe mobile terminals from accessing it, which guarantees the security of the network from the source. The validity of platform and trusted UMTS access scheme is verified with the proposed formal analysis method.

Index Terms—trusted computing; TNC; MTM; UMTS; direct anonymous attestation

I. INTRODUCTION

The security of 3G system has become a key problem during its developing and become one of the most important research areas. Nowadays, the researches of 3G security mainly focus on two aspects. One is the adoption of traditional security products into 3G system, so that the core network could be protected [1, 2]. The other is the identity authentication and the key exchange in Radio Access Network (RAN) [3, 4]. However, these researches can not solve the 3G security problem thoroughly. On one hand, traditional network security products can only protect core network from attacks which launched from the Internet, but not that from smart mobile terminals. On the other hand, identity authentication and key exchange based access control only focuses on the verification of SIM card, but without any consideration of the healthy status of mobile devices. It is often the unsafe mobile terminals which bring security threats to the whole network, so the authentication based access control could not be the silver bullet either. All these vulnerabilities will bring more threats to whole network operation and those business applications provided on top layer.

Trusted Computing considers credibility as an evaluable and verifiable performance metric, which can be used to solve the information security problem from a new angle [5]. Its main purpose is to enhance the security of the terminal architecture at source, by which terminal devices can acquire immunity to malicious code such as viruses, Trojans and other attacks, thus provide a reliable and credible network environment [6]. As 3G network has a rigorous identity management, authentication and billing mechanism, and easy to execute management of mobile devices, Therefore, this paper proposed an architecture of trusted mobile platform based on Mobile Trusted Module, and a Trusted Network Connect (TNC) scheme is given, by which the mobile terminals’ authenticity and integrity will be detected and unsafe terminal access will be automatic refused.

II. TRUSTED COMPUTING AND MOBILE TRUSTED MODULE

A. Trusted Computing

Anderson J.P. first proposed the concept of the credibility of the system so far trusted computing

Corresponding author: Yingyou Wen
Email: wenyy@neusoft.com

Figure 1 Architecture of trusted network connection.
research has gone through the course of more than 30 years [7, 8]. TCG (Trusted Computing Group) as an important representative body, has set up various working groups responsible for developing and publishing trusted computing architecture, trusted computing modules, Trusted Network Connect and credibility of the mobile platform specifications [9]. A typical application of utilizing trusted computing technology to protect network is Trusted Network Connect (TNC) [11]. Its main purpose is to provide network access control through the use of trusted computing technology. TNC architecture based on the TPM (Trusted Platform Module) is shown in Fig.1.

TNC is divided into three tiers, namely client-side AR, PDP strategic services and PEP access control. It also defines interface specification between different entities. Clients with the TPM provide hardware-based host authentication and host integrity verification. TNC allows only trusted terminal to access the network through the platform authentication, access restrictions, evaluation, isolation and remediation technology, and then prevent unsafe endpoints linked to the network and protect the entire network security.

B. Mobile Trusted Module

The wide use and increasing capabilities of mobile devices introduce security risks to the mobile phone users as well as mobile operators. Applying TPM directly to the mobile platform facing some problems.

First, mobile platform is subject to the limited computing capability, storage and energy capacity of mobile devices, so the consideration in designment of both security chip and security policy will be different with those in PC.

Secondly, the application environment of mobile platform is very complex, users, operators, content providers and device manufacturers have different requirements, all of which should be considered during the designment of trusted mobile platform. The Trusted Computing Group (TCG) specification [12] aims to address this problem and Mobile Phone Work Group (MPWG) in TCG specially deals with trusted computing in mobile environment [13].

MPWG has described a set of selected mobile phone uses cases, and announced Mobile Trusted Module specification (MTM), which provides protection for mobile terminals instead of TPM. Major components of a trusted mobile platform based on MTM include:

a) Reference Integrity Metric (RIM). RIM is a value used to validate the result of a measurement taken before software or hardware is loaded or initialized (for execution). Typically a digest of compiled software and configuration data which can affect the engine trust state.

b) Engine. Engine is a dedicated processor or runtime environment which is able to access trusted resources for the implementation of reliable service or ordinary service. TIM (Target Integrity Metric) is Integrity Metric of a target object or component as measured by the measurement agent of that object. Typically a digest of a software image and/or configuration data. trusted services execute reliable measurement and reliable verification, verify whether the integrity is damaged by comparison of TIM and RIM.

c) RIM authorization (RIM Auth). It is the entity that issued the certificate RIM_Certs.

d) MTM include trusted storage root, Root of Trust for Measurement and Root-of-Trust-for-Reporting, verify the integrity of RIM and RIM Auth and record the RTM/RTV measurement results.

III. SAFETY BOOTING OF TRUSTED MOBILE PLATFORM BASED ON MTM

A. Architecture of Trusted Mobile Platform

Processor of Trusted Mobile Platform include baseband processor and application processor, the two processors can be separated. We adopt dual-processor architecture in our solution. According to MTM specification of TCG and referring to the architecture of the mobile platform proposed by MPWG, revised trusted mobile platform architecture is proposed as shown in Fig.2.

The whole system consists of the following security components and constitutes a credible border.

a) Application processor: Application processor control system execution at boot time and later, isolates the trusted applications and their data with entrusted application.

b) MTM: MTM provides a protected execution environment with a variety of security features, including key algorithms, authentication, and trusted storage. MTM can provide more flexible use of the security and services to meet multi-owner requirements of the mobile platform. Equipment suppliers can ensure that the mobile platform is in a trusted state through the built-in MRTM, and provide trust assurance for the upper applications. The upper applications, such as mobile operators, application achieve reliable service through additional MRTM and MLTM.

c) RTV/RTM: RTM (Root of Trusted for Measurement) measure the integrity of subsequent components. TV is responsible for
the comparison of on-site measurement and reference value in RIM to determine whether the integrity of the measurement object is damaged, MTM visit RTV / RTM through the DMA.
d) DMA: In system with a High security level, DMA is controlled by a trusted core to ensure trusted application or operating system to access physical memory.

B. Safety Booting of Trusted Mobile Platform

Safety booting process is a basis to ensure that mobile platform is trusted. After safety booting, a trusted program execution environment can be established, ensure that the system in a trusted state both at startup and runtime, and provide protection for program. Safety booting process is transfer of trusted measurement and trusted verification along the chain of trust, also a delivery process of execution control.

RTM measure the integrity value of follow-up components in trust chain, RTV is responsible for the comparison of on-site measurement and reference value in RIM to determine whether the integrity of the measurement object is damaged. If verification is passed, it will pass execution control to components in next level. Booting is implemented step by step and the integrity of the entire platform is assured.

Safety booting process in MTM related specification is implemented by using a measurement and verification agent to measure and verify follow-up components. After completion of the measurement and verification, control of measurement and verification is passed to the measurement and verification agency in the next component. However, as the agency is not entirely credible, so that trust measurement and certification approach will bring loss of trust, loss of trust will be enlarged with the increase of trust chain length and leading to degrade of credibility of whole platform.

In this paper, a trusted safety booting process based on direct measurement is proposed, in which all the measurement and verification of credibility is done by RTM and RTV. None agent is involved in the process to avoid loss of trust. This safety booting sequence diagram is shown in Fig 3.

Assuming that a RIM certificate needed in the safety booting process has been generated, a safety booting process of mobile platform based on direct measurement is described using TCG specified original language as follows:

Stage 1: MTM → TPM_Init, TPM_Startup
MTM setup after TPM_Init and TPM_Startup has been called; all values of PCRs are initiated as 0;

Stage 2: MTM → TPM_VerifyRIMCertAndExtend(\(e'1\))
MTM verify integrity of RTM/RTV according to RIM certificate and record verification result in PCR;

Stage 3: RTM → Measure(\(e_2\))
\(RTV → LookUpRIMCert\)
\(MTM → TPM_VerifyRIMCertAndExtend(\(e\))\)

RTM measure the integrity value of \(e_2\), RTV load RIM certificate of \(e_2\), and verify that if the on-site measurement and the reference value in RIM are the same. TM executes TPM_VerifyRIMCertAndExtend to verify RIM certificate, and record result in PCR, and then \(e_2\) gains its execution right.

Stage 4: RTM → Measure(\(e_3\))
\(RTV → LookUpRIMCert\)
\(MTM → TPM_VerifyRIMCertAndExtend(\(e\))\)

RTM measure integrity value of \(e_3\), RTV load RIM certificate of \(e_3\) and verify that if the on-site measurement and the reference value in RIM are the same. MTM execute TPM_VerifyRIMCertAndExtend to verify RIM certificate and record result in PCR, then \(e_3\) gains its execution right.

Stage 5: RTM → Measure(\(e_4\))
\(RTV → LookUpRIMCert\)
\(MTM → TPM_VerifyRIMCertAndExtend(\(e\))\)

RTM measure integrity value of \(e_4\), RTV load RIM certificate of \(e_4\) and verify that if the on-site measurement and the reference value in RIM are the same. MTM execute TPM_VerifyRIMCertAndExtend to verify RIM certificate and record result in PCR, then \(e_4\) gains its execution right.

After the measurement and verification of RTM and RTV, integrity of all entities involved in the booting process are assured so that trusty of mobile platform can be achieved. If any return value of TPM_VerifyRIMCertAndExtend is error or no proper certificate is found, the process will be ceased.

IV. TC-BASED UMTS ACCESS APPROACH

As mentioned above, we introduce trusted computing techniques to 3G network security infrastructure, and design an integrity measurement and authentication-based UMTS access solution, which can secure the 3G network by prohibiting the untrusted terminal from accessing the mobile network. We assume here that the Mobile Trusted Module (MTM) is embedded in the mobile terminals [14]. The function of MTM is to generate and manage the keys,
and protect the information by making use of the hardware functions.

### TABLE I. ILLUMINATION OF IDENTIFIERS

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>AuC</td>
<td>Authentication Center</td>
</tr>
<tr>
<td>K_A</td>
<td>A’s Public key</td>
</tr>
<tr>
<td>K_A'</td>
<td>A’s Private key</td>
</tr>
<tr>
<td>Cert_A</td>
<td>The certificate for A’s public key</td>
</tr>
<tr>
<td>EK</td>
<td>Endorsement Certificate</td>
</tr>
<tr>
<td>AV</td>
<td>Authentication Vector</td>
</tr>
<tr>
<td>RTM</td>
<td>Root of Trusted Measurement</td>
</tr>
<tr>
<td>RTV</td>
<td>Root of Trusted Verification</td>
</tr>
<tr>
<td>IS</td>
<td>DAA Certificate Issuer</td>
</tr>
<tr>
<td>RAND</td>
<td>Random number</td>
</tr>
<tr>
<td>AUTN</td>
<td>Authentication Token</td>
</tr>
<tr>
<td>XRES</td>
<td>Expected Authentication Reply</td>
</tr>
<tr>
<td>Issuer</td>
<td>User authentication reply</td>
</tr>
<tr>
<td>AC</td>
<td>Access Controller</td>
</tr>
</tbody>
</table>

Table 1 shows the variables and symbols used in the context of this paper.

First, a description of key management is given. The pre-set shared key exists between the pre-ME with the HLR in 3G system, which is used to implement authentication and key agreement. TCG organization has proposed a key management scheme using Privacy CA and direct Anonymous Attestation (DAA) [15]. DAA implement the MTM certificate management mechanism using zero-knowledge proof techniques. To be proved as a legitimate MTM, only ownership of the DAA certificate is needed, in stead of certificates itself. Furthermore, if initialization of DAA certificate is issued off-line, it can not only solve the bottleneck problem of CA, but also provide platform with reliable identity protection. In view of this, We propose to issue a one-time DAA certificate for the MTM before mobile devices gaining mobile network license and store public key of MTM endorsement key in the AuC to execute management of DAA certificate such as revoking and updating when needed.

In order to facilitate the description, DAA relevant knowledge is still called as public key and private key, DAA certification process is as follows:

- **ISSUER(AuC)→MTM:** $K_{IS}^{+}, K_{IS}^{-}$

  DAA signer generates key pair, and release the public key:

  \[
  K_{IS} = \begin{pmatrix} n, g, \cdot, g, h, S, Z, R_0, R_1, y, r_1, r_2 \end{pmatrix} \quad (1)
  \]

- **MTM→ISSUER(AuC):** $K_{EK}^{+}, U, N_I$

  After MTM receive the $K_{IS}$, it generates a piece of secrete information $f$ based on $K_{EK}$ and a random number $v'$, then separate $f$ to $(f_0, f_1)$, and compute $U, N_I$, finally, MTM sends $K_{EK}^{+}, U$ and $N_I$ to DAA signer.

  \[
  U = R_i^{f_0} + R_{i'}^{f_1} \mod n \quad (2)
  \]

  \[
  N_I = e^{v'}_1 + e^{v'} \mod \Gamma \quad (3)
  \]

- **ISSUER(AuC)→MTM:** $A, e, v'$

  DAA signer validates the authenticity of $K_{EK}^{+}$, choose random numbers $e$ and $v'$, compute $A$ and send it to MTM

  \[
  A = \begin{bmatrix} Z \\ US \end{bmatrix}^{f_0/v_'} \mod n \quad (4)
  \]

  After MTM verify whether $A$ is correct, then save $(f_0, f_1, v' = v' + v')$ as knowledge, which can be used to generate DAA signature in each identity authentication process.

Fig.4 shows the trusted computing based UTMS access solution. First, the mobile platform performs secure boots, and then whenever the mobile terminal access the network, the platform will not only authenticate the mobile terminal, but also verify the configuration status integrity of the mobile terminal. Integrity indicates whether the mobile terminal is healthy or configured as expected. Authenticity assures that the system can only be used by authorized users.

Figure 4 Trusted access mechanisms of UMTS

Some key processes of access are given below:

a) $RTM/RTV → ME/MTM$: $TPM\_VerifyRIMCertAndExtend$

  After the mobile terminal set up, it first executes the trusted boot procedure. RTM measures the configure component integrity of the platform, RTV then compare the result with the baseline value stored in RIM certificate, in order to validate whether the object is attacked or destroyed. Then RTV report the measurement result to MTM by invoking $TPM\_VerifyRIMCertAndExtend$ function. MTM stores the result in PCRs after verification. The integrity of the platform can be assured by multi-stage measurement, verification, and transmission of control.

b) $ME/MTM → VLR/SGSN$: IMSI, HLR, $Cert_{DAA}^{+}$

  When MS access the network, it first send its IMSI, HLR and $Cert_{DAA}^{+}$ to VLR/SGSN, and request to register to the network.

c) $VLR/SGSN → HLR/AuC$: IMSI, $Cert_{DAA}^{+}$

  After VLR received the registration requirement from MS, it sends IMSI to the HLR which the MS belongs to, and requests the HLR to authenticate the specific user. After that, it sends $Cert_{DAA}^{+}$ to VLR and request the corresponding DAA to verify the key $K_{IS}$. 

d) $HLR/AuC → VLR/SGSN$: $RAND||XRES||CK||IK||AUTN, K_{IS}^{+}$
After HLR received the authentication request from VLR, it generates an authentication vector (RAND \{XRES\} \{CK\} \{IK\} \{AUTN\}), and releases the key \(K_{IS+}\) for verifying the DAA certificate.

e) VLR/SGSN \rightarrow ME/MTM: RAND \{AUTN, R_{PCRs}\}

After VLR receives the authentication vector, it sends RAND and AUTN to ME/MTM, and requests the user to generate authentication data. Then, it sends platform integrity request \(R_{PCRs}\).

After ME received RAND and AUTN, it computes the value of XMAC (Expected Message Authentication Code). Then it compares the result with the MAC value in AUTN. If they are not identical, ME will send refuse message, and stop the procedure. If they are identical, ME verify if SQN is within the predefined range, if it is out of the range, meaning the synchronization fails, then it will send synchronization failure message, and stop the procedure. If both above verifications succeed, ME believes VLR to be the authorized network of HLR, and then it computes CK, IK and RES.

f) ME/MTM \rightarrow IMV/AC: RES, \{K_{AIK-}, Verifier.t\} K_{DAA-}, \{PCRs\} K_{AIK-}, K_{AIK-}, SML

EM/MTM sends RES to IMV/AC to testify the authenticity of the user. It then uses \((f_o, f_i, v)\) to generate the DAA signature key \(K_{DAA-}\), and also the identification key pair \(K_{AIK-}\) and \(K_{AIK-}\). After that, it sends IMV/AC the \(K_{DAA-}\) signed \(K_{AIK-}\) information \([K_{AIK-}, Verifier.t, K_{DAA-}, K_{AIK-}, K_{AIK-}, SML]\), in order to verify the integrity and authenticity of the platform.

After IMV/AC received the message from ME/MTM, it compares RES and XRES. If they are identical, it believes ME to be a legal user.

IMV/AC validates the DAA certificate with \(K_{IS+}\). If the certificate is legal, it believes the \(K_{DAA-}\) signed \(K_{AIK-}\) information to be authentic, thus the platform is also authenticate. Then it validates \(K_{AIK-}\) signed PCRs with \(K_{AIK-}\). If the signature is authenticated, then it believes \(K_{AIK-}\) signed PCRs is authenticated. Furthermore, it Hashes the SML, and compares the result with the value stored in PCR to verify the integrity of the platform. Based on the verification results, and according to the access control policy, IMV/AC then assign corresponding resource access privilege to user, or isolate it and repair the platform configuration, and forbidden it from accessing the network until the platform configuration is satisfied.

When all the above procedure is executed, the UMTS access solution mainly achieves the following tasks: a) the user authenticates the network, and the user assures that VLR is a HLR authorized network; b) the network authenticates the user, and the network assures that the user is a legal user; c) the key exchange of encryption key (CK) and integrity key (IK); d) MTM-based mobile terminal integrity verification.

V. VALIDATION AND ANALYSIS

In order to validate our solution, a predicate logic-based trusted computing model is proposed as the method for solution validation and analysis. Based on trusted computing specifications, we have defined a variety of predicates to indicate the characteristics of objects and the relationships among these objects, as well as the inference rules for deducing the trust relationship.

A. Formalized validation of the trusted access solution

Definition 1. Given E the set of all entities in the trusted computing system, for \(\forall e_i, e_j \in E\), and \(i \in N\), the following predicates are defined:

- **Trusted** \((e_i)\), which means \(e_i\) is trusted.
- **Measure** \((e_i, e_j, \text{Integ})\), which means \(e_i\) measures the integrity of \(e_j\) and trusts \(e_j\).
- **Trusted** \((e_i) \wedge \text{Measure}(e_i, e_j, \text{Integ})\) is the rule of trusted domain extending. Initial trusted domain only includes \(e_i\). After \(e_i\) measures the integrity of \(e_j\), trusted domain was extended and includes both \(e_i\) and \(e_j\).

During the booting process we proposed above, trusted chain includes only \(e_i\) after platform is setup, namely trusted root RTM/RTV is credible, so initial state is **Trusted** \((e_i)\)

Our conclusion is that all of four entities in trusted chain are trusted after safety booting. Namely we have:

\[
E = \{e_i | \text{Trusted} (e_i) | 1 \leq i \leq 4\}
\]

According to inducing rule in [10], we have:

- a) **Trusted** \((e_i)\)
- b) **Trusted** \((e_i) \wedge \text{Measure}(e_i, e_j, \text{Integ})\)
- c) **Trusted** \((e_i) \wedge \text{trusted}(e_j) \wedge \text{trusted}(e_k)\)
- d) **Trusted** \((e_i) \wedge \text{trusted}(e_j) \wedge \text{trusted}(e_k) \wedge \text{trusted}(e_l)\)

\[
\exists E = \{e_i | \text{Trusted} (e_i) | 1 \leq i \leq 4\}
\]

By the above equations, based on initial conditions, using deduction rules, we can draw a conclusion that all components in platform are trusted.

B. Formalized validation of the trusted access solution

Definition 2. Given E the set of all entities in the trusted computing system, for \(\forall e_i, e_j \in E\), and \(i \in N\), the following predicates are defined:

- **Trusted** integrity measurement capability: **Trust** \((e_i, e_j, \text{Integ})\), which means \(e_i\) trusts that \(e_j\) has the capability of measuring the integrity.
- **Trusted** certification issuing capability: **Trust** \((e_i, e_j, \text{Cert})\), which means \(e_i\) trusts that \(e_j\) has the capability of issuing and maintaining a certification.
• Trusted integrity: \( \text{Trusted} \ (e_1, e_2, \text{Integ}) \), which means \( e_1 \) believes that \( e_2 \) has the attributes of trusted integrity.
• Trusted authenticity: \( \text{Trusted} \ (e_1, e_2, \text{Auth}) \), which means \( e_1 \) believes that \( e_2 \) has the attributes of trusted authenticity.
• Certificate Issue: \( \text{Cert} (e_1, e_2) \), which means after \( e_1 \) verifies the authenticity of \( e_2 \), it signs a certificate to \( e_2 \).
• Integrity Measurement: \( \text{Meas} (e_1, e_2, \text{Integ}) \), which means \( e_1 \) measure the integrity of \( e_2 \), if the measurement result of \( e_2 \) is as expected, \( e_1 \) is integrates.

**Definition 3.** Given \( E \) the set of all entities in the trusted computing system, for \( \forall e, e_i \in E \) and \( 1 \in N \), the following rules are defined:

1. **Rule 1.** Integrity verification rule

\[
\text{Trust} (e_1, e_2, \text{Integ}) \Rightarrow \text{Meas} (e_1, e_2, \text{Integ})
\]  

**Rule 2.** Authentication verification rule

\[
\text{Trust} (e_1, e_2, \text{Cert}) \Rightarrow \text{Cert} (e_2, e_1)
\]

**Rule 1** means \( e_1 \) believes \( e_2 \) has the capability of trusted integrity measurement, \( e_1 \) measures the integrity of \( e_2 \), verifies that \( e_2 \) has trusted integrity, then \( e_1 \) believes that \( e_2 \) has the attributes of trusted integrity.

**Rule 2** means \( e_1 \) believes \( e_2 \) is the legal certificate signer, \( e_1 \) signs a certificate for \( e_2 \), then \( e_1 \) believes the certificate of \( e_2 \) is trusted, and \( e_1 \) has the attributes of trusted authenticity.

The following procedure indicates how the formalized analysis of trusted computing based secure access of UTMS based on predicate logic is carried out. Firstly, present the predicate-based initial conditions and conclusions, clarify the consumptions according to the deducing requirements; secondly, apply the definition of predicates and inference rules, start from the initial conditions and assumptions, and deduce the conclusion; finally, analyze the rationality of the assumptions, and validate the correctness of the trusted computing model. As long as the assumption is rational, then the model is believed to be trusted, otherwise, look for the vulnerabilities of the model according to the irrationality of the assumptions.

Our proposed formalized validation for trusted computing model is as follows:

1. **Initial conditions**

In our proposed trusted computing based secure access solution, we assume the entities that represent the mobile terminal, MTM, visiting region and home region are respectively \( e_1, e_2, e_3 \) and \( e_4 \). At the initial state, \( e_2 \) and \( e_3 \) both trust \( e_1 \) to be the legal DAA certificate signer, so the initial state is:

\[
\text{Trust} (e_2, e_4, \text{Cert}) \land \text{Trust} (e_3, e_1, \text{Cert})
\]

Besides, we assume if \( e_1 \) can validate that \( e_2 \) is the legal MTM, and then \( e_1 \) believe \( e_2 \) has the capability of integrity measurement:

\[
\text{Trusted} (e_1, e_2, \text{Auth}) \Rightarrow \text{Trust} (e_1, e_2, \text{Integ})
\]

We also assume MTM is an indivisible part of the mobile terminal, so it can represent the authenticity of the platform. If \( e_1 \) can validate \( e_2 \) is authentic, the mobile terminal \( e_1 \) which \( e_2 \) embedded in can also be regarded as legal.

\[
\text{Trusted} (e_1, e_2, \text{Auth}) \Rightarrow \text{Trust} (e_1, e_2, \text{Auth})
\]

**Conclusions**

The conclusion is that \( e_3 \) validate \( e_1 \) to be the trusted mobile terminal, which has the correct attributes representing its authenticity and integrity, so the conclusion should be described as:

\[
\text{Trusted} (e_1, e_2, \text{Auth}) \land \text{Trusted} (e_1, e_1, \text{Integ})
\]

**Deducing procedure**

Since the original authentication and key exchange procedure of UTMS access solution is already recognized by the public, we do not further discuss it. In this paper, we only validate the integrity and authenticity of our trusted computing based secure access solution. The deducing process is as follows:

According to the access solution, before mobile terminal access the network, \( e_1 \) has already signed a DAA certificate to \( e_2 \). Based on rule 2 and initial condition (10), the trusted authenticity of \( e_1 \) can be:

\[
\text{Trust} (e_3, e_4, \text{Cert}) \land \text{Cert} (e_4, e_2)
\]

Conclusion (14) shows \( e_2 \) is authentic, and is the trusted MTM. Based on assumption (11) and (12), the following conclusion can be deduced:

\[
\text{Trusted} (e_1, e_2, \text{Auth}) \Rightarrow \text{Trust} (e_1, e_1, \text{Integ})
\]

Conclusion (15) shows that \( e_1 \) believe that \( e_2 \) has the trusted capability of integrity measurement. \( e_2 \) measures the integrity of \( e_1 \), stores the result of the integrity measurement in PCRs, and send it to \( e_3, e_4 \) can get the integrity status of \( e_1 \) based on the integrity results and the logs that \( e_2 \) generated.

\[
\text{Trust} (e_2, e_2, \text{Integ}) \land \text{Meas} (e_2, e_1, \text{Integ})
\]

According to conclusion (16) and (17) above, conclusion (13) can be deduced, which shows that \( e_1 \) believe \( e_2 \) to be the trusted mobile terminal, satisfying the requirement of integrity and authenticity:

\[
\text{Trusted} (e_1, e_1, \text{Auth}) \land \text{Trusted} (e_1, e_1, \text{Integ})
\]

As the assumption is rational, and from the initial condition we can deduce the integrity and authenticity of the platform, so the UTMS access solution is logically correct.

**C. Security and performance analysis**

The trusted UMTS access solution provides not only the authentication between user and network, but also the hardware-based identification and integrity verification of mobile terminals, which helps to secure the network from
With the introduction of MTM, our solution provide the hardware-based high level security assurance: a) secure storage is implemented based on the platform configure register embedded in the MTM hardware; b) trusted measurement and verification is implemented based on the measurement root and verification root embedded in MTM, and the authentication and integrity of the mobile terminal is assured; c) MTM provides an encapsulated and protected runtime environment for key generation, storage, encryption and decryption, signature authentication; d) platform status certification can be provided to the remote verification platform based on the MTM trusted report root.

From the data showed in TABLE II, the solution proposed in the paper is more secure and more adaptable than current solutions.

In this solution, since the DAA certificate is signed and distribute to the mobile terminal on its initial access to the network, the mobile terminal do not need to request the certificate for multiple times from DAA signer, so that the bottle neck at certificate signer is successfully avoided. The verification information is conveyed within the original UMTS authentication and key exchange process, and there’s no extra cost for integrity verification, so the total times of communication are not increased, although the average message length increases a bit, anyway, the change of message conveying method actually decrease the cost of communication.

<table>
<thead>
<tr>
<th>TABLE II</th>
<th>COMPARISON BETWEEN TWO SCHEMES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solution</td>
<td>UMTS Solution</td>
</tr>
<tr>
<td>Authentication (USIM)</td>
<td>√</td>
</tr>
<tr>
<td>Authentication VLR</td>
<td>√</td>
</tr>
<tr>
<td>Ik/Ck exchange</td>
<td></td>
</tr>
<tr>
<td>Terminal anonymity protection</td>
<td>×</td>
</tr>
<tr>
<td>Terminal Authenticity Measurement</td>
<td></td>
</tr>
<tr>
<td>Terminal Integrity Measurement</td>
<td></td>
</tr>
<tr>
<td>Security Level</td>
<td>Middle</td>
</tr>
</tbody>
</table>

VI. CONCLUSION

The architecture of trusted mobile platform based on Mobile Trusted Module is proposed and a trusted computing based UMTS access solution is proposed in this paper. The proposed solution can achieve not only the validity authentication between user and network, but also the authentication and integrity check of the mobile terminal, which can ensure that the mobile terminal will not bring security threats to UMTS network. The formalized analysis method of predicate logic based trusted computing model, which has the advantage of simplicity, efficiency and accuracy, is applied to the validation analysis of trusted UMTS access solution. The formalized analysis method provides us with a trust validation theory according to specific trusted computing application scenario. Our further work is to implement the trusted UMTS access system based on the proposed solution.

ACKNOWLEDGMENT

This work was supported in part by grants from the National Natural Science Foundation of China (Grant No.60803131) and The Science and Technology Plan of Shenyang(No.1091176-1-00).

REFERENCES

[9] TCG. TCPA Main Specification version 1.1b [EB/OL], https://www.trustedcomputinggroup.org/specs/TCPA_Main_TCG_Architecture_v1_1b.pdf, 2008.01
international conference on Trust, Turin, Italy Pages: 18 - 27, 2008


Yingyou Wen was born in Shenyang, China on March 12, 1974. He received the PH.D. degree in computer science from Northeastern University, China, 2005. His main research include: network security, wireless communication and sensor network. He is an associate professor of Northeastern University, and vice president of Neusoft Research, China. His current research focuses on Middleware of WSN and network security.

Xuena Peng was born in 1979, in Shenyang, China. She received her PH.D. degree in computer engineering and science from Northeastern University in China, in 2007. Her main research interest include: Intrusion detection, 3G network security, security alert analysis, P2P streaming, and sensor network. Currently, she is working as a postdoc in Northeastern University, China. Her current research focuses on WSN Middleware and network security.

Shuiyi Chen was born in Henna Province, China in 1972. He received the PH.D. degree in computer science from Northeastern University, China, 2009. His main research include: network security and wireless communication. His current research focuses on security of 3G/4G network.

Hong Zhao was born in Hejian, China in 1954. He received the PH.D. degree in computer science from Northeastern University, China, 1990. His main research include: computer network and information security. He is a professor of Northeastern University, and super vice president of Neusoft Group, China. His current research focuses on network security and wireless communication.
A Network Architecture for Load Balancing of Heterogeneous Wireless Networks

Wenxiao Shi, Bin Li, Na Li, Chuanjun Xia
Jilin University, Changchun, China
Email: swx@jlu.edu.cn

Abstract—The traditional centralized load balancing had a relatively low reliability, and the distributed load balancing had a huge overhead. To solve these problems, this paper mapped heterogeneous wireless networks to distributed grids by introducing Resource Management Unit, and then presented a hierarchical semi-centralized architecture for load balancing of heterogeneous wireless networks drawing on the idea of grid in computer networks. The analytical models for the integrated reliability and signaling overhead of the architecture were established. Theoretical analysis and simulation results indicate that the architecture can reduce the signaling overhead and improve the system reliability effectively.

Index Terms—network architecture, signaling overhead, integrated reliability, load balancing, heterogeneous wireless networks

I. INTRODUCTION

Radio systems are moving toward forming heterogeneous wireless networks: collaborations of multiple radio access networks, which in some cases operate different radio access technologies [1]. The deployment of heterogeneous wireless networks is spreading throughout the world as users want to be connected anytime, anywhere, and anyhow [2]. 3GPP specifies some recommendations for methods, architectures, and design of heterogeneous wireless network interconnection (in particular, the one formed by UMTS and WLAN networks), such as those presented in [3]. The key topics in heterogeneous wireless networks are referred to as spectrum sensing, coexistence, resource management, reliability and QoS support avoiding interference etc. [4].

As a recent research focus, load balancing [5, 6] which belongs to Radio Resource Management (RRM) is one of the key technologies in the convergence of heterogeneous wireless networks. Load balancing is a significant method to achieve the resource sharing over heterogeneous wireless networks, and it can improve resource utilization, enlarge system capacity, as well as provide better services for users.

Generally, load balancing is divided into two parts [7]: network architecture and load balancing algorithm. The former is the foundation of load balancing, and a good network architecture can improve the efficiency of load balancing. In the perspective of control mode, load balancing mechanisms can be classified as centralized, distributed and semi-centralized and semi-distributed [8, 9]. There are some problems in the first two mechanisms: the centralized one has a relatively low reliability, while the distributed one has a huge overhead [10].

A grid [11, 12] is a service for sharing computer power and data storage capacity over the Internet. Grid infrastructure is a large virtual organization that integrates a large mount of distributed heterogeneous resources and high performance computing capabilities into a super service, which can provide huge computing services, storage capability and so on. Grid allows the use of geographically distributed computing systems that belong to multiple organizations as a single system. Resource management and scheduling is the important components of the Grid. It efficiently maps jobs submitted by the user to available resources in grid environment [13]. There are some similarities between grid and heterogeneous wireless networks, such as the dynamic variation of resources, heterogeneous structure, and the key technology of integrating and deploying the distributed resources. Enlightened by the similarities, we propose a hierarchical semi-centralized architecture (HSCA) based on basic grids for load balancing of heterogeneous wireless networks. Since the reliability and overhead are important performances to a network, we analyze the two performances of HSCA we proposed in this paper.

The remainder of this paper is organized as follows. In Section II we briefly discuss the related researches into overhead and reliability of network architectures. Section III introduces the HSCA we proposed, and gives the signaling flows of the HSCA. In Section IV, the modeling of integrated reliability and signaling overhead for HSCA is presented followed in Section V by our simulation and the results. Finally in Section VI we conclude this paper with discussion of our work.

II. RELATED WORK

The authors of [14] developed a mathematical framework that can be used to compactly represent and analyze heterogeneous networks that combine multiple entity and link types. They generalized Bonacich centrality, which measures connectivity between nodes
by the number of paths between them, to heterogeneous
networks and used this measure to study network
structure. The authors of [9] proposed a semi-centralized
and semi-distributed architecture (SCSDA), in which a
BS just exchanges load information with several
neighboring BSs. Although the architecture can reduce
the overhead of control signaling, the authors neither
expressed the overhead in mathematical formula, nor
proved it by simulation.

Reference [15] designed a hybrid wireless network
architecture, [16] proposed a multiple mobile routers
based network architecture to support seamless mobility
across heterogeneous networks, and they both tested the
overhead by NS2 simulator. However, the model of the
overhead was not derived in neither [15] nor [16]. Route
overhead was analyzed in theory in [17], by calculating
the number of control messages generated in a BS/AP
service area due to maintaining route. Nevertheless, the
simulation for overhead was not given. The
communication overhead of the scheme presented in [18]
was calculated, and an algorithm for minimizing the
communication overhead was given, which was proved to
be effective through simulation. The authors of [19]
considered a general heterogeneous network architecture
with two basic entities in the system: mobile nodes
(MNs) and access points (APs). They formulated the
overhead of AP discovery which is divided into hello
messages and RREQ messages, and gave the simulation
results. Reference [20] proposed a hierarchical and
distributed (HD) architecture with three hierarchical
levels of mobility management being distinguished: end
terminal remains connected to the same radio access
network but it changes its point of attachments, end
terminal changes its radio access network but it remains
associated to the same operator and end terminal changes
its operator network. And it also studied signaling cost
generated by QoS negotiation during handover process in
both theory and simulation.

The research on reliability of telecommunication
network starts at the study on switched
telecommunication network by Lee [21]. Lee defined call
blocking as the link failure, and measured reliability
taking connectivity [22] as standard. Reference [23]
mentioned the concept of integrated reliability, which
took call loss as the evaluation indicator of network
reliability, and proved that the integrated reliability can
reflect the practical situation much better than taking
connectivity as standard. The authors of [24] analyzed the
reliability aspects of some access network topologies to
insure a certain level of quality of service at the lowest
cost for the end users, which are a little alike to our works.

III. HIERARCHICAL SEMI-CENTRALIZED ARCHITECTURE

A. Description of the architecture

The hierarchical semi-centralized architecture based on
basic grids is depicted in Fig. 1, which takes three
different types of access networks (UMTS, WLAN and
WiMax) for example. A basic grid is made up of several
adjacent cells. IS (Information Server), RA (Resource
Allotter) and RS (Resource Statistics) are collectively
referred to as Resource Management Unit (RMU), which
are responsible for managing the resources of basic grids.
Installed in the access point (AP), a RS is used to
calculate resources of its jurisdiction cell. A RA collects
load information from RSs, and balances the load in
virtue of the load and resources of the basic grid.
Normally an IS allocates resources for the borders of
basic grids and stores information of cell identification,
location and load states. However, it can take over the
broken RA immediately. In order to improve the system
reliability, a main IS and a standby IS are set up,
additionally, a RA and IS are connected by two junction
lines. Once the main IS (one junction line between RA
and IS) stops running, the standby IS (the other junction
line) will take over it.

A RA and the RSs located in the basic grid
administered by the RA compose the first layer
centralized architecture, while an IS (including a main IS
and a standby IS) and the RAs constitute the second layer
centralized architecture. The whole first layer is made up
of a series of basic grids and the corresponding RAs.
Thus, we call it the hierarchical semi-centralized
architecture.

The signaling overhead of transferring load
information between RSs and RAs (we call it the RSs to
RAs signaling overhead for short, and the similar terms
are used below) can be reduced by limiting the number of
RSs in a basic grid; the RAs to RS signaling overhead is
also small because of the limited number of RAs. The IS
can take over the RA immediately if the latter breaks
down, which ensures the communication of the basic
grid; the standby IS and the two junction lines between
each RA and IS make further improvement on system
reliability. Therefore, HSCA has the advantages of low
signaling overhead and high reliability, which are the
advantages of centralized architecture and distributed one
respectively.

There are two ways to fix the RA and IS. One way is
to install them in the existing equipments: RS in RNC
(Radio Network Controller) and RA in GGSN (Gateway
GPRS Support Node), the other way is to set them
separately. The advantage of the former is that it can
reduce the housing construction and maintenance cost,
while the disadvantage is that it is restrained by the existing network topology. The latter can design the network topology flexibly while increases the cost. Weighing the strong points and weaknesses of the two methods, a proper method to fix the RA and IS can be chosen.

B. Signaling flow

Normally a RS calculates load and resources of its jurisdiction cell according to the wireless parameters received from the AP, and then transfers the load, resources and location information to the RA. On the basis of load balancing algorithm (is not in the scope of this paper), the RA transfers load balancing information to RS. Then the RS changes load balancing information into load balancing instructions and transfers the load balancing instructions to the AP. The signaling flow chart is shown in Fig. 2.

![Signaling flow chart 1.](image)

When a subscriber is on the border of basic grids, the IS will allocate resources for the subscriber, and the signaling flow is shown in Fig. 3.

![Signaling flow chart 2.](image)

When a RA breaks down, the IS can take over it immediately, and the signaling flow is shown in Fig. 4.

![Signaling flow chart 3.](image)

IV. MODELING OF RELIABILITY AND SIGNALING OVERHEAD

In this section, we derive an analytical model for reliability of HSCA and two analytical models for signaling overhead of HSCA. The following notations are used in our analysis.

- $i$: System type. $i=1$ denotes UMTS system; $i=2$ denotes WLAN system; $i=3$ denotes WiMax system.
- $b_{ij}$: The traffic intensity between RS$_i$ and RA$_j$.
- $c_{ij}$: The traffic intensity between RA$_j$ and IS.
- $R_{IS}$: The reliability of IS.
- $R_{RA}$: The reliability of RA.
- $R_{IS}:$ The reliability of junction line between IS and RA.
- $a_i$: The signaling overhead of transferring load information once between one RS located in system type $i$ and RA.
- $d$: The signaling overhead of transferring load information once between one RA and IS.
- $e$: The signaling overhead of transferring load information once between one main IS and standby IS.
- $A_i$: The number of APs in system type $i$.
- $D$: The number of RAs.
- $A_{ij}$: The number of APs for system type $i$ in the basic grid $j$.
- $\lambda_i$: The traffic arrival rate of system type $i$.
- $\mu_i$: The service rate of system type $i$.
- $m_i$: The cell capacity of system type $i$.
- $k_{i1}$: The light threshold of system type $i$.
- $k_{i2}$: The heavy threshold of system type $i$.
- $T$: The period of transferring load information among RMU.

To facilitate the analysis, we assume that there are only one main IS and one standby IS.

A. Modeling of integrated reliability

Since the integrated reliability can reflect the practical situation very well [23], we use it to analyze the reliability of HSCA.

The HSCA we presented is actually a tree structure, as shown in Fig. 5.

When the number of RS is large, the breakdown of a RS or the junction line between a RS and the RA has little influence on the total traffic of the system. Thus, the breakdown of a RS or the junction line between a RS and the RA can be neglected.

Taking the main IS and the standby IS as a whole, the probability that the IS is in normal use is

$$p_a = 1 - (1 - R_{IS})^2.$$  \hspace{1cm} (1)
Taking the two junction lines between RA_{j} (j=1,2,...,D) and the IS as a whole, the probability that the junction line is out of fault is

\[ p_{0} = 1 - (1 - R_{j})^{2} . \]  

(2)

The probability that the whole system is in normal use is

\[ p_{b} = p_{0} \cdot p_{j}^{0} \cdot R_{k}^{0} \]  

(3)

then, the traffic of the whole system does not lose, namely

\[ L_{b} = 0 . \]  

(4)

The probability that \( K_{1} \) (\( K_{1}=1,2,...,D \)) junction lines between IS and RA are out of action is

\[ p_{k} = p_{0} \cdot \frac{D^{1}}{K_{1}(D-K_{1})} \cdot (1-p_{j})^{K_{1}} \cdot p_{j}^{0-K_{1}} \cdot R_{k}^{0} \]  

(5)

then, the traffic on the \( K_{1} \) junction lines loses, while all the traffic between RAs and RSs does not lose. Thus, the loss of the whole system traffic is

\[ L_{e} = \sum_{j=A}^{B} L_{j} . \]  

(6)

The probability that \( K_{2} \) (\( K_{2}=1,2,...,D \)) RAs are out of action is

\[ p_{k} = p_{0} \cdot R_{k}^{0} \cdot \frac{D^{1}}{K_{2}(D-K_{2})!} \cdot (1-p_{j})^{K_{2}} \cdot R_{k}^{0-K_{2}} \]  

(7)

then, the IS can take over the \( K_{2} \) breakdown RAs. Thus, the traffic of the whole system does not lose, namely

\[ L_{e} = 0 . \]  

(8)

The probability that the IS and \( K_{3} \) (\( K_{3}=1,2,...,D \)) RAs are out of action at the same time is

\[ p_{k} = (1-p_{b}) \cdot p_{j}^{0} \cdot \frac{D^{1}}{K_{3}(D-K_{3})!} \cdot (1-R_{k})^{K_{3}} \cdot R_{k}^{0-K_{3}} \]  

(9)

then, the traffic between all the RAs and the IS loses whether the junction lines between the RAs and the IS are out of action or not, and the traffic between the \( K_{3} \) broken RAs and the IS loses. Thus, the loss of the whole system is

\[ L_{e} = \sum_{j=A}^{B} c_{j} + \sum_{j=A}^{B} \sum_{i=0}^{k} n_{i} . \]  

(10)

The total traffic intensity of the system is

\[ B = \sum_{j=A}^{B} c_{j} + \sum_{j=A}^{B} \sum_{i=0}^{k} n_{i} . \]  

(11)

Therefore, the integrated reliability of the whole system is

\[ R = 1 - \frac{1}{T} \{ p_{b} \cdot L_{b} + p_{1} \cdot L_{1} + p_{2} \cdot L_{2} + p_{3} \cdot L_{3} \} . \]  

(12)

B. Modeling of signaling overhead

In the perspective of executing time, load balancing mechanisms can be classified as periodic and non-periodic [8]. Accordingly, the mode of transferring load information can be divided into periodic and non-periodic. Following we study the periodic and non-periodic signaling overhead respectively.

1) The periodic signaling overhead

The RSs to RAs signaling overhead in unit time is:

\[ O_{1} = \frac{1}{T} \sum_{j=A}^{B} (a_{j} \cdot A_{j}) . \]  

(13)

The RAs to IS signaling overhead in unit time is:

\[ O_{2} = \frac{1}{T} \cdot d \cdot D . \]  

(14)

The main IS to standby IS signaling overhead in unit time is:

\[ O_{3} = \frac{1}{T} \cdot e . \]  

(15)

The signaling overhead of transferring load information among RMU in unit time is:

\[ O_{i} = O_{1} + O_{2} + O_{3} = \frac{1}{T} \left[ \sum_{j=A}^{B} (a_{j} \cdot A_{j}) + d \cdot D + e \right] . \]  

(16)

For problem tractability, we assume that \( a_{1}=a_{2}=a_{3} \), then (16) can be reduced to:

\[ O_{i} = \frac{1}{T} \left[ A \cdot \sum_{j=A}^{B} A_{j} + d \cdot D + e \right] . \]  

(17)

2) The non-periodic signaling overhead

Enlightened by the ideas from [9] which employed two thresholds to classify cells into three classes, we introduce a heavy threshold and a light threshold to classify cells into three classes in terms of their load states: overloaded, under-loaded and balanced cells. A cell is in the under-loaded state when its load \( \leq k_{1} \), in the overloaded state when its load \( \geq k_{2} \); in the balanced state when \( k_{1} < \text{load} < k_{2} \).

RMU transfer load information when the following conditions are satisfied: RS transfers load information to RA when the load state of its jurisdiction cell changes; RA transfers load information to IS when the load state of one or more cells changes in the basic grid administered by the RA; the main IS transfers load information to
standby IS when one or more RAs transfer load information to the main IS.

For problem tractability, we assume that the arrival and service process are Poisson process and each cell is an independent M/M/m(m) queue.

According to queuing theory [25], the state probabilities of system type \(i\) are:

\[
P_s = e^{-\lambda_i T}_{\frac{(\lambda_i / \mu_i)^m}{m!}}
\]  

(18)

\[
P_u = \frac{(\lambda_i / \mu_i)^k}{k!} - P_s .
\]  

(19)

In time \(T\), the probability that a cell in system type \(i\) changes from under-loaded state to balanced state is:

\[
Pr(U \rightarrow B) = P_{u,i} \cdot \lambda T [1 - (k_i - 1)\mu T] .
\]  

(20)

In time \(T\), the probability that a cell in system type \(i\) changes from balanced state to overloaded state is:

\[
Pr(B \rightarrow O) = P_{u,i} \cdot \lambda T [1 - (k_i - 1)\mu T] .
\]  

(21)

In time \(T\), the probability that a cell in system type \(i\) changes from overloaded state to balanced state is:

\[
Pr(O \rightarrow B) = P_{u,i} \cdot (k_i + 1)\mu T (1 - \lambda T) .
\]  

(22)

In time \(T\), the probability that a cell in system type \(i\) changes from balanced state to under-loaded state is:

\[
Pr(B \rightarrow U) = P_{u,i} \cdot (k_i + 1)\mu T (1 - \lambda T) .
\]  

(23)

In time \(T\), the probability that the load state of a cell changes in system type \(i\) is:

\[
Pr_i = Pr(U \rightarrow B) + Pr(B \rightarrow O) + Pr(O \rightarrow B) + Pr(B \rightarrow U) .
\]  

(24)

In time \(T\), the probability that RA \(j\) transfers load information to IS is:

\[
Pr_j = 1 - \prod_{i=1}^{m} (1 - Pr_i)^k .
\]  

(25)

In time \(T\), the probability that main IS transfers load information to standby one is:

\[
Pr^- = 1 - \prod_{j=1}^{O} (1 - Pr_j) .
\]  

(26)

The signaling overhead of transferring load information among RMU in unit time is:

\[
O_2 = \frac{1}{T} \left[ a_i \cdot \sum_{j=1}^{O} (A \cdot Pr_j) + d \cdot \sum_{j=1}^{O} Pr_j + e \cdot Pr^- \right] .
\]  

(27)

V. SIMULATION STUDY

To evaluate the performance of HSCA, we employ a simulation model using the simulator MATLAB, and compare the signaling overhead of HSCA with that of SCSDA.

A. Simulation Scenario

The scenario is a medium urban area, where both UMTS system and WiMax system cover the whole area while WLAN system covers the hot spots only. In order to reduce the complexity of simulation, we assume that there are all the three types of APs in each basic grid, and the number of APs for the same system is equal in every basic grid.

The values of parameters used in simulation are as follows. \(T=0.1s, A_1=600, A_2=900, A_3=600; R_S=0.99, R_K=0.98, R_K=1; b=1erl; K_1=1, K_2=1, K_3=1; a_i=1, d=1, e=1; m_1=60, m_2=20, m_3=80; \eta_{stb}=0.7, \eta_{sel}=0.9; \mu_i=1/180s.\) Where \(i=1, 2, 3.\)

B. Simulation Results

Fig. 6 indicates the integrated reliability of HSCA with different number of RAs. Because of the assumption in Simulation Scenario, the number of RAs is discrete, and its value set is \{1, 2, 3, 4, 5, 6, 10, 12, 15, 20, 25, 30, 50, 60, 75, 100, 150, 300\}. We can see that the integrated reliability of HSCA rises with the number of RAs increasing, and the integrated reliability is always very high. When there is only one RA, the HSCA degenerate to be a centralized architecture with a lower reliability. The larger the number of RAs is, the more distributed the architecture is, and the higher the reliability is.

Fig. 7 and 8 illustrate the signaling overhead of HSCA and SCSDA with different number of RAs. The signaling overhead of SCSDA, which has nothing to do with the number of RAs, is associated only with the number of APs. Therefore, the signaling overhead of SCSDA is constant when the number of RAs varies. It can be observed that the signaling overhead of HSCA is always much smaller than that of SCSDA despite whether transferring load information is periodical or non-periodical, which indicates that HSCA has great advantages in reducing system signaling overhead.

Equation (17) tells us that the periodic signaling overhead includes three parts: the RSs to RAs signaling overhead, the RAs to IS signaling overhead and the main IS to standby IS signaling overhead. Since the number of RSs, main IS and standby IS is constant, the signaling overhead of the first part and the third part is invariable, while the second part is variable. We can see in (5) that the signaling overhead of the second part has a linear
relationship with the number of RAs. Therefore, the periodic signaling overhead of HSCA increases linearly with the rising of the number of RAs, which can be seen in Fig. 7.

![Figure 7. The periodic signaling overhead with different number of RAs](image1)

Fig. 7 shows that the periodic signaling overhead of HSCA rises rapidly with the number of RAs increasing from 1 to 20, while it rises slowly with the number of RAs increasing from 20 to 300. When there are a few RAs (less than 20), the number of APs in a basic grid is large, and the probability that one or more cells change load states in a basic grid is large, which makes the probability of transferring load information from RA to IS large, as a result, the signaling overhead of HSCA rises rapidly with the number of RAs increasing from 1 to 20; vice versa.

Comparing Fig. 7 with Fig. 8, it can be seen that the non-periodic signaling overhead of the two architectures is much smaller than the periodic one. Therefore, it can save lots of resources to employ the mode of transferring load information non-periodically.

The relationship between the non-periodic signaling overhead of the two architectures and the traffic arrival rate of the three systems are shown in Fig. 9, 10, and 11.

The following conclusions can be drawn from Fig. 9: in UMTS cells, the signaling overhead of the two architectures remains the same with \( \lambda_1 \) increasing from 0 to 0.15, rises gradually with \( \lambda_1 \) increasing from 0.15 to 0.32, drops gradually with \( \lambda_1 \) increasing from 0.32 to 0.65,

![Figure 9. The signaling overhead of the two architectures with different traffic arrival rate of UMTS (\( \lambda_2=0.2 \), \( \lambda_3=0.5 \), D=30)](image2)

and drops slowly with \( \lambda_1 \) increasing from 0.65 to 1. The above phenomenon is due to the following aspects. When \( \lambda_1 \) increases from 0 to 0.15, UMTS cells always have small amount of subscribers and are always in underloaded state, which causes RSs hardly to transfer load information to RA. When \( \lambda_1 \) increases from 0.15 to 0.32, the probability that the number of subscribers varies in the vicinity of \( k_{11} \) and \( k_{12} \) increases gradually, and system states change frequently. As a result, the probability that
RSs transfer load information to RA increases gradually. When \( \lambda_1 \) increases from 0.32 to 0.65, the probability that the number of subscribers exceeds \( k_{12} \) increases gradually, and the system states hardly change, which makes the probability that RSs located in UMTS APs transfer load information to RA reduce gradually. When \( \lambda_1 \) increases from 0.65 to 1, UMTS cells are almost always in the overloaded state as a result of large number of subscribers, which leads to the probability that RSs transfer load information to RA smaller and smaller.

The tendencies of the curves in Fig. 10 and 11 are similar with that in Fig. 9, and the reasons are also similar. It is unnecessary to give more details.

VI. CONCLUSIONS

In this paper, we proposed a hierarchical semi-centralized architecture based on basic grids for load balancing of heterogeneous wireless networks. An IS is not only a superior server for RAs but also a backup for RAs, which improves the system reliability; the standby IS and the two junction lines between each RA and IS make a further improvement on system reliability. It has been shown in our simulation that HSCA can reduce signaling overhead to a great degree while maintaining a very high reliability. The proposed network architecture has properly solved the problem of low reliability in centralized load balancing and high overhead in distributed load balancing.

APPENDIX A SIGNALING OVERHEAD OF SCSDA

According to the SCSDA presented in [3] and the scenario mentioned in this paper, the LBAs transfer load information as follows: a WiMax LBA transfers load information to its six neighboring WiMax LBAs, the UMTS LBAs and the WLAN LBAs whose coverage area overlaps part coverage area of the WiMax; a UMTS LBA transfers load information to the WiMax LBA whose part coverage area overlaps that of the UMTS, its six neighboring UMTS LBAs, and the WLAN LBAs whose coverage area overlaps part coverage area of the UMTS; a WLAN LBA transfers load information to the WiMax LBA and the UMTS LBA whose part coverage area overlaps that of the WLAN. WLAN system only covers the hot spots, and usually their coverage area doesn’t overlap. Therefore, the WLAN LBAs do not transfer load information to each other.

On the basis of the analysis above, we can derive the periodic signaling overhead and the non-periodic signaling overhead of SCSDA.

The periodic signaling overhead of transferring load information among LBAs in unit time is

\[
O_1 = \frac{1}{T} \left( a_i \cdot (4a_i + 2a_i + 3a_i) \right) \quad (A2)
\]

The non-periodic signaling overhead of transferring load information among LBAs in unit time is

\[
O_2 = \frac{1}{T} \left( a_i \cdot (4a_i \cdot 2a_i + 3a_i \cdot 2a_i + P_r) \right) \quad (A3)
\]

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China (No. 60972028).

REFERENCES


Wenxiao Shi received the Master degree of Engineering from Harbin Institute of Technology in 1991 and Ph.D in Communication and Information System from Jilin University in 2006. He is now a Professor of Department of Communication Engineering in Jilin University. His research interests include broadband communication theories and wireless communication network.

He is a co-author of the textbook principle and applications of communication network (Publishing House of Electronics Industry, 2008), Fast dynamic channel allocation algorithm based on priority channel reservation (Journal on Communications, 2009), and Hot-cell clusters deriving algorithm for slow dynamic channel allocation (Journal on Communications, 2009). He has a long-term cooperation with China Mobile Group Guangdong Company, engaging in the theory and technology research of programming and optimizing, performance evaluation and quality analysis in mobile communication network. He engaged in the research of wireless resource management technology, and was responsible for the project of The Science-Technology Plan of Jilin Province: Fast Dynamic Channel Allocation technology in TD-SCDMA; current is responsible for the project of The National Natural Science Foundation of China: researching on distributed Grid-based heterogeneous wireless network architecture and load balancing methods.

Prof. Shi is a member of Institute of Communications of Jilin Province and China Association of Automation of Jilin Province.

Bin Li received the Bachelor Degree in Communication Engineering from Jilin University in 2008. He is now a postgraduate student of Jilin University. His research interests include Core Network technology of GPRS, Radio Resource Management technology in TD-SCDMA and Load Balancing technology for Wireless Heterogeneous Networks. He is a co-author for two papers and a patent now.

Na Li received the Bachelor Degree in Communication Engineering from Jilin University in 2009. She is now a graduate student of Jilin University. His current research interests include the Load Balancing technology for Wireless Heterogeneous Networks.

Chuanjun Xia received the Bachelor Degree in Communication Engineering from Jilin University in 2008. He is now a postgraduate student of Jilin University. His current research interests include Load Balancing Technology for Wireless Heterogeneous Networks.
An Identity Based Aggregate Signature from Pairings

Yike Yu\textsuperscript{1,2}

\textsuperscript{1}School of Information Engineering, University of Science and Technology Beijing, Beijing 100083, P.R. China
\textsuperscript{2}College of Computer and Information Engineering, Jiang Xi Agricultural University, Nanchang 330045, P.R. China
yyk312@163.com

Xuefeng Zheng and Hua Sun
School of Information Engineering, University of Science and Technology Beijing, Beijing 100083, P.R. China
zxfustb@126.com, sh1227@163.com

Abstract—An aggregate signature is a useful digital signature that supports aggregation: Given $n$ signatures on $n$ distinct messages from $n$ distinct users, aggregate signature scheme is possible to aggregate all these signature into a single short signature. This single signature, along with the $n$ original messages will convince any verifier that the $n$ users did indeed sign the $n$ original messages respectively (i.e., for $i = 1,\ldots,n$ user $i$ signed message $m_i$).

In this paper, we propose an identity based aggregate signature scheme which requires constant pairing operations in the verification and the size of aggregate signature is independent of the number of signers. We prove that the proposed signature scheme is secure against existential forgery under adaptively chosen message and identity attack in the random oracle model assuming the intractability of the computational Diffie-Hellman problem.

Index Terms—Aggregate signature, Bilinear pairings, Identity-based cryptograph, Computational Diffie-Hellman problem

I. INTRODUCTION

In 1984, Shamir[1] brought forward the concept of Identity-based (from now on, ID-based) Cryptography as an alternative to traditional public key cryptography, based on infrastructures(PKI). In PKI-based cryptography, a certification authority must sign a digital certificate which links the identity of the user and his public key. Obviously, the management of digital certificates decreases the efficiency of practical implementations of public key cryptosystems. The idea of ID-based cryptography is that the public key of any user directly infers from his identity (e-mail address, telephone number, etc.). In 2001, Boneh and Franklin [2] proposed an secure and practical ID-based encryption scheme based on bilinear maps on an elliptic curve. Since then, several ID-based encryption and signature schemes have been proposed based on bilinear maps [3,4,5,6].

Many real-world applications involve signatures on many different messages generated by many different users. There are some efficiency problems (includes both communication and computation efficiency) of applications in which one entity has to verify many signatures simultaneously. For example, in a Public Key Infrastructure (PKI) of depth $n$, each user is given a chain of $n$ certificates. The chain contains $n$ signatures by $n$ Certificate Authorities (CAs) on $n$ distinct certificates. Similarly, in the Secure BGP protocol (SBGP) [7] each router receives a list of $n$ signatures attesting to a certain path of length $n$ in the network. A router signs its own segment in the path and forwards the resulting list of $n+1$ signatures to the next router. As a result, the number of signatures in routing messages is linear in the length of the path. Both applications would benefit from a method for compressing the list of signatures on distinct messages issued by distinct parties.

An aggregate signature scheme enables us to achieve precisely this type of compression. In 2003, Boneh et al. [8] introduced the concept of aggregate signatures and proposed the first aggregate signature (BGLS, based on BLS [8] short signature scheme) in which many signatures on different messages computed by different users can be aggregated into a single signature. In this scheme, while compressing $n$ signatures into one, it could compress only a half of signature. Ideally, the length of the aggregate signature (excluding the messages and the public keys of the signers) should be constant, independent of the number of signed messages. This concept is very useful in situations where a device must store many signatures, for example routing protocols in wireless networks requiring authentication.

Since aggregate signature was presented, several schemes have been proposed so far, e.g. [9-16] and the aggregate signatures are used in several fields such as contract signing, cascaded authorization, key agreement and wireless routing protocols, etc [17-21]. In 2004,
Lysyanskaya et al. [10] proposed a sequential signature aggregation, in which aggregation can only be done during the signing process. In 2006, Okstrovsky et al. [12] proposed a sequential aggregate signature without random oracles. However, its public key parameters are linear with the length of the messages signed, thus, it is not practice in fact.

There are two types of aggregate signatures according to the signature aggregation techniques.

- **General aggregate signatures** [8]. In a general signature aggregation scheme each user $i$ signs her message $M_i$ to obtain a signature $\sigma_i$. Then anyone can use a public aggregation algorithm to take all $n$ signatures $\sigma_1, \ldots, \sigma_n$ and compress them into a single signature $\sigma$.

- **Sequential aggregate signatures** [10]. In a sequential aggregation scheme, signature aggregation can only be done during the signing process. Each signer in turn sequentially adds her signature to the current aggregate. Thus, there is an explicit order imposed on the aggregate signature and the signers must communicate with each other during the aggregation process.

In 2004, Cheon et al. [22] presented the first ID-based aggregate signature (IBAS). In 2005, Xu [23] defined a security model and proposed an ID-based aggregate signature and proof of its security under the assumption of computational Diffie-Hellman problem (CDH-problem) in the random oracle model. In 2006, Herranz [24] proposed a deterministic ID-based signature for partial aggregation, which just allows aggregation when the signatures to be aggregated come all from the same signer. Gentry and Ramzan [25] presented the most efficient ID-based aggregate scheme which requires only three pairing computations (independent of the number of signers). In this scheme all the signers participating in aggregation have to agree upon a common randomness value which makes it unsuitable for most real life scenarios. In 2008, Wang et al. in [26] proposed a practical aggregate signature scheme with constant pairing operation but was able to achieve only partial aggregation. A valid user of the system will be able to forge a signature on any message by any user by just seeing a single signature on some message by the corresponding user.

In this paper, We give a formal definition of ID-based aggregate signatures and its security model. We then propose a new aggregate signature scheme, which requires only three pairing operations in verification (independent of the number of signers). Furthermore, our scheme’s security can be proved tightly related to CDH-problem in the random oracle model.

The rest of this paper is organized as follows. In section II we introduce preliminaries and the computational assumption which we take into considerations. In Section III we first give the security model for our aggregation signature, then propose an ID-based aggregate signature scheme and formally analyze its security and efficiency. And in Section IV we post the conclusions and the current open problems in this area.

### II. Preliminaries

Let $G$ and $G_i$ be two cyclic groups of the same large prime $q$. $G$ is a cyclic additive group and $G_i$ is a cyclic multiplicative group. We assume that the discrete logarithm problems in both $G$ and $G_i$ are hard.

#### A. Bilinear Pairing

**Definition 1.** (Bilinear Pairing) A bilinear pairing is a computable bilinear map between two groups $\hat{e} : G \times G \rightarrow G_i$. It is the modified Weil pairing or Tate pairing which has the following properties:

- **Bilinearity**
  $$\forall P, Q \in G, \forall a, b \in \mathbb{Z}_q, \hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}.$$  

- **Non-degeneracy**
  There exists $P, Q \in G$, such that $\hat{e}(P, Q) \neq 1$.

- **Computability**
  $\hat{e}(P, Q)$ can be efficiently computed for all $P, Q \in G$.

#### B. Gap Diffie-Hellman (GDH) Groups

**Definition 2.** (CDHP) Computational Diffie-Hellman Problem. For $\forall a, b \in \mathbb{Z}_q^*$, given $P, aP, bP \in G$, compute $abP \in G$.

**Definition 3.** (DDHP) Decision Diffie-Hellman Problem. For $\forall a, b, c \in \mathbb{Z}_q^*$, given $P, aP, bP, cP \in G$, decide whether $c = ab$. If so, $(P, aP, bP, cP)$ is called a valid Diffie-Hellman (DH) tuple.

It is commonly believed that there is no polynomial time algorithm to solve CDH problem with non-negligible probability.

**Definition 4.** The advantage of an algorithm $A$ in solving the CDH problem on group $G$ is

$$\text{Adv}_{A}^{\text{CDH}} = \text{Pr}[A(P, aP, bP) = abP : \forall a, b \in \mathbb{Z}_q^*]$$

The probability is taken over the choice of $a, b$ and $A$’s coin tosses. An algorithm $A$ is said $(t, \varepsilon)$-breaks the CDH problem on $G$ if $A$ runs in time at most $t$, and $\text{Adv}_{A}^{\text{CDH}}$ is at least $\varepsilon$.

**Definition 5.** A group $G$ is a $(t, \varepsilon)$-gap Diffie-Hellman (GDH) group if the DDH problem in $G$ can be efficiently computable and there exists no algorithm $(t, \varepsilon)$-breaks CDH in $G$.

If we have an admissible bilinear pairing $\hat{e} \in G$, we can solve the DDH problem in $G$ efficiently as follows:

$$(P, aP, bP, cP) \text{ is a valid DH tuple } \iff \hat{e}(aP, bP) = \hat{e}(P, cP).$$

Hence an elliptic curve becomes an instance of a GDH group if the Tate (or the Weil) pairing is efficiently computable and the CDH is sufficiently hard on the curve.

**Definition 6.** Two order group $(G, G_i)$ is a $(\hat{e}, t, \varepsilon)$-gap Diffie-Hellman (GDH) groups if there exists an admissible bilinear pairing $\hat{e} : G \times G \rightarrow G_i$ but there exists no algorithm $(t, \varepsilon)$-breaks CDH in $G$.

### III. ID-BASED AGGREGATE SIGNATURE
We define ID-based aggregate signatures and describe the security model for ID-based aggregate signatures which defined in [23].

A. Definition of IBAS

Consider a set \( U \) of users. Each user \( u \in U \) has an unique identity \( ID_u \), and his correspond signing key pair is \( (ID_u, sk_u) \). We wish to aggregate the signatures of some subset \( U' \subseteq U \). Each user \( u_i \in U' \) produces a signature \( \sigma_i \) on a message \( m_i \) of her choice. These signatures are then combined into a single aggregate signature \( \sigma \) by an aggregating party. The aggregating party verifies the signature that if it is valid.

- For each user \( u_i \), returns \( \sigma_i \) along with \( ID_i \) and \( m_i \), \( i \in U' \).

B. Security model of IBAS

Here describes a security model for aggregate signature in [23]. In this aggregate model, the adversary \( \mathcal{F} \) is given a single \( ID \). His goal is the existential forgery of an aggregate signature. An aggregate forger is allowed to choose all identities except the challenge \( ID \). The aggregate forger is also given access to a signing oracle with respect to the challenge \( ID \). His advantage \( \text{Adv}_{\text{sig},\mathcal{F}}^{\text{agg}} \) is defined to be his probability of success in the following game.

- **(Setup:)** The aggregate forger \( \mathcal{F} \) is provided with \( ID \), which is an identity generated at random.

- **(Extraction Queries:)** Given an identity \( ID_i \) \( (i \neq 1) \), the challenger returns the private key \( sk_i \) corresponding to \( ID_i \).

- **(Signature Queries:)** Proceeding adaptively, \( \mathcal{F} \) requests signatures with respect to identity \( ID_i \) on messages of his choice.

- **(Response:)** Finally, \( \mathcal{F} \) outputs \( n-1 \) additional identities \( ID_1 , ID_2 , \ldots , ID_n \). Here \( n \) is at most \( N \), a game parameter. The forger \( \mathcal{F} \) shall also output messages \( m_1 , m_2 , \ldots , m_n \) and an aggregate signature \( \sigma \) with respect to these \( n \) identities, on the corresponding messages.

The forger \( \mathcal{F} \) wins if the aggregate signature \( \sigma \) is valid on messages \( m_1 , m_2 , \ldots , m_n \) under \( ID_1 , ID_2 , \ldots , ID_n \), and \( \mathcal{F} \) did not request a signature on \( m_i \) under \( ID_i \).

The probability is over the coin tosses of the key-generation algorithm and of \( \mathcal{F} \).

IV. THE PROPOSED ID-BASED AGGREGATE SIGNATURE SCHEME

A. IBAS Scheme

This IBAS scheme consists of six algorithms: \( G , K , S , V , AS , AV \). The first four algorithms are as in ordinary ID-based signature schemes; the last two provide the aggregation capability. It works as follows.

- **(Setup:)** The aggregate forger \( \mathcal{F} \) runs the setup algorithm which outputs \( G , K , S , V , AS , AV \). The system’s public parameters are \( params \) which defined in [23].

- **(Response:)** Finally, \( \mathcal{F} \) outputs \( n-1 \) additional identities \( ID_1 , ID_2 , \ldots , ID_n \). Here \( n \) is at most \( N \), a game parameter. The forger \( \mathcal{F} \) shall also output messages \( m_1 , m_2 , \ldots , m_n \) and an aggregate signature \( \sigma \) with respect to these \( n \) identities, on the corresponding messages.

The forger \( \mathcal{F} \) wins if the aggregate signature \( \sigma \) is valid on messages \( m_1 , m_2 , \ldots , m_n \) under \( ID_1 , ID_2 , \ldots , ID_n \), and \( \mathcal{F} \) did not request a signature on \( m_i \) under \( ID_i \). The probability is over the coin tosses of the key-generation algorithm and of \( \mathcal{F} \).
U = \sum_{i=1}^{n} U_i \quad h_i = \text{H}_i(\text{ID}_i, m_i, U) \quad (1 \leq i \leq n). He accepts the signature \sigma if and only if \( \hat{a}(P, V) = \text{a}(Q, U) \cdot \hat{a}(P_{\text{pub}}, h_{Q_{m_i}}) \).

- **AS**: We assume that the single signatures are all valid. DP computes \( V = \sum_{i=1}^{n} V_i \). The aggregate signature on \( n \) different messages \( m_1, m_2, \ldots, m_n \) given by \( n \) users \( P_1, P_2, \ldots, P_n \) is \( \sigma = (U, V) \).

- **AV**: The verifier is given an aggregate signature \( \sigma = (U, V) \) as above, the list of \( \langle \text{identity}, \text{message} \rangle \) pairs \( \langle \text{ID}_i, m_i \rangle \), indexed as before. To verify the aggregate signature \( \sigma \), the verifier computes \( Q_{\text{id}} = \text{H}_z(\text{ID}) \) and \( h_i = \text{H}_i(\text{ID}_i, m_i, U) \) for \( i = 1, 2, \ldots, n \). The aggregate signature \( \sigma \) is accepted if and only if \( \hat{a}(P, V) = \hat{a}(Q_1, U) \cdot \hat{a}(P_{\text{pub}}, \sum_{i=1}^{n} h_{Q_{m_i}}) \).

**B. Correctness Analysis**

- Correctness of the single signature:
  \[ \hat{a}(P, V) = \hat{a}(Q, P) \cdot \hat{a}(P_{\text{pub}}, h_{Q_{m_i}}) = \hat{a}(Q, U) \cdot \hat{a}(P_{\text{pub}}, h_{Q_{m_i}}) \]
- Correctness of the aggregate signature:
  \[ \hat{a}(P, V) = \hat{a}(P, \sum_{i=1}^{n} V_i) = \prod_{i=1}^{n} (P_{\text{pub}} \cdot h_{Q_{m_i}}) = \prod_{i=1}^{n} (P_{\text{pub}} \cdot h_{Q_{m_i}}) = \hat{a}(Q, \sum_{i=1}^{n} U_i) \cdot \hat{a}(P_{\text{pub}}, \sum_{i=1}^{n} h_{Q_{m_i}}) = \hat{a}(Q, U) \cdot \hat{a}(P_{\text{pub}}, \sum_{i=1}^{n} h_{Q_{m_i}}) \]

**C. Security Analysis**

Here we define the IBAS security model based on Xu’s model [23]. In the model, the adversary \( A \) is given a single \( \text{ID}^* \). His goal is the existential forgery of an aggregate signature. We allow an adversary \( A \) to corrupt all but one honest signer \( \text{ID}^* \). His advantage \( \text{Adv}^{\text{IBAS}}_A \) is defined to be his probability of success in the following game.

- **Setup**: The challenger \( C \) runs the setup algorithm and generates the \( \text{params} \) and master secret key \( s \). The Challenger \( C \) gives the \( \text{params} \) to the adversary \( A \). The adversary \( A \) is provided with \( \text{ID}' \), which is an identity generated at random.

- **Extraction Queries**: Given an identity \( \text{ID} \) \( (\text{ID} \neq \text{ID}') \), the challenger returns the private key \( s_{\text{id}} \) corresponding to \( \text{ID} \).

- **Signature Queries**: Proceeding adaptively, \( A \) requests signatures with respect to identity \( \text{ID}_i \) on message \( m_i \) of his choice.

- **Response**: Finally, \( A \) outputs \( n \) identities \( \text{ID}_1, \text{ID}_2, \ldots, \text{ID}_n \). Without lose of generality, we assume \( \text{ID}_1 = \text{ID}' \), \( n \) is at most \( N \), a game parameter. The adversary \( A \) shall also output messages \( m_1, m_2, \ldots, m_n \) and an aggregate signature \( \sigma \) with respect to these \( n \) identities, on the corresponding messages.

The forger \( A \) wins if the aggregate signature \( \sigma \) is valid on messages \( m_1, m_2, \ldots, m_n \) under \( \text{ID}_1, \text{ID}_2, \ldots, \text{ID}_n \), and \( A \) did not request a signature on \( m_i \) under \( \text{ID}_i \). The probability is over the coin tosses of the key-generation algorithm and of \( A \).

**Definition 7**. An aggregate adversary \( A \) is said \( (t, q_{H}, q_{E}, \ell, e, N) \)-breaks an \( N \)-user aggregate signature scheme in the aggregate model if: \( A \) runs in time at most \( t \); \( A \) makes at most \( q_{H} \) queries to the hash function \( H_i (i = 1, 2) \), at most \( q_{E} \) queries to the key extraction oracle and at most \( q_{E} \) queries to the signing oracle; \( \text{Adv}^{\text{IBAS}}_A \) is at least \( e \). An IBAS scheme is \( (t, q_{H}, q_{E}, \ell, e, N) \)-secure against existential forgery in the aggregate model if no forger \( (t, q_{H}, q_{E}, \ell, e, N) \)-breaks it.

**Theorem 1**. Let \( G \) be a \( (t, \epsilon) \)-gap Diffie-Hellman (GDH) group of prime order \( q \). Then the ID-based aggregate signature scheme on \( G \) is \( (t, q_{H}, q_{E}, \ell, e, N) \)-secure against existential forgery in the aggregate model for any \( t \) and \( e \) satisfying

\[ e \geq e(q_{E} + N) \epsilon \]

\[ t \leq t - C_{\epsilon}(q_{H} + q_{E} \cdot 4q_{E} + N + 2) \]

where \( e \) is the base of natural logarithms, and \( C_{\epsilon} \) is the time of computing a scalar multiplication and inversion on \( G \).

**Proof**. Suppose \( A \) is a forger algorithm that \( (t, q_{H}, q_{E}, \ell, e, N) \)-breaks the IBAS scheme. We show how to construct a \( t \)-time algorithm \( C \) that solves CDH in \( G \) with probability at least \( \epsilon \). This will contradict the fact that \( G \) is a \( (t, \epsilon) \)-GDH group.

We may assume the forger is well-behaved in the following sense: A forger \( A \) makes an \textbf{Extraction} query for an ID only if a \( H_1 \) query has been made before for the ID. Also, a \textbf{Signature} query is made for a message \( m \) only if a \( H_1 \) queries has been made before for the \( m \).

Algorithm \( C \) is given \( aP \in G \) and \( bP \in G \). Its goal is to output \( abP \in G \). Algorithm \( C \) simulates the challenger and interacts with forger \( A \) as follows.

**Setup**: Algorithm \( A \) choose a random value \( t \in \mathbb{Z}_q^* \), computes \( Q = tP \), let \( P_{\text{pub}} = aP \) as a system’s overall public key and choose a random identity \( \text{ID}' \) (here, we assume \( \text{ID}' = \text{ID}_1 \)), sends the system public parameters \( \text{params} \) and \( \text{ID}' \) to the adversary \( A \).

**H_1-Query**: To respond to queries to \( H_1 \) oracle, \( C \) maintains a list \( L_0 \) of tuple \( (\text{ID}_i, m_i, U_i, v_i) \) as explained
below. When a tuple \((ID, m, U)\) is submitted to the \(H_i\) oracle, algorithm \(C\) responds as follows:

1. If the query tuple already appears on the \(L_i\) in some tuple \((ID, m, U, v)\) then algorithm responds with \(H_i(ID, m, U, v) = v\).

2. Otherwise, algorithm \(C\) picks \(v \in \mathbb{Z}_q^*\) at random, store the tuple \((ID, m, U, v)\) in the list \(L_i\) and returns \(v\) as a hash value to \(A\).

**\(H_2\)-Query:** At any time algorithm \(A\) can query the random oracle \(H_2\). To respond to these queries, \(C\) maintains a list \(L_2\) of tuples \((ID, w, x, y)\) as explained below. The list is initially empty. When an identity ID is submitted to the \(H_2\) oracle, algorithm \(C\) responds as follows:

1. If the query ID already appears on the \(H_2\) in some tuple \((ID, w, x, y)\) then algorithm \(C\) responds with \(H_2(ID) = w \in \mathbb{G}\).

2. Otherwise, \(C\) generates a random coin \(y \in \{0, 1\}\) such that \(Pr[y = 0] = 1/(q_z + N)\).

3. Algorithm \(C\) picks a random \(x \in \mathbb{Z}_q\). If \(y = 0\) holds, \(C\) computes \(w = x(bP) \in \mathbb{G}\). If \(y = 1\) holds, \(C\) computes \(w = xP \in \mathbb{G}\).

4. Algorithm \(C\) adds the tuple \((ID, w, x, y)\) to the list \(L_2\) and returns to \(A\) with \(H_2(ID) = w \in \mathbb{G}\).

**Exaction Query:** When \(A\) requests the private key associated to an identity ID, \(C\) recovers the corresponding \((ID, w, x, y)\) from \(L_2\). If \(y = 0\), then \(C\) output “failure” and halts. Otherwise, it means that \(H_2(ID)\) was previously defined to be \(xP\) and \(xP_{ab} = x(aP) \in \mathbb{G}\) is then returned to \(A\) as a private key associated to ID.

**Signature Query:** Algorithm \(A\) requests a signature on some message \(m\) under \(ID\). Algorithm \(C\) responds to this query as follows: algorithm \(C\) first recovers the corresponding \((ID, w, x, y)\) from \(L_2\).

1. If \(y = 0\), then \(C\) output “failure” and halts.

2. Otherwise, it means that \(H_2(ID) = xP\). Algorithm \(C\) picks a random \(r \in \mathbb{Z}_q\), computes \(U = rP\). Then Algorithm \(C\) searches the tuple \((ID, m, U, v)\) in the list \(L_i\), if no exist, then picks a random \(v \in \mathbb{Z}_q^*\), and adds the tuple \((ID, m, U, v)\) to the list \(L_i\).

3. Algorithm \(C\) computes \(V = rQ + v(xP_{ab})\), and gives \(\sigma = (U, V)\) to algorithm \(A\). It is obvious that \(\sigma\) is a valid signature on message \(m\) under \(ID\).

**Output:** Finally, \(A\) halts. It either concedes failure, in which case so does \(C\), or it returns a forged aggregate signature \(\sigma\) on message \(\{m_1, m_2, \ldots, m_l\}\) under \((ID_1, ID_2, \ldots, ID_l)\). Forger \(A\) must not have requested a signature on \(m_i\) under \(ID_i\).

Algorithm \(C\) recovers the corresponding \(n\) tuples \((ID, w_i, x_i, y_i)\) on the list \(L_2\). Algorithm \(C\) now proceeds only if \(y_i = 0\) and \(y_i = 1\) for \(2 \leq i \leq n\). Otherwise, \(C\) declares failure and halts. Since \(y_i = 0\), it follows that \(Q_{ab_i} = x_i(bP)\). And for \(i > 1\), since \(y_i = 1\), it follows that \(Q_{ab_i} = x_iP\). The aggregate signature \(\sigma = (U, V)\) must satisfy the aggregate verification equation

\[\hat{e}(P, V) = \hat{e}(Q, U) \cdot \hat{e}(P_{ab}, \sum_{i=1}^{n} h_i Q_{ab_i})\]

Next, algorithm \(C\) recovers the \(n\) corresponding tuples \((ID_i, m_i, U_i, v_i)\) on the list \(L_i\). Let \(V_i = tU_i + v_i x_i P_{ab}\) for \(i > 1\). Then we have

\[\hat{e}(P, V_i) = \hat{e}(P, tU_i + v_i x_i P_{ab}) = \hat{e}(Q, U_i) \cdot \hat{e}(P_{ab}, v_i Q_{ab_i})\]

\[= \hat{e}(Q, U_i) \cdot \hat{e}(P_{ab}, h_i Q_{ab_i})\]

Thus \((U_i, V_i)\) is a valid signature on \(m_i\) under \(ID_i\).

Now \(C\) constructs \(V_i\) as \(V_i = V - \sum_{i=2}^{n} V_i\). Then we can deduce

\[\hat{e}(P, V_i) = \hat{e}(P, V - \sum_{i=2}^{n} V_i)\]

\[= \prod_{i=2}^{n} \hat{e}(Q, U_i) \cdot \hat{e}(P_{ab}, h_i Q_{ab_i}) \cdot \hat{e}(Q, U_i) \cdot \hat{e}(P_{ab}, h_i Q_{ab_i})^{-1}\]

\[= \hat{e}(Q, U_i) \cdot \hat{e}(P_{ab}, h_i Q_{ab_i}) = \hat{e}(P, tU_i) \cdot \hat{e}(x_i P, abP)\]

Here \(t_i = v_i\). Then \(C\) calculates and outputs the required \(abP\) as \(abP = v_i^{-1} x_i^{-1} (V_i - tU_i)\).

This completes the description of algorithm \(C\). To complete the proof, we shall show that \(C\) solves the given instance of CDH problem in \(\mathbb{G}\) with probability at least \(\epsilon\). First, we analyze the four events needed for \(C\) to succeed:

- **\(E_1\):** \(C\) does not abort as a result of any of \(A\)’s key extraction queries.
- **\(E_2\):** \(C\) does not abort as a result of any of \(A\)’s signature queries.
- **\(E_3\):** \(A\) generates a valid and nontrivial aggregate signature forgery \(\sigma = (U, V)\).
- **\(E_4\):** Event \(E_0\) occurs, and, in addition, \(y_i = 0\), and \(y_i = 1\) for \(2 \leq i \leq n\) where for each \(i\), \(y_i\) is the \(y\)-component of the tuple containing \(ID_i\) on the list \(L_2\).

Algorithm \(C\) succeeds if all of these events happen. The probability \(Pr[E_1 \land E_2 \land E_3 \land E_4]\) can be decomposed as

\[Pr[E_1 \land E_2 \land E_3 \land E_4] = Pr[E_1] \cdot Pr[E_2] \cdot Pr[E_3] \cdot Pr[E_4] \cdot Pr[E_1 \land E_2 \land E_3] \]

**Claim 1.** \(Pr[E_1] \geq (1-(1/(q_z + N))^N)\)

**Proof:** The probability that algorithm \(C\) does not abort as a result of \(A\)’s key extraction queries is at least \((1-(1/(q_z + N))^N)\). Hence \(Pr[E_1] \geq (1-(1/(q_z + N))^N)\).

**Claim 2.** \(Pr[E_2 | E_1] = 1\).
Proof: The probability that algorithm C does not abort as a result of A’s signature queries is 1. Hence \( Pr[E_1 | E_1 \wedge E_2] = 1 \).

Claim 3. \( Pr[E_1 | E_1 \wedge E_2] \geq \varepsilon \).

Proof: If algorithm C does not abort as a result of A’s signature queries and key extraction queries then algorithm A’s view is identical to its view in the real attack. Hence, \( Pr[E_1 | E_1 \wedge E_2] \geq \varepsilon \).

Claim 4.
\[
Pr[E_1 | E_1 \wedge E_2, E_2] \geq (1 - 1/(q_\ell + N))^{x-1} \cdot 1/(q_\ell + N).
\]

Proof: The probability that algorithm C does not abort after A outputting a valid and nontrivial forgery is at least
\[
(1 - 1/(q_\ell + N))^{x-1} \cdot 1/(q_\ell + N).
\]
Hence, \( Pr[E_1 | E_1 \wedge E_2, E_2] \geq (1 - 1/(q_\ell + N))^{x-1} \cdot 1/(q_\ell + N) \).

According to the above discussion, we can conclude that algorithm C produces the correct answer with probability at least
\[
(1 - 1/(q_\ell + N))^{x-1} \cdot \varepsilon \cdot (1 - 1/(q_\ell + N))^{x-4} \cdot 1/(q_\ell + N)
\]
\[
\geq \varepsilon \cdot (q_\ell + N) \geq \varepsilon.
\]
Hence, \( \varepsilon \geq \varepsilon(q_\ell + N) \varepsilon \) as required.

Algorithm C’s running time is the same as A’s running time plus the time to respond to \((q_m + q_n + q_s)\) hash queries, \( q_s \) key extraction queries and \( q_s \) signature queries, and the time to transform A’s final forgery into the CDH solution. Hence, the total running time is at most \( t + C_\varepsilon(q_\ell + q_s) + q_s + 4q_s + N + 2) \leq t \) as required.

This completes the proof of Theorem.

D. Efficiency Analysis

Given \( n \) signatures \( \sigma_1 = (U_1, V_1), \ldots, \sigma_n = (U_n, V_n) \) for messages \( m_1, m_2, \ldots, m_n \) issued by \( ID_1, ID_2, \ldots, ID_n \) respectively. When we verify these signatures, we can first aggregate them to \( \sigma = (U, V) \) and then verify the aggregated signature, here \( U = \Sigma_{i=1}^{n} U_i, V = \Sigma_{i=1}^{n} V_i \). The \( n \) signatures are accepted if and only if
\[
\hat{a}(P \Sigma_{i=1}^{n} U_i) - \hat{a}(P \Sigma_{i=1}^{n} Q_i) = 0.
\]

Since elliptic curve additions and hash operations are far more efficient than pairing operations, the aggregate verification of our scheme is more efficient than individual verification of signatures, it only requires three pairing operations. And we can see that the aggregate signature 'length' is same as the individual signature’s.

We compare our scheme with other ID-based aggregate signature schemes for \( k \) signatures from computation overhead view point.

Let \( C_1 \) be the pairing operation, \( C_2 \) be the point scalar multiplication in \( G \), \( C_3 \) be the point addition in \( G \), and \( C_4 \) be the exponential operation in \( G \), respectively. We summarize the results in Table I (we ignore the general hash operation).

V. Conclusion

In this paper we proposed an ID-based aggregate signature scheme with constant pairings. Our scheme is secure against existential forgery under adaptively chosen messages attacks, and the security is tightly related to Computational Diffie-Hellman problem in the Random Oracle model. Our aggregate signature scheme is based on bilinear pairing. Just like all other pairing based cryptosystems, it is simple, efficient and has short signature size.

Finding an efficient aggregate signature with constant pairing computations in verification and constant size of aggregate signature without any interaction among users remains as an interesting open problem in this field, also developing an aggregate signature in the standard model is another open problem to look at.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of P.R. China (No.60674054 and No.60673187).

REFERENCES


Yike Yu is an Assistant Professor in College of Computer and Information Engineering at Jiang Xi Agricultural University, P.R. China. He also is Ph.D. candidate in School of information engineering at university of science and technology Beijing, P.R. China. His research interests include the provable security of public-key cryptosystem, identity-based cryptosystem, elliptic curve cryptosystem.

Xuefeng Zheng is Professor and Ph.D. Supervisor in School of Information Engineering at University of Science and Technology Beijing, P.R. China. His research interests include the security of sensor net, cryptosystem, network security.

Hua Sun is Ph.D. candidate in School of information engineering at university of science and technology Beijing, P.R. China. His research interests include identity-based cryptosystem, elliptic curve cryptosystem, network security.
Detecting Malware Variants by Byte Frequency

Sheng Yu, Shijie Zhou, Leyuan Liu, and Rui Yang
School of Computer Science & Engineering, University of Electronic Science and Technology of China
Chengdu, China
Email: {yusheng123, sjzhou16, leyuanliu}@gmail.com, yru@uestc.edu.cn

Jiaqing Luo
Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China
Email: csjluo@comp.polyu.edu.hk

Abstract—In order to make lots of new malwares fast and cheaply, attacker can simply modify the existing malwares based on their binary files to produce new ones, malware variants. Malware variants refer to all the new malwares manually or automatically produced from any existing malware. However, such simple approach to produce malwares can change signatures of the original malware so that the new malware variants can confuse and bypass most of popular signature-based anti-malware tools. In this paper we propose a novel byte frequency based detecting model (BFBDM) to deal with the malware variants identification issue. The byte frequency of software refers to the frequency of the different unsigned bytes in the corresponding binary file. In order to implement BFBDM, two metrics, the distance and the similarity between the suspicious software and base sample, a known malware, are defined and calculated. According to the experimental results, we found out that if the distance is low and the similarity is high, the suspicious software is a variant of the selected malware with very high probability. The primary experimental results show that our model is efficient and effective for the identification of malware variants, especially for the manual variant.

Index Terms—Malware variants, malware identification, byte frequency, byte distribution, software proximity

I. INTRODUCTION

Malware spreading is a serious problem in real life network. Malware, which is also known as malicious software or malicious code, is the software designed to realize some malicious and shady goals on the attacked computers or the networks [1]. Viruses, worms, and spywares are practical examples of malwares. The malware is one of the most severe risks of computers and networks. Unfortunately, producing and distributing malwares now strongly relate to gaining economic profit [2] [3]. For example, the master of a botnet can gain profits by helping companies to launch DDoS attacks to the web sites of their rivals. The wide spreading of the Trojan that steals the online game account aptly illustrates the risk of the malwares.

The harm of malwares stems from not only their destructive power, but also their vast volume on the Internet. The volume of malwares is made up of two parts: the newly produced malwares and the malware variants.

Making a completely new malware from scratch requires the makers should be skillful specialists in computer and information security. Moreover, it also takes the maker tons of times and energies to code, test and debug. Another way to produce the new malwares is to modify the existing malwares based on their source code, which is simpler and easier slightly. All of them are referred as new malwares.

Another popular approach attackers adopt to produce “new” malwares is to modify the existing malwares based on their binary file. Such “new” malware is known as the malware variants. Compared with producing completely new malwares, making malware variant is simple and laborsaving. For example, if the maker can only access the binary file of the malware, which is the most common case, the skilled maker can directly modify the binary file manually to produce malware variants. Although manually modifying binary file needs more skills and professional knowledge in the realm of reverse engineering, it is not a big problem for professional malware makers or skillful attackers. These “new” malware is referred as malware variant.

However, such simple approach to produce malwares makes a heavy handicap to detect and defense malwares. Most of existing malware detection and defense tools rely on malware signatures. That is, signature of a malware must be constructed before using it. The malware variants produced from the original malware manually are able to change their signatures skillfully and aptly to bypass the antivirus software. Accordingly, the original signatures constructed from the original malwares can not directly used to detect and defend their malwares variants. Therefore, the subtle attackers can employ code obfuscation technique to produce a mass of malware variants to get rid of being detected by anti-malware tools.

It was reported that 18 malwares among the top twenty malicious programs in 2008 have variants [2]. According to the observation and statistic analysis of ESET [4], on April 13, 2009, all the top 5 threats in the last 24 hours have variants. Therefore, the overwhelming amount of malware variants brings more difficulties for the identification of malwares.
Consequently, how to detect malwares and its variants effectively and efficiently is a hot topic. In this paper we propose a novel byte frequency based detecting model (BFBDM) to deal with the problem of malware variants detection. In our novel model, the byte frequencies of any known malwares and the suspicious malwares are computed. Then, if a suspicious malware is similar to any one known malware in terms of byte frequency, which is indicated by the distance and similarity between them, the suspicious malware is determined to be a variant of the latter. Primary experimental results show that our novel model is effective and efficient.

The rest of this paper is organized as follows. In section 2, the related work of the malware variants detection is introduced. The BFBDM for malware variants is present in section 3. In the section 4, some experiments are provided and analyzed. The complexity of BFBDM is discussed in the section 5. Finally, we make the conclusion in section 6.

II. RELATED WORK

Initially, the malwares are detected by their signatures. Signature based detection model involves searching for some special patterns of known malwares in executable code. The signature patterns can be simple binary sequence, binary sequence with mask byte, or specially designed checksum [5]. Signature based detection model is one of the best ways to identify known malware. But the signature based detection model is not effective for the new malwares and malware variants.

An effective way to detect new malwares and malware variants is to analyze them based upon semantics [6] [7]. Detecting the malware by the software’s behaviors is another way [8]. The semantics based detection model analyzes the software statically, while the behavior based detection model dynamically.

The semantics based detection model searches a binary file for malware-like instructions. This model analyzes the file based upon the instructions’ semantics, not the instructions themselves. For example, the instruction “add eax, 1” and the instruction “inc eax” have the same function. So in the semantics model they are treated as the same code. If the instructions fit in some malware-like patterns, such as remote thread injecting, the software is detected as malware.

The behavior based detection model monitors the suspicious program behavior in the real system or virtual machine environment. Unlike the semantics model analyzes the software’s activities base upon the instructions’ semantics, the behavior model monitors and analyzes the running software’s actual activities.

The heuristic approach is also a hot area of malware detection [9]. The suspicious OEP, sections’ characteristics, API calling, multiple PE headers, and some other features can be synthetically used to detect malware.

Additionally, the algorithmic scanning, which is sometimes named as virus-specific detection algorithm, is introduced to deal with specific malware [9]. The geometric detection, skeleton scanning and neural network based detection model are also discussed [9].

In this paper we are considering identification and detection of the malware variants. The packed malware [10] and polymorphic malware [11] are not discussed in this paper.

All the malware detection models can be classified as non-exact identification and exact identification. The non-exact model only identifies whether it is a malware or not. However, the exact identification needs to identify the version detail of malware variants. Our BFBDM is of non-exact identification.

III. A BYTE FREQUENCY BASED DETECTION MODEL FOR MALWARE VARIANTS

A. Problem Statement

Code obfuscations [12] are some methods used to produce malware variants by changing the contents of binary files of malwares while preserving their destructive functions. Popular code obfuscations include code reordering, packing, junk insertion, register reassignment, instruction replacement, case switch, and so on. Obfuscations, especially manual obfuscations do change the content of the malware files, especially the signature bytes, which enables them to bypass the antivirus software. However, most of them change only tens of bytes. This little change results in that the byte frequency varies lightly on the statistical level. So in this paper we choose the byte frequencies of malware files as the statistic metric.

For example, table 1 shows a segment of codes and its bytes based contents in hexadecimal. The code denotes the instructions on the IA 32 architecture machine. The byte refers to the contents stored in the disk file.

The byte frequency of the above code fragment is shown in table 2. If this code segment is signature of one known malware, the attacker could use any code obfuscation techniques to produce many malware variants. For example, the code reordering can be used to change the orders of two or more instructions or program sentences. Apparently, although the order of the two instructions in table 1 is changed, its byte frequency keeps the same as in table 2.

From the above example, it’s clear that usually the code obfuscations change little byte frequency of the software.

<table>
<thead>
<tr>
<th>TABLE I. CODE FRAGMENT</th>
<th>Code fragment</th>
<th>Byte(hexadecimal)</th>
</tr>
</thead>
<tbody>
<tr>
<td>mov eax, 1</td>
<td>B8 01000000</td>
<td></td>
</tr>
<tr>
<td>xor ebx, 255</td>
<td>21F3 55020000</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TABLE II. BYTE FREQUENCY OF CODE</th>
<th>Byte</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x00</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>0x01</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0x02</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0x21</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>0x55</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0xB8</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
Therefore, the byte frequency can be used to determine whether a suspicious file is the variant of known malware.

Being similar with the functions’ CRC32 checksum proposed by [5], our model tries to inspect the malware variants on the statistic level. Since the detection model focusing on the detail information has been studied many years, the detail based model has been developed fully. If we turn our energy to the statistic level, maybe we can make some breakthroughs.

B. The Byte Frequency Based Detection Model

In this section, we will discuss how to construct the byte frequency based detection model.

If we treat all files as binary files, all files are composed by tons of bytes. Considered as an unsigned integer, every byte’s value varies from 0 to 255. Traversing a file as binary file, we can get the count of the byte of value 0, that of value 1, that of value 2 and so on. So we can get the file’s byte frequency.

In the following discussion, the software A’s byte frequency array is denoted as \( M = \{m_1, m_2, m_3, ..., m_{254}, m_{255}\} \), where the \( m_i \) indicates how many bytes in the software A are of value \( i \) (\( i = 1, 2, ..., 255 \)). Similarly, the software B’s byte frequency array is denoted as \( N = \{n_1, n_2, n_3, ..., n_{254}, n_{255}\} \). Except for the bytes of value 0 and 90 which are usually used as filling bytes, there are 254 classes of bytes of different values.

Like message digest (for example MD5) of a file, byte frequency can also be used to uniquely identify any software. In a 254-dimensional vector space, if assuming the point’s coordinates are the corresponding software’s byte frequency array, every point represents one software. For example, in this vector space, the point \( (m_1, m_2, m_3, ..., m_{254}, m_{255}) \) represents the software A. Generally, different software has different byte frequency.

C. The Metrics of the Proximity between Samples

Based on software’s 254-dimensional vector space, we can estimate the proximity of two files. In our model, the distance and similarity are both adopted to measure whether the selected two files hold affinity or not.

The distance indicator is the Euclidean distance of two files, \( \text{Dis}(A, B) \). Geometrically, the distance denotes the spatial distance of two vector points. Therefore, if M and N are the byte frequency vectors of file A and file B, the distance between them is calculated as:

\[
\text{Dis}(A, B) = \sqrt{\sum (m_i - n_i)^2}, \quad i \in [1, 255] \quad \text{and} \quad i \neq 90.
\]

The similarity indicator is the cosine similarity of two files, \( \text{Sim}(A, B) \), which is computed as:

\[
\text{Sim}(A, B) = \frac{\sum m_i n_i}{\sqrt{\sum m_i^2 \sum n_i^2}}, \quad i \in [1, 255] \quad \text{and} \quad i \neq 90.
\]

Euclidean coordinates, supposing the origin is O, the \( \text{Sim}(A, B) \) refers to the cosine value of the Euclidean vector \( \overrightarrow{ON} \) and \( \overrightarrow{OM} \).

For example, the byte frequency vector of the segment of codes in table 2 is \( N = (1, 0, 0, 0, ..., 1, ..., 1, ...) \). After its reordering, the byte frequency vector is \( M = (1, 0, 0, 0, ..., 1, ..., 1, ...) \). Because the reordering does not change the byte frequency, we get \( N = M \).

Therefore, \( \text{Dis}(N, M) = \sqrt{\sum (m_i - n_i)^2} = 0 \) and \( \text{Sim}(N, M) = \frac{\sum m_i n_i}{\sqrt{\sum m_i^2 \sum n_i^2}} = 1 \). Consequently, we can make sure that these two segments can be regarded as the same code.

Generally, if two files are similar with each other, the value of \( \text{Dis}(A, B) \) should be low and the value of \( \text{Sim}(A, B) \) should be high. Especially, when the byte frequencies of two files are the same, \( \text{Dis}(A, B) = 0 \) and \( \text{Sim}(A, B) = 1 \).

While implementing our byte frequency based detecting model (BFBDM), one existing malware will be picked out to be the base sample. Then, the distance and the similarity between any suspicious software and the base sample are calculated. If the suspicious and the base sample have low distance and high similarity, the suspicious is said to be a variant of the base with high probability.

The existing malwares are denoted as a set \( M \), where \( M = \{m_1, m_2, ..., m_i\} \) and \( m_i \) is a known malware. The suspicious malwares are denoted as a set \( S \), where \( S = \{s_1, s_2, ..., s_n\} \) and \( s_j \) is a suspicious malware. Then, each of \( M \) could be selected as the base sample. Thus, we can calculate \( \text{Dis}(m_i, s_j) \) and \( \text{Sim}(m_i, s_j) \) \( (i = 1, 2, ..., n; j = 1, 2, ..., m) \). Therefore, while we get a couple of known malwares, our byte frequency based detecting model can be used to determine whether a suspicious software is a variant of any existing malware or not.

In real life application, two files are regarded as holding affinity relationship when \( \text{Dis}(N, M) \leq \alpha \) and \( \text{Sim}(N, M) \geq \beta \), where \( \alpha \) is the threshold of distance and \( \beta \) is the threshold of similarity. In our following experiments, we will show how to get these thresholds.

D. The Value of Threshold

In order to attain the effective and applicable thresholds of the similarity and the distance, we do a training experiment.

In this experiment, all test samples are divided into two groups. In the first group, there are 66 samples, all of which are identified by popular anti-virus tools to be malware Trojan-GameThief.Win32.OnLineGames.ttgu. The second group consists of 538 malware samples, which are not considered as the variants of the first group, and 266 non-malware samples.

To get a threshold keeping the misjudgment rate as low as possible, we do not add any variant of Trojan-
GameThief.Win32.OnLineGames.tgu into this training set. The result is shown in the Fig. 1.

In the Fig. 1, the red points denote the samples in the first group, and the black ones denote the samples in the second group. All the red points have high similarity and low distance, while contrastively the black ones have low similarity and high distance. The points with distance higher than 20,000, which are not displayed in the Fig.1, are all the black. Distinctively, all the red points concentrate in a small range, which is close to the point (1, 0) and surrounded by the magenta dashed line.

Additively, we randomly choose 10 pairs of samples, and calculate the distance and similarity between them. The two samples in each pair have the same file size, but hold no affinity. The result is shown in the following table 3.

Thus, from the result, there is little probability that two executable files without any relationship will have short distance and high similarity.

Based upon the experiment result, we find that if $\text{Dis}(N, M) \leq 1000$ and $\text{Sim}(N, M) \geq 0.999$, the two files almost denote the same software. If $\text{Dis}(N, M) \leq 5000$ and $\text{Sim}(N, M) \geq 0.99$, the two files are suspicious to be the same. However, If $\text{Dis}(N, M) > 5000$ or $\text{Sim}(N, M) < 0.999$, there is a weak correlation between these two samples. In the following experiments, we will adopt these standards to determine if the testing samples are malware variants.

### IV. PRIMARY EXPERIMENTAL RESULTS

In following experiments, if not mentioned specially, all the malware is detected and named by Kaspersky Internet Security 2009 (KIS2009) which is updated on March 2, 2009 [13].

All the experiment samples, including the malwares and non-malwares, are not packed. So before the experiment, some of them have been unpacked using SUCOP’s VM Unpacker. This is because, as we mentioned previously, the identification of packed malware is beyond the BF BDM’s ability. We pack one random chosen executable file with UPX 3.03 with default setting, and compare the packed file with the original file. The distance is 623.928681822 and the similarity is 0.725564758941, which indicates that the byte frequency varies greatly between them.

#### A. Some Definitions

In this section, we give some useful and essential definitions in the following experiments.

**Base sample:** when comparing many samples with one sample, we call the one compared sample base sample.

Comparing two samples, the result should be one of the following three:

1. Detected (D): the BF BDM detects the two samples as the same software.
2. Suspicious (S): the two samples are suspicious to be the same.
3. Not (N): according the BF BDM, the two samples are not the same.

#### B. Experiments to Test the False Negative Rate and False Positive Rate

The basic purpose of a malware detection model is that, giving one malware sample, the model can detect out the other malware samples which have the completely same instructions with the given sample. An important and primary requirement of the malware detection model is that it must keep the false positive rate as low as possible. In other words, the detection model should never detect the non-malware as malware.

In this section, we do five experiments to testify the BF BDM’s false negative rate and false positive rate.

Firstly, in the experiment 1-1, we choose 92 samples of malware named Trojan-GameThief.Win32.Magania.amjb. All of them are different in MD5 checksum, which indicates that they all are different. If not mentioned specially, so are all the following other testing samples. We randomly choose one of them as the base sample, and

![Figure 1. Result of the Training](image-url)

(Note: the points with distance higher than 20,000 are not displayed)

#### TABLE III.

**RESULTS OF RANDOM CHOICE**

<table>
<thead>
<tr>
<th>Result</th>
<th>Minimum</th>
<th>Median</th>
<th>Maximum</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distance</td>
<td>180.616</td>
<td>1452.998</td>
<td>2850.212</td>
<td>1644.292</td>
</tr>
<tr>
<td>Similarity</td>
<td>0.484</td>
<td>0.798</td>
<td>0.920</td>
<td>0.760</td>
</tr>
</tbody>
</table>

**False negative rate:** the possibility to misjudge the malware as non-malware. Assuming there are $n$ malwares tested, and $m$ out of them are detected as non-malware, the false negative rate is $n/m$.

**False positive rate:** the possibility to misjudge the non-malware as malware. Assuming there are $m$ non-malwares tested, $d$ out of them are detected as malware, and $s$ out of them are detected as suspicious, the false positive rate is $(d+s)/m$.

**Detection rate:** the possibility to identify an unknown variant type of a known malware. Assuming there are $v$ types of malware variants, which are variants from the same one known malware, and $d$ types out of $v$ are detected as malware, the detection rate is $d/v$. 

© 2011 ACADEMY PUBLISHER
compare the others with this specific base sample. The result is shown in the table 4 and Fig. 2.

In the Fig. 2, every point indicates one sample’s distance and similarity calculated with the base one. The X-axis means the similarity, while the Y-axis means the corresponding distance between the two samples. According to the BFBDM, it seems that all the tested samples are detected as the same as the base sample malware. Therefore, the false negative rate of our novel model in this experiment is 0%.

Secondly, in experiment 1-2, we choose 199 samples of the malware Trojan-Dropper.Win32.Agent.yux. We randomly choose one of them as the base sample, compare the others with the base sample, and get the result shown in the table 4 and Fig. 2. According to the model, we can conclude that all these 199 samples belong to the same malware. The result from our model is the same as that from KIS2009. In this experiment, the false negative rate is also 0%.

Thirdly, in experiment 1-3, in order to test the BFBDM’s false positive rate, we choose 456 non-malware samples, which are randomly collected from the windows directory. The base sample is the same as that used in the experiment 1-2. Then, we compare them with the base malware sample. The experiment result is shown in the table 4 and Fig. 2. Clearly in this experiment, all the tested samples are detected as not the same as the base one. That is, all tested samples are non-malware. Therefore, the false positive rate of BFBDM in this experiment is 0%.

Fourthly, in the experiment 1-4, we calculate the distance and the similarity among one non-malware and other 2926 malware samples. In our experiment, we choose the windows edit tool, notepad.exe, as the base sample. Then we get the result shown in the table 4 and Fig. 2. Clearly, none of the malware is considered as the notepad.exe or the notepad.exe’s variant. The false positive rate of BFBDM in this experiment is also 0%.

The experiment results in 1-1 and 1-2 show that among the samples of the same malware there are short distance and high similarity. Contrastively, in the experiments 1-3 and 1-4, the distance is long and the similarity is low.

In the Fig. 2, the overlapping points of experiments 1-1 and 1-2 are concentrated in a small range, which is very close to the point (1, 0) and surrounded by the magenta dashed line, while these of experiments 1-3 and 1-4 are far from the point (1, 0).

In table 5, we summarize the four previous experiments. In all experiments in this section previously, our model has a false negative rate of 0% and a false positive rate of 0%. In this respect, the BFBDM is as effective as the traditional signatures.

Furthermore, in the experiment 1-5, we randomly choose 2,796 malware samples and 10,764 non-malware and non-exe samples. Then we calculate the distance and similarity between each non-malware and malware. The result shows that, in these 30,096,144 compares, only 136 pairs of samples are detected as holding affinity. Thus in this experiment, the false positive rate is 0.0004519%.

At last, in the experiment 1-6, on the base of the experiment 1-1, we calculate the distance and similarity between every two samples used in the experiment 1-1. The result is shown in the table 6. In the result, all the distance is shorter than 1,000, and all the similarity is higher than 0.999. Therefore, with any sample as the base sample, using the BFBDM, the other same samples can be identified with the false negative rate of 0%. In other words, in our BFBDM model, the
choice of the base sample does not influence its validity. So in real life application, any sample, of one type of malware, can be considered to be the base sample of that type. This important feature greatly improves the applicability of our new approach.

### TABLE V.
SUMMARY OF SECTION 4.2

<table>
<thead>
<tr>
<th>Exp</th>
<th>Exp. 1-1</th>
<th>Exp. 1-2</th>
<th>Exp. 1-3</th>
<th>Exp. 1-4</th>
</tr>
</thead>
<tbody>
<tr>
<td>False Negative Rate</td>
<td>0%</td>
<td>0%</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>False Positive Rate</td>
<td>N/A</td>
<td>N/A</td>
<td>0%</td>
<td>0%</td>
</tr>
</tbody>
</table>

### TABLE VI.
EXPERIMENTAL RESULTS OF 1-6

<table>
<thead>
<tr>
<th>Result</th>
<th>Minimum</th>
<th>Median</th>
<th>Maximum</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distance</td>
<td>2.8284</td>
<td>210.4804</td>
<td>293.1109</td>
<td>203.2321</td>
</tr>
<tr>
<td>Similarity</td>
<td>0.9991</td>
<td>0.9995</td>
<td>0.9995</td>
<td>0.9995</td>
</tr>
</tbody>
</table>

### C. Experiments to Verify the Ability to Detect the Malware Variants

Besides the low false negative rate and the low false positive rate, to the malware detection model, it also requires that, giving a malware, the model can detect as many variants of the given one as possible. In this section, three experiments are used to verify the BFBDM’s ability to detect the malware variants.

In experiment 2-1, on the base of the malware Trojan-Dropper.Win32.Agent.yux, the other 24 types of malware variants we collected can be detected. We estimate the distance and similarity between these variants and the base sample, and use the BFBDM to get the result shown in the table 7 and Fig. 3. In the table 7, the prefix of some malwares’ name, “Trojan-”, is ignored.

In this experiment, approximately, 98.56% of the tested samples are identified as the same as the base sample. The other 12 exceptions are detected as suspicious. That is, when we have a sample of the Trojan-Dropper.Win32.Agent.yux, all the other 24 kinds of variants can be detected out.

But this does not mean that the BFBDM can detect all the variants of the base malware, Trojan-Dropper.Win32.Agent.yux. We check the proximity between the base sample and other 40 kinds of variants of Trojan-Dropper.Win32.Agent.yux. The result shows that none of these 40 kinds can be detected base upon the Trojan-Dropper.Win32.Agent.yux.

Thus, in this case, on the scope of our data sets, the detection rate is 37.5%. In sum, base upon the known malware samples, the BFBDM can detect some variants but not all. Even so, the BFBDM is, to some extent, better than the signature based detection model.

In above experiments, all the variants are identified by popular anti-virus software. In the following two experiments, we want to test whether our model can detect some variants which are not identified by popular anti-virus software.

When we check the correlation between these two samples, we find there are short distance of 1624.61903227 and high similarity of 0.999981664416 between them. Then we check them on the website VirusTotal with 39 antivirus engines [15], for more information on March 17, 2009.

Among the 39 antivirus engines, there are 32 engines considering the former sample as malware, while 33 the latter. Ignoring the result without a certain name, such as suspicious, heuristic, generic, unknown, and so on, there are 15 engines treating them as the same malware or variants of one malware. And 7 other engines consider them as different.

According to the data from VirusTotal, it seems that these two samples should be variants of the same malware.

It’s also observed that the former sample is detected as non-malware by Eset NOD32 2.70.27, and probably a variant of Win32/Hupigon by Eset NOD32 4.0.314.0 with the same virus database of version 3941.

In experiment 2-3, we consider two special samples. One of them is considered as non-malware by the KIS2009, while the other Trojan-Dropper.Win32.Agent.zgu.

Firstly, we compute the distance and the similarity between these two softwares. Surprisingly we find out the two softwares have a strong correlation. The distance between them is 77.6337555449, and the similarity is 0.999999223745.

Then, we check them on the website VirusTotal on March 17, 2009. Three engines, AntiVir 7.9.0.116, Avast 4.8.1335.0 and McAfee-GW-Edition 6.7.6, detect them as the same malware. The AntiVir names them as TR/StartPage.fxc, Avast Win32:Agent-AEFX, and McAfee-GW-Edition Trojan.StartPage.fxc. In addition, The AVG 8.0.0.237 treats them as the same Win32/PolyCrypt, and the BitDefender 7.2 and the GData 19 name them as the same Trojan.Heur.AutoT1. None of the engines indicates that they are different.

Finally, we use the software C32ASM to disassembly these two executable files and inspect them carefully. The result shows that their assembly codes are completely the same. After computing their MD5 digest, it shows that their MD5 values are different. Therefore, it’s clear that the differences only exist in their data section.

Clearly we can confirm that these two softwares have the same structure and function. That is, these two executable files are totally the same.

In the experiments 2-1, on the base of a sample of Trojan-Dropper.Win32.Agent.yux, the BFBDM is successful to detect 24 other types of malware variants. In the experiments 2-2 and 2-3, the BFBDM also successfully detects out the malware variants, while the Eset NOD32 2.70.27 and the KIS2009 fail to identify the corresponding samples. In other words, our novel approach has more merits than current popular industrial anti-malware tools when identifying malware variants.

V. ALGORITHM COMPLEXITY

For comparing one sample with the base one, the algorithm description is shown as the following.

```
Function: calculate the distance and similarity between the base sample and the comparing sample
Input: the comparing sample’s full name
Output: the distance and similarity between the base sample and a given one
1. Open comparing sample file in binary mode
2. Read 512 more bytes in the opened file
3. While not the end of the file
  4. For each byte read
     5. dic[(int)byte] ++ = 1
  6. }
7. Read 512 more bytes in the opened file
8. }
9. For each i between 1 and 256 && i != 90{
   10. dot_matrix += basedic[i]* dic[i]
   11. dis1 += basedic[i]**2
   12. dis2 += dic[i]**2
   13. dis += (dic[i]-basedic[i])**2
   14. }
15. similarity = dot_matrix/(sqrt(dis1*dis2) )
16. distance = sqrt(dis)
```

The dictionary basedic, which stores the byte frequency information of the base sample, is prepared in advance. The dictionary dic is used to get and store the information of the comparing sample. The items of dic are set to 0 initially.

For a file with size of $n$ bytes, the step 02 and 07 will be executed by $n/512$ times. The process to get and store the byte frequency information in the step 05 will be executed by $n$ times. Constantly, for calculating the distance and similarity, each step in the loop, from the step 10 to the step 13, will be executed by 254 times. So processing a file of $n$ bytes, the asymptotic time complexity of the BFBDM is $T(n)=O(n)$.

Dealing with a file of $n$ bytes, this process needs a buffer of 512 bytes to get the file’s content, two dictionaries of 256 items to store the byte frequency information, and some bytes of constant amount to store the temporary variable and the result. So the asymptotic space complexity of this process is $S(n)=O(1)$.

In an implementation with python 3.01 of single thread, we do some experiments to test the BFBDM’s efficiency in the actual use, and get the result in the table 8.

Every round of this experiment runs three times, and the arithmetic average value of three times’ results is used as the final result. The “Read File” is the step 02 and 07 in the algorithm description, the “Get Byte Frequency” from the step 04 to the step 06, and the “Calculate” from the step 09 to the step 16. The byte frequency information of the compared files is got in advance and stored in memory, while that of the comparing files is read and counted on need.

From the result, the phase “Read File” and the phase “Get Byte Frequency”, whose efficiency is strongly determined by the I/O performance of the hard disk and memory, are the bottlenecks of the whole process. The phase “Get Byte Frequency” is also influenced by the implementation itself. And calculating the distance and similarity between two files takes only about 1.26ms averagely.
In general, the BFBDM’s algorithm complexity is low.

TABLE VIII.

PERFORMANCE OF BFBDM

<table>
<thead>
<tr>
<th>Items</th>
<th>Round</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comparing File</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Count</td>
<td></td>
<td>1 file</td>
<td>1 file</td>
<td>150 files</td>
</tr>
<tr>
<td>Total Size (bytes)</td>
<td>66,560</td>
<td>66,560</td>
<td>59,489,928</td>
<td></td>
</tr>
<tr>
<td>Average Size (bytes)</td>
<td>66,560</td>
<td>66,560</td>
<td>396,600</td>
<td></td>
</tr>
<tr>
<td>Compared File Count</td>
<td>1</td>
<td>2000</td>
<td>2000</td>
<td></td>
</tr>
<tr>
<td>Read File</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total (ms)</td>
<td>1.866</td>
<td>1.845</td>
<td>1497.719</td>
<td></td>
</tr>
<tr>
<td>Mean (ms/file)</td>
<td>1.866</td>
<td>1.845</td>
<td>9.985</td>
<td></td>
</tr>
<tr>
<td>Get Byte Frequency</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total (ms)</td>
<td>41.948</td>
<td>42.929</td>
<td>43842.543</td>
<td></td>
</tr>
<tr>
<td>Mean (ms/file)</td>
<td>41.948</td>
<td>42.929</td>
<td>292.284</td>
<td></td>
</tr>
<tr>
<td>Calculate</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total (ms)</td>
<td>1.31</td>
<td>2511.683</td>
<td>378116.144</td>
<td></td>
</tr>
<tr>
<td>Mean (ms/calculate)</td>
<td>1.31</td>
<td>1.256</td>
<td>1.26</td>
<td></td>
</tr>
</tbody>
</table>

VI. CONCLUSION

The experiments show our novel BFBDM can be used to detect the malware with a low false negative rate and a low false positive rate. And the BFBDM has the ability to detect the malware variants, to the extent where is much beyond that the traditional malware variants detection model can reach.

Further, the BFBDM can be used to identify the proximity of two executable files. During the experiments, it’s found that the stronger the correlation exists between two executable files, the shorter the distance is, and the higher the similarity is.

REFERENCES


Sheng Yu was born in Sichuan, China on June 24, 1985. In 2007 and 2010 respectively, he got the Bachelor of Engineering (B.E.) degree and Master of Engineering (M.E.) degree both in School of Computer Science and Engineering at University of Electronic Science and Technology of China (UESTC). He was major in information security in undergraduate period, and information and communication engineering in master stage. Currently, he is studying in UESTC seeking for a Ph.D degree. Since 2007, he works in the Network and Data Security Key Laboratory of Sichuan Province, Chengdu, China. His previous research interest is information security. Currently, his research interests are bioinformatics and social networking.
A Novel Timing and Frequency Synchronization Technology for OFDM System

Yi Sha, Mingming Li, Jiafu Chu
Northeast University, Shenyang, China
Email: shayi@mail.neu.edu.cn, chujiale@163.com

Abstract—Due to its high spectrum efficiency and good ability in coping with multi-path fading, orthogonal frequency division multiplexing (OFDM) technology has attracted considerable attention from researchers. However, OFDM system is very sensitive to symbol timing offset and carrier frequency offset, so accurate timing synchronization and frequency synchronization are necessary to OFDM system. This paper discusses on OFDM synchronization technology based on training symbol. OFDM synchronization algorithms based on training symbol are investigated, and two algorithms are proposed. An improved algorithm is proposed aimed at resolving training symbol redundancy and timing metric plateau in Schmidl algorithm. The training symbol in this algorithm is conjugated symmetrical and repeated. The timing metric has a sharp peak and eliminates the plateau. Both integral and fractional frequency offset estimation employ the same training symbol, and therefore the system overhead is saved and the complexity is degraded. The second algorithm proposed in this paper does not just improve symbol timing method or frequency offset estimation method. It is a complete set of solution project of OFDM synchronization. A new training symbol with special structure is designed, and a new joint symbol timing and frequency synchronization algorithm is presented based on this symbol. New frequency offset estimation algorithms are characteristic of large estimation range and high accuracy. At the same time, timing synchronization can produce a sharp peak, through which we can find where the OFDM symbol starts. The simulation results show that the performance of the synchronization algorithms proposed here is better than that of conventional methods.

Index Terms—orthogonal frequency division multiplexing, symbol timing synchronization, carrier frequency offset estimation, training symbol.

I. INTRODUCTION

OFDM is a highly efficient technology for data transmission, using mutual orthogonal subcarrier to transmit parallel data. It has good ability of resisting multi-path interference and has extremely high availability of frequency spectrum [1]. In recent years, OFDM has been deeply researched and widely applied, such as Digital Audio Broadcasting (DAB), Digital Video Broadcasting (DVB), High Definition Television (HDTV), WLAN IEEE 802.11a [2] and HIPERLAN of European Telecommunications Standards Institute etc.

OFDM symbol synchronization includes symbol timing synchronization and carrier frequency offset estimation. The symbol timing synchronization of the receiver identifies the start position of an OFDM symbol. However, as a technique of multi-carrier modulation, the system of OFDM is more sensitive to carrier frequency offset than the single-carrier system. When there is carrier frequency offset in the system, the orthogonality between subcarriers are destroyed, resulting in inter-carrier interference (ICI), then system performance deteriorates rapidly[3]; when there is timing synchronization error, phase offset and inter-symbol interference arise in the system of OFDM[4].

In recent years, several OFDM synchronization algorithms [5-9] have been proposed. In [6], in the condition of complete synchronization of symbol timing, Moose researched carrier frequency offset estimation algorithm. The sender sent two same training symbols, then the receiver used maximum likelihood estimation to estimate frequency offset. The range of frequency offset estimable is ± 1/2 subcarrier interval. Moose also proposed that increasing subcarrier interval could improve the range of frequency offset estimation, but it would decline the precision of the estimation. In [7], Schmidl used two different training symbols. Data in the two half periods of the first symbol were same, which was used for symbol timing synchronization and decimal frequency offset estimation; the second training symbol sent different pseudo-random sequences over even and odd subcarrier, which were used for integer frequency offset estimation. The timing synchronization of this algorithm exists peak platform phenomenon. Because of using two training symbols, the system’s redundancy rate is high. In [8], Morelli and Mengali (M & M) provide an improved frequency offset estimation algorithm based on the best linear unbiased estimate (BLUE) guidelines. The algorithm only needs a training symbol with L (L> 2) components of the same, and the estimated range is ± L / 2 sub-carrier intervals.

This paper proposes an improved Schmidl algorithm. It only uses the first symbol of the Schmidl training sequence, decreasing the system cost. The range of frequency offset estimation of this algorithm can reach the whole bandwidth of OFDM signal. At the same time, it eliminates the peak platform phenomenon of the
Schmidl algorithm, which can use sharp peak to precisely indicate symbol timing position.

At the same time, a new training symbol structure is proposed, being different from many classical algorithm, the training symbol has special phase rotation relationship in the frequency domain, but not repeatability in the time-domain. A new synchronization algorithm for the joint is presented for this training symbol. In the new algorithm, frequency offset estimation is characteristic of large estimation range and high accuracy, and timing synchronization can produce a sharp peak, through which we can find where the OFDM symbol starts accurately.

II. OFDM SIGNAL MODEL

In OFDM system, the baseband form of transmission signal in the sender can be expressed as:
\[
x(t) = \sum_{n=-\infty}^{\infty} \sum_{k=0}^{N-1} X_{n,k} \psi_{t,k}(t)
\]
where,
\[
\psi_{t,k}(t) = \exp\left[j 2\pi (k / T_s)(t - T_g - iT)\right] u(t - iT)
\]
\[
u(t) = \begin{cases} 1 & 0 \leq t < T_s \\
0 & \text{others}
\end{cases}
\]
\[X_{n,k}\text{ represents the } k_{th} \text{ subcarrier plural data of } \ell\text{th OFDM signal, } f_k = k / T_s \text{ represents carrier frequency of every subcarrier, } T_g \text{ represents the lengths of the protected interval, } T_s = T_u + T_g \text{ represents the duration time of one OFDM signal, } N_i = N + N_g \text{ represents the numbers of sampling points during the one period of an OFDM signal.}
\]
After discrete multipath fading channel, the output signal can be expressed as:
\[
s(t) = \sum_{i} h_i(t)x(t - \tau_i) + n(t)
\]
where, \( h_i(t) \text{ represents the gain of channel } i \text{ at time } t, \)
\( \tau_i \text{ is delay time, } L \text{ represents numbers of channel paths.}
\)
After ideal synchronization, received time zone signal is:
\[
r(t) = s(t)
\]
At the time of \( t_s = nT, T = T_u / N, \) sample the signal, the sampling points \( r(t_s) \) can be expressed as:
\[
r(t_s) = \sum_{i} h_i(nT)x(nT - \tau_i) + n(nT)
\]
Removing the protect interval, the received N sampling points of \( l_{th} \text{ OFDM symbol can be expressed as:}
\[
\tau_i = \{r_{i,0}, r_{i,1}, ..., r_{i,N-1}\}
\]
\[
r_{i,n} = r((n + N_g + IT) \cdot T)
\]
The channel is assumed as a slow fading channel, and it is quasi-stationary in the \( l_{th} \text{ OFDM signal. Let these } N \text{ sampling points go through FFT, the } k_{th} \text{ subcarrier demodulation data of } l_{th} \text{ OFDM signal can be expressed as [11]:}
\[
z_{l,k} = X_{l,k}H_{l,k} + n_{l,k}
\]
where, \( n_{l,k} \text{ is plural white Gaussian noise whose equalizing value is 0 and variance is } \sigma^2, H_{l,k} \text{ is channel frequency domain response.}
\]
\[
H_{l,k} = \sum_{i=0}^{L-1} h_i(nT) \cdot e^{-j2\pi kl/N}
\]
Neglecting the influence of the sampling clock offset, OFDM discrete signal should be re-expressed, in order to easily study carrier frequency offsets and symbol timing. In OFDM system, the baseband form of transmission signal in the sender can be expressed as [13]:
\[
x(n) = \sum_{n=0}^{N-1} X_{n,k} \exp\left(j \frac{2\pi kn(n - N_i)/N}{N}\right)
\]
where, \( n = 0, 1, ..., N+L-1, X_i \text{ represents plural data modulated to subcarrier, } N \text{ represents the numbers of the subcarrier in system, } L \text{ represents the length of the loop prefix.}
\)
The receipt signal which passed through multi-path attenuation signal path under the condition of ideal synchronization can be expressed as:
\[
s(n) = \sum_{l} h(l)x(n-l)
\]
where, \( h(l) \text{ represents the fading coefficient of the channel } l \).
In the receiver, because of delay constraint, frequency offset and Gaussian noise the receipt signal can be expressed as:
\[
r(n) = s(n-\theta)\exp\left(j \frac{2\pi \Delta f n}{N}\right) + w(n)
\]
where, \( \theta, \Delta f \text{ represent the number of delay constraint sampling points and frequency offset normalized value respectively. } w(n) \text{ represents additive white Gaussian noise, whose equalizing value and variance are 0 and } \sigma^2.
\]
III. IMPROVED SCHMIDL ALGORITHM

A. Structure of training symbol

In the sender, even subcarrier transmits real PN sequence and odd subcarrier transmits “0”. In this way, the composed symbol is not only on the N/2 conjugate symmetry in time domain, but also repetitive structure. The first half of the training symbol is the same as the last half. This unique structure can be expressed as followed [15]:

(1) Repetitive structure:
\[
x_T(n + N / 2) = x_T(n)
\]
(2) Conjugate symmetry structure:
\[
x_T(N / 2 - n) = x_T^*(N / 2 + n)
\]
where, \( x_T(n) \text{ represents the training symbol.}
\]
In this method conjugate symmetry structure was introduced to Schmidl training symbol. So, at one
hand, conjugate symmetry can be used to detect symbol timing position, at the other hand, repetitive structure is used to estimate the frequency offsets.

Training symbol structure is expressed as Figure 1 and Figure 2. Figure1 and 2 are two different performances of the one training symbol. Figure1 is the conjugate symmetry, and Figure2 is the repetitive structure.

![Figure 1. The conjugate symmetry of the training symbol](image1)

![Figure 2. The repetitive structure of the training symbol](image2)

As it is, training symbol time domain sampling points are conjugate symmetry in the sender. Though multi-paths channel, ignoring the noise, \( r_t(N/2+k)*r_t(N/2-k) \)

\[ k = 0,1,..., N/2-1 \] have the same phases, and

\[ |r_t(N/2+k)| = |r_t(N/2-k)| \]

B. Symbol Timing Algorithm

In the sender, even subcarrier transmits PN real sequences, and odd subcarrier transmits “0”. Thus, after IFFT, the timing domain samples of the training symbol obtained own properties, which can be expressed as:

\[ x_f(n+N/2) = x_f(n) \] (12)

\[ x_f(N-n) = x_f^*(n) \] (13)

Then:

\[ x_f(N/2-n) = x_f^*(N/2+n) \] (14)

So, in the receiver, the timing domain sample points of the training symbol are conjugate symmetrical distributions, whose centre is \( x_f(N/2) \). After signal path, neglecting the effect of noise, \( r_t(N/2+k)*r_t(N/2-k) \) own the same phase, and

\[ |r_t(N/2+k)| = |r_t(N/2-k)| \]. Here, \( k=0,1,...,N/2-1 \).

In the receiver, two parts of the training symbol both contain \( N/2 \) plural samples. The sum of product of paired samples can be expressed as:

\[ P(d) = \sum_{k=0}^{N/2-1} r(d+k)*r(d-k) \] (15)

The energy of \( N/2 \) samples received are as followed:

\[ R(d) = \sum_{k=0}^{N/2-1} |r(d+k)|^2 \] (16)

Here, we define timing measure as:

\[ M(d) = \frac{P(d)}{R(d)} \] (17)

where, \( d \) represents serial number of the samples received. \( M(d) \) reaches maximum when the \( N/2 \) samples come. That is to say, ignoring noise effects, \( M(d)=1 \). At other sample moments, the value of \( M(d) \) is much less than 1. This algorithm overcomes the platform phenomenon of Schmidl algorithm, so, it can obtain precise symbol timing position, as followed:

\[ \hat{d} = \arg \max_{d} \{ M(d) \} \] (18)

The estimated value of timing offset \( \hat{d} \) is as followed:

\[ \hat{\theta} = \hat{d} - L - N/2 \] (19)

C. Carrier Frequency Offsets Estimation

The carrier frequency offsets of OFDM can be expressed as:

\[ \Delta f = \Delta f_I + \Delta f_F \] (20)

where \( \Delta f_I \) represents integer carrier frequency offsets, and \( \Delta f_F \) represents fraction carrier frequency offsets. Fraction carrier frequency offsets can destroy the orthogonality of subcarrier, which can bring about ICI and bring down the system’s performance. Although integer carrier frequency offsets can’t destroy the orthogonality of subcarrier, it can change the structure of OFDM’s signal frequency spectrum, which can cause the data sequence receiver recovered cyclic shift [10]. So, it is necessary to execute integer and fraction carrier frequency offsets estimation and compensation in the receive end.

Schmidl algorithm uses two different training symbols. It uses the first training symbol to estimate fraction carrier frequency offsets. After the offsets compensated and FFT, it uses the second symbol to estimate integer carrier frequency offsets. This paper improved the Schmidl algorithm. Towards the Schmidl algorithm’s first symbol, firstly, we use the algorithm this paper proposed to estimate and compensate the integer carrier frequency offsets. Then we use Schmidl algorithm to estimate fraction carrier frequency offsets, so we can finish the frequency offsets.

After the receiver received the precise timing position, the samples of the training symbol, taking out the cyclic prefix, can be expressed as:

\[ r_t(n) = x_f(n) \exp \left( \frac{2\pi f_f(n + L + \hat{\theta})}{N} \right) + w(n + L + \hat{\theta}) \] (21)

n = 0, 1,..., N-1

Because the training sequences sent by sender is known for receiver, we can define the multiplication factor \( f(n) \) as followed:

\[ f(n) = x_f^*(n) / \|x_f(n)\| \] n = 0, 1,..., N-1 (22)

The first and second half of the training sequence are the same, so \( f(n) \) only need to calculate the first \( N/2 \) points. The last half points are the same as the first half, thus, we can save workload by 50%. Multiply \( r_t(n) \) by \( f(n) \), we can obtain as followed:

\[ r_t(n) = r_t(n) * f(n) \]
\[
\begin{align*}
\varphi(n) &= c(n) \exp \left( j \frac{2\pi \Delta f (n + N + \hat{\phi})}{N} \right) + w(n + N + \hat{\phi}) \ast f(n) \\
&= h(0)x_r(n) \exp \left( j \frac{2\pi \Delta f (n + L + \hat{\phi})}{N} \right) \varphi_r(n) \right] + \sum_{n=1}^{N-1} h(l)x_r(n-l) \exp \left( j \frac{2\pi \Delta f (n + L + \hat{\phi})}{N} \right) f(n) + w_l(n) \\
&= h(0) \exp \left( j \frac{2\pi \Delta f (n + L + \hat{\phi})}{N} \right) f(n) + w_s(n) \\
\end{align*}
\]

(23)

where, \( n=0, 1 \ldots N-1 \), \( w_1(n) \) and \( w_2(n) \) are distracters, \( w_1(n) = w(n + L + \hat{\phi}) f(n) \)

\( w_2(n) = \sum_{n=1}^{N-1} h(l)x_r(n-l) \exp \left( j \frac{2\pi \Delta f (n + L + \hat{\phi})}{N} \right) f(n) + w_l(n) \)

1) Integer Carrier Frequency Offsets Estimation

Neglecting the distracters, \( r_l(n) \) possesses only one unknown frequency, which is OFDM system’s frequency offsets. So, we can estimate OFDM system’s frequency offsets through making \( r_l(n) \) FFT [16].

\[
\Delta f_I = \begin{cases} 
\arg \max_{f_i} \{ 2I(f_i) + I(f_i,\ldots) \} & 0 \leq f_i < N / 2 \\
\arg \max_{f_i} \{ 2I(f_i) + I(f_i,\ldots) \} - N & N / 2 \leq f_i < N 
\end{cases}
\]

(24)

\[
I(f_i) = \left| \sum_{n=0}^{N-1} r(n)n \exp \left( -j \frac{2\pi f_s n}{N} \right) \right| = \left| \text{FFT}(r(n)) \right|
\]

where, \( f_k = 0, 1, \ldots N-1 \). Integer carrier frequency offsets \( \Delta f \) can be estimated by detecting the peak position of the \( 2I(f_k) + I(f_k+1) \). This algorithm is competed through FFT so that the amount of calculation is cut to the bone.

2) Fraction Carrier Frequency Offsets Estimation

After integer carrier frequency offsets estimation is finished, the normalized range of frequency offsets left is \( [0, 1) \). As well as that, in Schmidl algorithm, the range of fraction frequency offsets is \((-1, 1)\). So, we can use Schmidl algorithm to estimate the remaining fraction frequency offsets. From formula (12), we can deduce as followed (neglecting the noise):

\[
r_f(n + N / 2) = r_f(n) \exp \left( j \frac{2\pi \Delta f N / 2}{N} \right) \\
= r_f(n) \exp \left( j \pi \Delta f \right)
\]

(25)

So, fraction frequency offsets can be expressed as:

\[
\Delta f_f = \frac{1}{\pi} \sum_{n=0}^{N/2-1} r_f(n + N / 2) r_f(n)
\]

(26)

D. Simulation Results and Performance Analysis

Simulation parameters are as followed:

1) The complex-valued baseband signal \( X_k \) is selected randomly from the QPSK constellation points.

(2) Each frame of data contains one training symbol and 10 data symbols, and each symbol has 256 sampling points.

(3) Cyclic prefix contains 32 sampling points.

(4) Multi-path channel is composed of 25 paths. The delay length of each path is followed by 0, 1 ... 24 sampling points, the amplitude range of diameter 1 is \( \exp(-1/5) \) as well.

(5) Symbol timing error \( \theta = 20 \), the normalized frequency offset \( \Delta f = 3.4 \).

(6) Simulation cycles \( nLoop = 100 \).

Figure 3 is the comparison of the performance of the timing synchronization algorithm improved with that of Schmidl algorithm, when SNR is 20. The Figure obviously shows that there is platform phenomenon in Schmidl algorithm, yet the improved algorithm possesses sharp peak. So the timing synchronization performance is remarkably improved.

Figure 3. The comparison of the performance of the timing synchronization algorithm improved with that of Schmidl algorithm.

Figure 4 is the comparison of the range of frequency offset estimation of the carrier frequency offsets estimation algorithm improved with MM algorithm, when SNR is 20. MM algorithm’s training symbol includes 8 same parts. The figure shows that the normalized range of frequency offsets estimation of MM algorithm is ±4. As well, that of the algorithm is ±128, which is the signal bandwidth of OFDM.

Figure 4. The comparison of the range of frequency offset estimation of the carrier frequency offsets estimation algorithm improved with MM algorithm.
A. Symbol Timing Algorithm

Training symbols sent in the transmitter has the following properties in the frequency domain:

\[ X_{k+l} = X_k \exp \left( j \frac{4\pi k}{N} \right) \]  (27)

Then

\[ X_{N-k-l} = X_0 \exp \left( j \frac{2\pi (N-1)(N-2)}{N} \right) \]  (28)

After converting series to paralleling, inverse fast fourier transforming, the relationship between training symbols in the time domain is as follows:

\[ x(n+2) = \frac{1}{N} \sum_{k=0}^{N-2} X_k \exp \left( j \frac{2\pi k(n+2)}{N} \right) \]

\[ = \frac{1}{N} \sum_{k=0}^{N-2} X_k \exp \left( j \frac{2\pi (N-1)(N-2)}{N} \right) \]

\[ + \frac{1}{N} X_{N-1-k-l} \exp \left( j \frac{2\pi (N-1)(N-2)}{N} \right) \]

\[ = \frac{1}{N} \sum_{k=0}^{N-2} X_k \exp \left( j \frac{2\pi (N-1)(N-2)}{N} \right) \]

\[ = \frac{1}{N} \exp \left( -j \frac{2\pi n}{N} \right) \sum_{k=0}^{N-2} X_k \exp \left( j \frac{2\pi nk}{N} \right) + X_0 \]

\[ = \exp \left( -j \frac{2\pi n}{N} \right) x(n) \]  (29)

Then

\[ \|x(n+2)\| = \|x(n)\| \]  (30)

Neglecting the effect of noise, the received time-domain training sequence through the channel still has the following relationship:

\[ |r(n+2)| = |r(n)| \]  (31)

At the receiving end, so that:

\[ R(d) = \sum_{n=0}^{N-3} |r(n+d+2) + r(n+d)| \]

\[ - \sum_{n=0}^{N-3} \|r(n+d+2)\|^2 \]  (32)

The analysis shows that, when \( d \leq \theta + Ng \), \( R(d) \) is zero, when \( d > \theta + Ng \), \( R(d) \) is not zero. In order to find the critical point for further, the following operation is calculated:

\[ P(d) = \frac{|R(d+1)|}{R(d)} \]  (33)
When \( d = 0 \) + \( N_g \), \( P(d) \) has infinite value. By detecting the location of this infinite value, we can pinpoint the exact location of Symbol Timing.

\[
\hat{d} = \arg \max_d \{ P(d) \}
\]

(34)

The estimated value \( \hat{\theta} \) of timing error is:

\[
\hat{\theta} = \hat{d} - N_g
\]

(35)

**B. Carrier frequency offset estimation**

After obtaining the accurate symbol timing location at the receiving end and removing the cyclic prefix, the sampling point of the training symbol can be expressed as (ignoring noise effects):

\[
\begin{align*}
  r(n) &= s(n) \exp \left( j \frac{2 \pi \Delta f (n + N_g + \hat{\theta})}{N} \right) \\
  &= \frac{1}{N} \exp \left( j \frac{2 \pi \Delta f (n + N_g + \hat{\theta})}{N} \right) \\
  &\quad \ast \sum_{k=0}^{N-1} H_k X_k \exp \left( j \frac{2 \pi m k}{N} \right)
\end{align*}
\]

(36)

The algorithm is based on the following assumptions: adjacent Channels experience almost the same decline. That is \( H_{K+1} \approx H_K \). At the same time, we can see that \( H_0 \approx H_{N-1} \) when \( N \) is large enough.

At the receiving end, the received time-domain training sequence has the following relationship:

\[
\begin{align*}
  r(n + 2m) &= \frac{1}{N} \exp \left( j \frac{2 \pi \Delta f (n + 2m + N_g + \hat{\theta})}{N} \right) \\
  &\quad \ast \sum_{k=0}^{N-1} H_k X_k \exp \left( j \frac{2 \pi k (n + 2m)}{N} \right) \\
  &= \frac{1}{N} \exp \left( j \frac{2 \pi \Delta f (n + 2m + N_g + \hat{\theta})}{N} \right) \\
  &\quad \ast \sum_{k=0}^{N-1} H_k X_{k+1} \exp \left( j \frac{2 \pi k (n + 2m)}{N} \right) \exp \left( - j \frac{4 \pi k}{N} \right) \\
  &\quad + \frac{1}{N} \exp \left( j \frac{2 \pi \Delta f (n + 2m + N_g + \hat{\theta})}{N} \right) \\
  &\quad \ast \sum_{k=0}^{N-1} H_k X_{k+1} \exp \left( j \frac{2 \pi k (n + 2m)}{N} \right) \exp \left( - j \frac{4 \pi k}{N} \right) \\
  &= \exp \left( - j \frac{2 \pi \Delta f}{N} r(n + 2m - 2) \right) r(n + 2m - 2)
\end{align*}
\]

(37)

Thus, the carrier frequency offset of OFDM system can be expressed as:

\[
\Delta f = \frac{N}{4 \pi m} \left\{ \sum_{a=0}^{N-2m-1} r(n + 2m) r^* (n) \ast \exp \left( j \frac{2 \pi m (m - 1)}{N} \right) \exp \left( j \frac{2 \pi m m}{N} \right) \right\}
\]

(38)

When \( m = 1 \), the largest range of frequency offset estimation is \( \pm N / 4 \) sub-carrier spacing in this algorithm. When \( m \) increases, the range of frequency offset estimation becomes smaller, but its estimation accuracy becomes higher. Thus, the following offset synchronization process is presented: firstly, calculate \( \Delta f_1 \) using the formula (38), when the compensation is completed, the remaining offset ( \( \Delta f - \Delta f_1 \) ) is very small, at this time, choose a larger \( m \) and calculate \( \Delta f_m \) using the formula (38), then Carrier offset \( \Delta f = \Delta f_1 + \Delta f_m \). In this way, we can have a greater range of frequency offset estimation, but also can guarantee high estimation accuracy.

**C. Simulation results and performance analysis**

Simulation parameters are as follows:

1. The complex-valued baseband signal \( X_k \) is selected randomly from the quaternary phase shift keying (QPSK) constellation points.
2. Each frame of data contains a training symbol and 10 data symbols, and each symbol has 128 sampling points.
3. Cyclic prefix contains 32 sampling points.
4. Multi-path channel is composed of 25 paths, the delay length of each path followed by 0,1, ..., 24 sampling points, the amplitude range of diameter \( l \) is \( \exp (-l/5) \).
5. Symbol timing error \( \theta = 20 \), the normalized frequency offset \( \Delta f = 1.3 \).
6. Simulation cycles \( n \)loop = 100.

Figure 6 is timing synchronization algorithm simulation proposed in this paper when the signal to noise ratio (SNR) is 20. Simulation result shows that the algorithm can produce a sharp peak, through which we can find where the OFDM symbol starts.

Figure 7 and Fig.6 are separately the bit error rate (BER) and mean square error (MSE) comparison curves between carrier frequency offset estimation algorithm presented in this paper and MM algorithms under different SNR. Here, training symbols used in MM algorithm contain eight identical parts. From the figure we can see that the performance of this algorithm is obviously superior to MM algorithm.
Training sequences used for synchronization is generally composed of repeated sequences in order to have a large range of frequency offset estimation. At the same length of time, the more the number of repeat sequences is, the greater the estimated range is, but the worse the estimation accuracy becomes. A new training symbol structure is presented, and a new joint synchronization algorithm for the training symbols is proposed. In the new algorithm, frequency offset estimation is characteristic of high accuracy and large estimation rang, accurate symbol timing position is available, at the same time, timing synchronization can produce a sharp peak, through which we can find where the OFDM symbol starts. Simulation results demonstrate the effectiveness of the new algorithm.

V. CONCLUSION

In recent research of OFDM synchronization, generally, carrier frequency offsets estimation and symbol timing synchronization were separated. However, these two are influenced with each other under actual condition. So, we designed two effective algorithms, which consider the union estimation with carrier frequency offsets and symbol timing offsets. An improved algorithm is proposed aimed at resolving training symbol redundancy and timing metric plateau in Schmidl algorithm. Both integral and fractional frequency offset estimation employ the same training symbol, and therefore the system overhead is saved and the complexity is degraded. Secondly, a new training symbol with special structure is designed, and a new joint symbol timing and frequency synchronization algorithm is presented based on this symbol. New frequency offset estimation algorithms are characteristic of large estimation range and high accuracy.

REFERENCES


Yi Sha received the B.S. degree in radio communication from Beijing University of Aeronautics & Astronautics, China, in 1982 and the M.S. degree in digital signal processing from Northwestern Polytechnical University, China, in 1988. He is currently an assistant professor in Northeast University. His research interests include software radio application theory and RFID technology.

Mingming Li received the B.S. degree in communication engineering from Northeast University in 2007 and the M.S. degree in communication and information system in Northeast University, China, in 2009. His major research area is broadband wireless communication.

Jiafu Chu received the B.S. degree in communication engineering from Northeast University in 2009. He is currently a graduate student of Northeast University. His major research area is broadband wireless communication.
Analysis on Stability of a Network Based on RED Scheme

Shengbo Hu
Institute of Intelligent Information Processing, Guizhou Normal University, Guiyang, China
hsb@gznu.edu.cn

Heng Shu and Tiancheng Lin
Institute of Intelligent Information Processing, Guizhou Normal University, Guiyang, China
shuheng_mast@yahoo.com.cn

Abstract—The RED scheme allows to prevent global synchronization of the sources associated with drop-tail buffers. However, from a control point of view, the drop-tail discipline could lead to strong oscillations and complex behavior of the system. In this paper, the behavior of TCP in high bandwidth-delay product network is analyzed. Secondly, a model of TCP network using RED is described, including RED drop function and model of TCP source. Thirdly, the linear analysis of a single link topology is focused. Finally, a sufficient condition for the stability of a network using RED is given and an engineering approach to select network and protocols’ parameters that lead to stable operation of the linear feedback control system is presented.

Index Terms—RED; congestion; stability; control

I. INTRODUCTION

The majority of Internet routers implements drop tail queues. However, from a control point of view, the drop-tail discipline is an on-off control strategy. It is known that on-off mechanisms (also called relay controllers in the literature of feedback systems) could lead to strong oscillations and complex behavior of the system. Such oscillations lead to alternate the state of queue from emptiness to overflow. While on one hand the buffer overflow results in a waste of bandwidth due to the retransmission of lost packets, on the other hand empty queues cause the inactivity of the server and therefore the underutilization of links. Such extreme cases should be avoided and this is the reason RED (Random Early Detection)[1], one of the most popular AQM (Active Queue Management) [2][3] schemes designed to address some of the problems arising in TCP networks [4], was introduced for.

Another objective of the RED controller is to reduce as much as possible mean queuing delays and delay variations. The smaller the queue-size the less the time a packet spends in the queue and the less the end-to-end delay of a connection. However, forcing small queues may again lead to link underutilization. The queue management should be robust to variation of network parameters. These conditions include the variation in the number of TCP session sharing the bottleneck, the variation of round trip delay and the presence of uncontrolled disturb traffic on the direct path. So, a condition that guarantees stability independently of working conditions is very important.

The paper, as a whole, is organized as follows. In section 2, a model describing TCP throughput is presented. Moreover, the behavior of TCP in high bandwidth-delay product network is analyzed. In section 3, a model of TCP network using RED is described. In section 4, the linear analysis of a single link topology is focused. A sufficient condition for the stability of a network using RED is given in section 5. Finally, the paper is concluded.

II. A SINGLE CONNECTION MODEL

In this section, a model for single connection and the analytical derivation of TCP throughput are presented. Later, the behavior of TCP in high bandwidth-delay product network will be discussed.

A. A Single Connection Model

Let us consider a TCP source running over a lossy path [5] with sufficient bandwidth and sufficiently low competing traffic, so that it can be assumed that the queuing delay contribution to RTT is negligible. For ease of derivation, it is also assumed that the link introduces one drop after the successful delivery of $1/p$ consecutive packets. Under these gross assumptions, the congestion window, indicated as $cwnd$ in the following, assumes the periodic evolution depicted in Fig.1. Indicating with $W$ the maximum value of $cwnd$ reached at the equilibrium,
The throughput achieved by the connection is the ratio between the total amount of data delivered in a cycle $A_{cycle}$ and the duration of the cycle $T_{cycle}$. Substituting in the equation below, the mean throughput $B$ is as:

$$B = \frac{A_{cycle}}{T_{cycle}} = \frac{MSS \cdot b \cdot W^2}{RTT \cdot \sqrt{W}} = \frac{3}{2b} \frac{MSS}{RTT \sqrt{p}}$$  \(4\)

This expression shows that the throughput is proportional to the packet size and inversely proportional to the square root of loss probability and RTT. The constant of proportionality, that in this calculation turns out to be $\sqrt{3/2b}$, is in general a function of TCP implementation and loss model.

Nevertheless, this formulation is only an optimistic estimate of the throughput, the more accurate the lower the frequency of losses. The main reason of its inaccuracy is that it does not account for timeouts occurring at high loss rates. Indeed, when TCP meets multiple packet drops in a window of data, the Fast Recovery algorithm could not be triggered since the number of duplicated ACKs is not sufficient. Therefore, the sender loses its self-clocking capability and a timeout expires. The throughput of TCP, then, can be quite lower than the one predicted by the previous expression.

**B. Behavior of TCP in High Bandwidth-delay Product Network**

To further understand the behavior of TCP, the following simple analysis of a TCP source running on a single link that was proposed in [6] is presented.

Suppose that $c$ is the capacity of the link in packets per seconds and the $\tau$ round-trip propagation delay of the reference connection. The minimum observed round trip time $T$ is the sum of the propagation delay and transmission delay, that is $T=\tau+1/c$. The product $cT$ is known as the bandwidth-delay product, and expresses the ideal amount of data that a TCP connection should keep unacknowledged to achieve optimal path utilization. While the bandwidth-delay product generally does not play a significant role in congestion control in Local Area Network, in Wide Area Networks (WAN) or Satellite Network it is an important parameter since the round trip delay is of the same order of magnitude of buffering delay at the bottleneck. The authors of [7] pointed out that the sender may experience problems to keep the pipe full in high bandwidth-delay product networks, especially if in addition to losses for buffer overflow the connection suffers also from occasional random losses due to transient congestion.

It is assumed that at a given instant the bottleneck buffer is not empty. The packets are forwarded at rate $c$ by the link server, ACKs are generated by the destination at rate cadv, therefore, new packets can be released by the source every $1/c$ seconds. Thus, the maximum possible number of unacknowledged packets is the sum of the packets in transit across the path, which is equal to $cT$, and of the packets in the buffer $B$. Therefore, if the size of the window exceeds $W_{max}=cT+B$, a buffer overflow occurs. Actually, when the packet loss occurs, the exact size of window is difficult to evaluate, since it depends on the link capacity and on the RTT. However, as an approximation, it is assumed that TCP, after retransmitting a lost packet, resumes congestion avoidance with the cwnd set to $(cT+B)/2$.

A differential equation to describe the evolution of cwnd using fluid approximation will be derived. Let $a(t)$ denote the number of ACKs received by the source after $t$ units of time in the congestion avoidance phase. The derivative of cwnd as a function of the ACK reception rate is expressed

$$\frac{dW}{dt} = \frac{dW}{da} \frac{da}{dt}$$  \(5\)

If the window size if large enough to keep the server continuously busy, then the rate at which the ACKs are received is $c$. Otherwise, the ACK reception rate will be equal to the sending rate $W/T$. Thus,

$$\frac{da}{dt} = \min\left\{\frac{W}{T}, c\right\}$$  \(6\)

and recalling that during the congestion avoidance phase cwnd is increased by $1/W$ for each ACK received, then

$$\frac{dW}{da} = \frac{1}{W}$$  \(7\)

that is

cwnd is backed off to $W/2$ after each loss, starting a new congestion avoidance phase.

How much the cwnd opens during congestion avoidance depends on the way the receiver is acknowledging packets. The standard implementation requires TCP receivers to acknowledge upon the reception of each packet. Some implementations, however, adopt delayed-ACK, that consists of transmitting one cumulative ACK every two packets received, as a matter of fact halving the number of ACKs transmitted per round. Thus, the duration of a cycle is $bW/2rounds$, where the constant $b$ assumes the value two or one depending on the delayed-ACK being enabled or disabled respectively. The duration of a cycle turns out to be

$$T_{cycle} = RTT \cdot b \cdot \frac{W}{2}$$  \(1\)

The total number of segments delivered within each cycle is equivalent to the area under a period of the sawtooth, which is

$$W = \frac{bW^2}{2} \left(\frac{W}{2} + W\right) = \frac{bW^3}{8}$$  \(2\)

packets per cycle. Since, by hypothesis, the packets delivered in a cycle are $1/p$, $W$ can be solved

$$W = \frac{8}{3p}$$  \(3\)

The throughput achieved by the connection is the ratio between the total amount of data delivered in a cycle $A_{cycle}$ and the duration of the cycle $T_{cycle}$. Substituting in the equation below, the mean throughput $B$ is as:

$$B = \frac{A_{cycle}}{T_{cycle}} = \frac{MSS \cdot b \cdot \frac{W^2}{2}}{RTT \cdot \sqrt{W}} = \frac{3}{2b} \frac{MSS}{RTT \sqrt{p}}$$  \(4\)

This expression shows that the throughput is proportional to the packet size and inversely proportional to the square root of loss probability and RTT. The constant of proportionality, that in this calculation turns out to be $\sqrt{3/2b}$, is in general a function of TCP implementation and loss model.
\[ \frac{dW}{dt} = \begin{cases} \frac{1}{T}, & \text{if } W \leq cT \\ \frac{c}{W}, & \text{if } W > cT \end{cases} \quad (8) \]

which means that the congestion avoidance phase consists of two sub-phase corresponding to \( W \leq cT \) and \( W > cT \) respectively. It is now able to evaluate the duration of the first phase as

\[ T_1 = T \left( cT - \frac{1}{2} W_{\text{max}} \right) \quad (9) \]

and the number of packets successfully transmitted in this phase

\[ N_1 = \int_0^T \frac{W(T)}{T} \, dt = \frac{1}{2T} \left( W_{\text{max}} T_1 + \frac{T^2}{2} \right) \quad (10) \]

When \( W > cT \), the queue is increasing, the \( RTT \) increases too and hence the \( cwnd \) opens more and more slowly. By integrating (8) in the second phase,

\[ W(t)^2 = 2c(t - T_1) + (cT)^2 \quad (11) \]

and evaluating this expression for \( t = T_1 + T_2 \), the duration of the phase can be obtained

\[ T_2 = \frac{W^2_{\text{max}} - (cT)^2}{2c} \quad (12) \]

and the number of packets delivered

\[ N_2 = cT_2 \quad (13) \]

since the link is fully utilized during this phase. Having the amount of packets delivered during each phase and the duration of the phase, we can evaluate the throughput as a function of the ratio between the bottleneck buffer and the bandwidth-delay product

\[ B' = \frac{N_1 + N_2}{T_1 + T_2} = 3c \cdot \left( \frac{1 + \frac{cT}{W_{\text{max}}}}{4} \right)^2 \quad (14) \]

The previous expression, which holds for \( \frac{cT}{W_{\text{max}}} \leq 1 \), points out that the performance of a long lived connection suffers from the presence of small buffers at the bottleneck as compared to the bandwidth-delay product. It shows that, in order to fully exploit the capacity of the path, the buffer size should be at least equal to the bandwidth-delay product.

The analysis proceeds considering a scenario where, in addition to buffer overflows, each packet can be lost even after successful transmission at the bottleneck link with probability \( q \) independently of other packets. In such case, it is possible to exactly compute the throughput through Markov chain analysis. In the following this method can be sketched.

In absence of random losses, the evolution of \( cwnd \) in TCP is wholly determined by the window size at the beginning of the cycle \( w \), which is half of \( cwnd \) at the end of the previous cycle. Then, the evolution of \( cwnd \) can be described introducing the following well-defined functions:

- \( W(n,w) \) the window size after \( n \) successful packet transmissions.
- \( T(n,w) \) the time required to complete the transmission of \( n \) packets.

If the cycle terminates with losses due to buffer overflow, then \( N_{\text{max}}(w) \) represents the number of packets transmitted in this period; otherwise, the cycle terminate with a random loss after successfully transmitting \( N \leq N_{\text{max}}(w) \) packets. According to the random loss model, the distribution of \( N \)'s given by:

\[ P_r\{N = n\} = \begin{cases} q(1-q)^n, & \text{for } n < N_{\text{max}} \\ (1-q)^{n_{\text{max}}}, & \text{for } n = N_{\text{max}} \end{cases} \quad (15) \]

The window size when the cycle terminates is \( N(n,w) \) and its duration is \( T(n,w) \). For the \( i \) th cycle, let \( w_i \), \( N_i \), and \( T_i \) denote respectively the \( cwnd \) at the beginning of the congestion avoidance phase, the number of successful transmissions in the cycle, and the duration of the cycle. The evolution can be expressed through recursive equations

\[ \begin{cases} w_{i+1} = \frac{1}{2} W(N_i, w_i) \\ T_{i+1} = T_i(N_i, w_i) \end{cases} \quad (16) \]

The (16) together with (15) defines the transition probabilities for the continuous-time Markov chain of \( \{w_i\} \), whose solution gives the stationary distribution of \( w_i \). The long-run throughput is then given by

\[ B' = \frac{E[N_i]}{E[T_i]} \quad (17) \]

Since the exact solution of this Markov chain can be computationally expensive, an approximation of the previous method is also given in [7].

The key result of this analysis is that the presence of random losses leads to a significant throughput deterioration when bandwidth-delay product is high. In particular, the throughput is strongly dependent on the product of the loss probability and of the square of the bandwidth-delay product, and decrease sharply when this quantity becomes larger than some units. This phenomenon can be roughly explained by the fact that relatively earlier drops in the cycle lead to small initial values of the \( cwnd \), which implies several rounds of transmission to fill-up the pipe again. As a consequence, TCP is not suitable to run over path introducing sporadic losses, such as when the flow coexists with real-time traffic or over wireless links. In order to prevent random losses, some countermeasures, as flow isolation or ad-hoc link layer protocols, should be used.
III. A MODEL OF TCP NETWORK BASED ON RED

A. RED Drop Function

The RED scheme is showed to improve the performance of the network by reacting earlier to incipient congestion, that is introducing a proactive form of congestion control. The RED scheme allows prevent global synchronization of the sources associated with drop-tail buffers. Indeed, when a drop-tail buffer is congested, it is likely that packets of many flows would be discarded simultaneously, so inducing multiple sources to slowdown and perhaps leading to a sudden fall of instantaneous link utilization. RED, instead, similarly to many other AQM schemes, discards packets arriving at the queue randomly with a given probability which depends on the mean queue length. In other words, upon a new packet arrival, RED draws a random variable to see if the packets can be admitted to the service and computes the dropping probability on the basis of an averaged value of queue length.

It is noteworthy that RED introduces the concept of fairness in flow management, which is not present in the simpler drop-tail discipline. Indeed, if a flow is achieving more bandwidth than the others, it is more likely to have its packets dropped and therefore more signals of congestion. Then, the TCP source will decrease its rate quickly.

Some RED implementations mark a flag into the packet header instead of actually dropping the packet. The status of the flag is then copied into the returning ACK by receiver, to explicitly notify the sender the congestion. Then, the TCP source will decrease its rate quickly.

The RED scheme is showed to improve the performance of the network by reacting earlier to incipient congestion, that is introducing a proactive form of congestion control. The RED scheme allows prevent global synchronization of the sources associated with drop-tail buffers. Indeed, when a drop-tail buffer is congested, it is likely that packets of many flows would be discarded simultaneously, so inducing multiple sources to slowdown and perhaps leading to a sudden fall of instantaneous link utilization. RED, instead, similarly to many other AQM schemes, discards packets arriving at the queue randomly with a given probability which depends on the mean queue length. In other words, upon a new packet arrival, RED draws a random variable to see if the packets can be admitted to the service and computes the dropping probability on the basis of an averaged value of queue length.

It is noteworthy that RED introduces the concept of fairness in flow management, which is not present in the simpler drop-tail discipline. Indeed, if a flow is achieving more bandwidth than the others, it is more likely to have its packets dropped and therefore more signals of congestion. Then, the TCP source will decrease its rate quickly.

Some RED implementations mark a flag into the packet header instead of actually dropping the packet. The status of the flag is then copied into the returning ACK by receiver, to explicitly notify the sender the presence of congestion on direct path. This mechanism potentially prevents the flow to suffer from losses, but requires the cooperation of the two parties of the connection and the intermediate routers, which places serious problems of compatibility with existing Internet.

From a modeling point of view, the marking scheme can be treated in the same manner as dropping, since the impact of loss packets is not relevant on the overall throughput. Many variants of RED [8] have been proposed with the objective of gradually increase the dropping probability as the mean queue size increases.

Fig.2 shows the dropping profile in this paper. It is a linear function between a lower $t_{min}$ and an upper $t_{max}$ threshold with a discontinuity at $t_{max}$.

\[
p(x) = \begin{cases} 
0, & 0 \leq x < t_{min} \\
\frac{x-t_{min}}{t_{max}-t_{min}}, & t_{min} \leq x < t_{max} \\
1, & x \geq t_{max} 
\end{cases}
\]  

(18)

B. Model of Network

The network is modeled as a set of $L$ links with capacities $c_{ij}$, $l \in \{1,2,\cdots,L\}$. The links are shared by a set of $S$ sources indexed by $s \in \{1,2,\cdots,S\}$, each using a subset $L_s$ of links. The sets define $L\times S$ routing matrix

\[
A_s = \begin{cases} 
1, & \text{if } l \in L_s \\
0, & \text{otherwise}
\end{cases}
\]

(19)

which is a binary matrix where the 1s on a $l$th row indicate the sources that share the link $l$ and the 1s on sth column represent the link crossed by source $s$. Each source is associated with its congestion window $W_s(t)$ (in packets) at time $t$ and each link $l$ is associated with both its packet loss probability $p_l(t)$ (a scalar congestion measure) and with the instantaneous queue size $q_l(t)$ (in bits).

The average round trip time of the $s$th TCP source at time $t$ is approximated by

\[
RTT_s(t) = \tau_s + \sum_{l=1}^{L_s} \frac{q_l(t)}{c_l}
\]

(20)

that is the sum of the round trip delay $\tau_s$ associated with the connection and the total queuing delay of the path. Considering the losses at the different queues as independent of each other (a suitable assumption in modeling RED policy), the packet loss probability $\hat{p}_s(t)$ can be expressed as

\[
\hat{p}_s(t) = 1 - \prod_{l=1}^{L_s} (1 - A_s p_l(t)) = \sum_{l=1}^{L_s} A_s p_l(t)
\]

(21)

which corresponds to the end-to-end congestion measure for the $s$th source.

The following equation is the differential version of the Lindley equation, describing the dynamic of $l$th queue

\[
\frac{dq_l(t)}{dt} = -q_l(t)c_l + \sum_{s=1}^{S} A_s W_s(t) \frac{RTT_s(t)}{RTT_l(t)}
\]

(22)

Here, the derivative of the instantaneous queue length is the sum of two terms. The first one models the decreasing of queue length, as long as it is greater than zero, due to service of packets at a constant rate. The second term corresponds to the increasing in queue length due to the arrival of packets from the TCP flows that share the $l$th queue. Since we are interested in a mean value analysis, both sides of (18) are taken the expectation:

\[
\frac{dq_l(t)}{dt} = E[-q_l(t)c_l] + \sum_{s=1}^{S} A_s E\left[\frac{W_s(t)}{RTT_l(t)}\right]
\]
\[ \approx -E[l_{q_{i}(t)}]c_i + \sum \lambda_i \frac{\overline{W}(t)}{RTT_i(t)} \]  

In the derivation, the approximation \( E[f(x)] \approx E[x] \) has been made, which is not strictly correct and could cause errors. However, we observe that at the equilibrium the system reaches a quasi-periodic evolution (as suggested by results of detailed simulations) where the random component of state variables, such as the instantaneous queue size, makes up a smaller and smaller contribution with respect to the deterministic one as the number of flows increases. Since the system is dominated by the deterministic evolution and the extent of fluctuations is small, the previous approximation is justified.

In order to approximate the term \( E[l_{q_{i}(t)}] \), it should be considered that the bottleneck queues have \( q_{i}(t) > 0 \) with probability close to one, while the non-bottlenecked queues are typically unchanged, which means \( q_{i}(t) > 0 \) with probability close to zero. On the basis of this observation, thus

\[ \frac{d\bar{q}_i(t)}{dt} = -1 \pi c_i + \sum \lambda_i \frac{\overline{W}(t)}{RTT_i(t)} \]  

The knowledge of the mean queue length allows RED to evaluate the drop probability. It is assumed that RED estimates the average queue length using an exponential weighted moving average based on samples taken every \( T \) seconds. The smoothing filter (with \( \alpha > 0, \alpha < 1 \) as weight) is described by

\[ x_i[(k+1)T] = (1-\alpha)x_i[kT] + \alpha x_i[kT] \]  

It is useful to convert the above equation into a differential equation. Since this equation is a first order difference equation, the natural candidate

\[ \frac{dx_i}{dt} = ax_i(t) + bq_i(t) \]  

So

\[ x_i[(k+1)T] = e^{\alpha T} x_i[kT] + \int_{0}^{T} e^{\alpha (T-t)} d\mu q_i[kT] \]  

and comparing the coefficients (26) and (22), having

\[ a = -b = \frac{\ln(1-\alpha)}{T} \]  

Then, the expression (26) is rewritten, describing the behavior of \( x(t) \) by taking the expected value of both sides:

\[ \frac{\pi_i(t)}{T} = \frac{\ln(1-\alpha)}{T} (\pi_i(t) - \bar{q}_i(t)) \]  

C. Model of Source

The next step is to build a model of a TCP source. The model is based on the assumption that packet losses of a flow are described by a Poisson counting process \( \{N_i(t)\} \) with time varying rate \( \lambda_i(t) \). The rationale behind this assumption is to model losses as a flow moving from the network towards the source, rather than assuming that packets in transit across the network are discarded with a given probability. The Poisson is suitable to model the independent marking scheme commonly used in AQM/RED.

Now, if \( N_i(t) \) denotes the number of losses detected by source \( s \) at time \( t \), the following differential equation describes the evolution of the congestion window

\[ dW_i(t) = \frac{dt}{RTT_i(t)} \frac{W_i(t)}{2} dN_s \]  

This equation models the additive-increase multiplicative-decrease (AIMD) behavior of TCP. The first term corresponds to the AI part, which increases the window size by one packet every round trip time. The second term corresponds to the MD part, which halves the congestion window immediately after the drop is detected by the sender ( \( dN_i(t) = 1 \) in this case). Again, taking expectation and having

\[ dE[W_i(t)] = E\left[ \frac{dt}{RTT_i(t)} \frac{W_i(t)}{2} dN_s \right] \]  

where \( \lambda_i(t) \) is the rate of loss indication at the sender. Note that in (31) it has been assumed that the terms \( W_i(t) \) and \( dN_s \) are independent, in order to split the term \( E[W_i(t)dN_s] \) in a product of two factors \( E[W_i(t)]E[dN_s] \) . Actually, this is not exact, especially for proportional marking schemes, but it does not change the fundamental nature of TCP mechanism and allows capture the dynamic of AIMD evolution.

In proportional marking schemes (such as RED) the rate of marking/dropping indications is proportional to the share of bandwidth of the connection. That is, if the bandwidth achieved by source \( s \) is \( W_i(t)/RTT_i(t) \) the expected value for drop rate at link \( l \in L \) is

\[ p_i(t) = \frac{W_i(t)}{RTT_i(t)} \]  

However, it is noted that drops occur at the node about a round trip time before they can be detected by the sender. In order to take into account the latency of feedbacks, the rate of congestion signals (32) must be shifted forward in time of \( \tau \) seconds. Then, from (31) the evolution of \( cwnd(N_i(t)) \) is governed by

\[ \frac{d\bar{W}_i(t)}{dt} = \frac{1}{RTT_i(t)} - \frac{\bar{W}_i(t) - \bar{W}_i(t-\tau)}{2RTT_i(t)} \bar{p}_i(t-\tau) \]  

IV. LINEAR ANALYSIS OF A SINGLE LINK TOPOLOGY

In the following, let us accomplish the task of linearizing the previous set of equations in the case of a single link topology. The linearized system is suitable to be studied through the classic tools of linear control theory and gives us many suggestions on the way to modify the algorithm to fulfill the requirements of stability and robustness of the system, which in the case of RED is deemed a difficult task \[8][9].

Considering \( N \) identical TCP Reno (i.e. with the same RTT) sharing a common link with capacity \( C \), the (24) for a generic TCP flow is rewritten, defining the evolution of the mean value of \( cwnd \), and (33) concerning the dynamic of the queue.
where the term $R(t) = \tau + q(t)/C$ represents the round trip time for all connections. When writing an equation for $\dot{q}(t)$, it is assumed that the server of the queue is always transmitting packets, which is a reasonable assumption since the dynamic of the bottleneck is being studied. To complete the system of (34), to specify the relationship between the queue size and the dropping probability is needed, which depends on the employed AQM strategy. By borrowing the terminology from control-system language, the AQM control block will be referred as the controller and the rest of the system as the plant. First a linear form for the plant is determined, that, in this case, consists of the TCP flows and queue dynamics and has loss probability as input and queue size as output. Then, the design of AQM controller will be concentrated on with the support of linear control theory.

The first step to linearize the system is to find the operating point $(W_0, q_0, p_0)$, which is defined by $W = 0$ and $q = 0$. From (34), $W_0$ can be expressed

$$W_0 = \frac{1}{R(t)}R_0 = W_0/C/N$$

(35)

where $R_0 = q_0/C + \tau$. The operating point is the state-space point to which the system would converge, if the system were globally stable. This is a desirable behavior since, at the equilibrium, to each flow is allocated exactly of the available bandwidth-delay product. Note that, since (33) is used to ignore timeouts, the long term throughput $W_0/R_0$ of one connection follows the one-on-square-root-p law, which has been claimed inaccurate for high values of $p$. In this case, a term related to timeouts should be added to the right hand side of (33); this is done in [10] introducing strong assumptions (for instance, at most one packet is dropped per round) to make the model tractable.

Introducing difference variables $(\delta W, \delta q, \delta p)$, (34) can be linearized in a neighborhood of the operating point, yielding

$$\begin{align*}
\delta W(t) = & -\frac{N}{R_0C}(\delta W(t) + \delta W(t - \tau)) - \frac{R_0C^2}{2N^2}\delta p(t - \tau) \\
\delta \dot{q}(t) = & \frac{N}{R_0}\delta W(t) - \frac{1}{R_0}\delta q(t)
\end{align*}$$

(36)

To determine a simple form for the eigenvalues of the linearized system, the delay term $e^{-\tau}$ can be approximated with a constant if the condition below holds

$$W_0 \gg 1 \Rightarrow \frac{N}{R_0C} = \frac{1}{W_0R_0} \ll \frac{1}{R_0}$$

(37)

Indeed, in this case, the response-time of the aggregate of TCP flows is dominated by the term $N/RC$. Hence, it can be claimed that in an interval $\tau$ the mean window size does not vary significantly with respect to the absolute value of the window size itself and the system of equations by merging the terms $W(t)$ and $W(t - \tau)$ can be approximated

$$\begin{align*}
\delta \dot{W}(t) = & -\frac{2N}{R_0C}\delta W(t) - \frac{R_0C^2}{2N^2}\delta p(t - \tau) \\
\delta \dot{q}(t) = & \frac{N}{R_0}\delta W(t) - \frac{1}{R_0}\delta q(t)
\end{align*}$$

(38)

From the above, it is straightforward to get a block representation of the system in terms of Laplace Transform,

Figure 3. Linearized block diagram.

as shown in Fig.3, where $P_{\text{tcp}}$ and $P_{\text{queue}}$ are defined as follows:

$$P_{\text{tcp}}(s) = \frac{R_0C^2}{s^2 + \frac{2N}{R_0C}}$$

and $P_{\text{queue}}(s) = \frac{N}{s + \frac{R_0}{K}}$

(39)

Then, the Laplace representation of the plant, relating the packet-marking probability to the queue length, is given by:

$$P(s) = P_{\text{tcp}}(s)P_{\text{queue}}(s)e^{-\tau} = \frac{R_0C^2}{s^2 + \frac{2N}{R_0C}}\frac{e^{-\tau}}{(1 + s)^2 + \frac{2N}{R_0C}}$$

(40)

According to $P(s)$, the static gain is $R_0C^2/4N^2$ proportional to the round trip time and the capacity of the link and inversely proportional to the number of active flows. It influences directly the gain margin, which is the factor by which the open loop gain of a stable system should be multiplied to make the system unstable. It is not a surprise that a small number of TCP flows increases the static gain and leads to a more oscillatory response.

V. A SUFFICIENT CONDITION FOR THE STABILITY OF A NETWORK USING RED

To find a Laplace representation of AQM/RED controller, RED can be viewed as the cascade of a smoothing filter (29) and a dropping function (18). Since the operating point is between $t_{\text{min}}$ and $t_{\text{max}}$, there exists a direct linear dependence between a small variation of dropping probability $\delta p$ and a small variation of the averaged queue size $\delta q$. Then, the AQM/RED control block is described by the following transfer-function:

$$C_{\text{red}}(s) = K \frac{1}{1 + \frac{\beta}{s}}$$

(41)

where $K = -\ln(1 - \alpha)/T$, $\beta = p_{\text{max}}/(t_{\text{max}} - t_{\text{min}})$.

Note that $C_{\text{red}}(s)$ acts as a proportional controller with static gain $K$ defined by the slope of the RED dropping profile. In designing $C_{\text{red}}(s)$ to stabilize the AQM control system, variations in both the number of TCP sessions
and round-trip time should be taken into account. In this case, the variations of \( R_o \) are due to the propagation variable \( \tau \).

It is assumed that the number of TCP sessions \( N \) is larger than a threshold \( N_{\min} \) and the round trip time \( R_p \) is less than \( R_{\max} \). The goal is to select the RED parameters \( K \) and \( \beta \) that stabilize the system for all the \( N \) and \( R_p \) included in these intervals. A closed-loop control system is stable if the response to any bounded input is a bounded output. In this case we have no input, so the system is stable if the response to whatever initial condition converges exponentially to zero. The proposition is expressed as follow:

**Proposition** Let \( K \) and \( \beta \) satisfy the condition:

\[
K\frac{(R_{\max}C)^\beta}{(2N_{\min})^\beta} \leq \frac{\omega_g^2}{\beta} + 1
\]

(42)

where \( \omega_g = \frac{1}{10} \min \left\{ \frac{2N_{\min}}{(R_{\min})^2}, \frac{1}{R_{\max}} \right\} \)

(43)

then the linear system (40) using \( C_{\text{col}}(s) \) in (41) as controller is stable for all \( N \geq N_{\min} \) and \( R_p \leq R_{\max} \).

**Proof:** Let consider the frequency response of the open-loop system

\[
L_{\text{col}}(j\omega) = C_{\text{col}}(j\omega)P(j\omega) = \frac{K}{\omega_{\max}^\beta + 1} \left( \frac{j\omega}{\omega_{\max}^\beta + 1} \right)
\]

From this expression and (42),

\[
L(\omega) \approx \frac{K}{(\omega_{\max}^\beta + 1)}, \omega < \omega_g
\]

Then, given any \( N \geq N_{\min} \) and \( R_p \leq R_{\max} \),

\[
|L(\omega)| \leq \frac{K}{\omega_{\max}^\beta + 1}
\]

From this and (42) it follows that \(|L(\omega)| \leq 1\) for all \( N \geq N_{\min} \) and \( R_p \leq R_{\max} \). Thus, the unit gain crossover frequency is upper bounded by \( \omega_g \). To establish the closed loop stability, the Nyquist criterion can be invoked. It states that, if the open loop system has no unstable roots, the closed loop system is stable if the number of clockwise encirclements around the complex number \((-1+0j)\) of the trajectory of \( L(j\omega) \), is equal to zero.

\[
\angle L(j\omega_g) \geq \frac{K}{\omega_{\max}^\beta + 1} - \omega_g R_p \geq \frac{\pi}{2} - 0.1 > -\pi
\]

Then, the system is stable.

The above proposition is an example of how the linear model can be used to build a robust design of RED, which accounts for the variation of the number of flows \( N \) and of the round trip time \( R \). The rationale of this design is to force the controller to dominate the closed-loop behavior. This is done by choosing a closed loop time-constant (close to \( \omega_g \)) at least a decade higher than TCP time-constant or queue time constant. The expression (43) leaves a degree of freedom in choosing the parameters \((K, \beta)\) on the boundary of the set (42). To determine the parameters, other constraints can be placed.

**VI. CONCLUSION**

In this paper, the problem of analytically describing the interaction between TCP and AQM mechanisms has been. In particular, the attention on networks with AQM RED policy is focused. The analytical results are derived which allows a qualitative understanding of the transient behavior of TCP over RED networks. As a final result, the whole discussion in the paper is addressed to allow an engineering approach to select network and protocols’ parameters that lead to stable operation of the linear feedback control system.

**ACKNOWLEDGMENT**

The authors wish to thank Guizhou S&T Department, Guizhou Normal University. This work was supported in part by a grant from Guizhou Science Fund [(2009)2127], Guizhou Ph.D Fund and Joint Guizhou S&T Department and Guizhou Normal University Fund [(2009)11903-05022Y025].

**REFERENCES**


Shengbo Hu received the B.S degree in Information Engineering in 1992 from Southeast University, Nanjing, China and M.S degree in Communication System from Southeast University in 1997. He received the Ph.D degree in Control Engineering in 2006 from Chongqing University, Chongqing, China. In 2004, he joined Guizhou Normal University, Guiyang, China, where he is a professor. His work focuses on satellite and wireless communication networks, NGI and future mobile communication.

Heng Shu received the B.S degree in Information Engineering in 2004 from Jiangnan University, Wuxi, China and M.S degree in Computer System from Jiangnan University in 2008. In 2008, he joined Guizhou Normal University, Guiyang, China, where he is a lecturer. His work focuses on engineering analysis, simulation, modeling, and optimization in Network.

Tiancheng Lin received the B.S degree in Theory Engineering in 2004 from Guilin Electronics S&T University, Guilin, China and M.S degree in Computer System from Guilin Electronics S&T University in 2008. In 2008, he joined Guizhou Normal University, Guiyang, China, where he is a lecturer. His work focuses on engineering analysis, simulation, modeling, and optimization in Network.
Analysis and Improvement of the BAN Modified Andrew Secure RPC Protocol

Weibo Liu
Ministry of Education Key Lab of CNIS, Xidian University, Xi’an, P.R. China
Email: iamliuweibo@163.com

Wenping Ma, Yuanyuan Yang
Ministry of Education Key Lab of CNIS, Xidian University, Xi’an, P.R. China
Email: wp_ma@mail.xidian.edu.cn, yangyuan2_2001@yahoo.cn

Abstract—In this paper, We have found a new man-in-the-middle attack on the BAN modified Andrew Secure RPC protocol with a protocol model-checker based on SAT. The man-in-the-middle attack, during which an intruder can impersonate an honest agent and forge a set of messages to communicate with another honest agent, destroys the assumed authentication of the protocol, one of the important properties of security protocol. Subsequently, we have reasoned about vulnerability of the protocol and proposed a remedial method to overcome the weakness of the protocol. The method, simple and effective, can be helpful to analyze and design other security protocols.

Index Terms—model-checker, SAT, BAN modified Andrew Secure RPC, man-in-the-middle attack, a remedial method, an identifier

I. INTRODUCTION

As the popularization of Internet and rapid increase of serves based on networks, the security problem about how to protect the computer networks attracts more and more attentions. Security protocols are used to build a secure and efficacious communication channel, so their security is very critical. There are many methods [1, 2, 3, 4] to analyze the safety of security protocols, among which model checking [4] is an important and modern one. Model-checker based on SAT named automated validation tool based on SAT, one kind of model checking technologies, has the advantages of simple realization and high efficiency.

The Andrew Project was a distributed computing environment begun in 1983, driven by the Information Technology Center, a joint Carnegie Mellon University and IBM project. As one part of this project, Andrew Secure RPC protocol [5] proposed by M. Satyanarayman in 1987, is a primitive authentic protocol to provide authentication between mistrust client and server by updating the existing shared key with a new key.

However, in 1989, Burrows, Abadi and Needham firstly found the Andrew Secure RPC protocol unsafe in BAN logic method [1]. Because the message 4 in the Andrew Secure RPC protocol contains nothing that one honest agent A knows to be fresh, there is a reply attack in the protocol, and i.e. an intruder can replay this message in another session of the protocol to convince the other honest B to accept an old compromised key. To remedy the found defect of the Andrew Secure RPC protocol, an amended protocol named BAN modified Andrew Secure RPC protocol was proposed by adding a nonce into message 4 of the Andrew Secure protocol [1].

And now, we detect the BAN modified Andrew Secure RPC protocol by the automated validation tool based on SAT, and find that the modified protocol still suffers from a new man-in-the-middle attack. The intruder can impersonate an honest agent B and forge a set of messages to communicate with the honest agent A to deceive A, which destroys the assumed authentication of the protocol. We analyze the attack in the BAN modified Andrew Secure RPC protocol and propose a new improved protocol by adding an identifier. Then we detect the improved protocol by the automated validation tool based on SAT, and find the protocol is safe. At last, according to comparing with other similar protocols which are vulnerable to man-in-the-middle attack, we find that adding an honest agent identifier is efficacious to prevent the similar protocols from the attack.

The rest of this paper is organized as follows. Section II makes an introduction to the model-checker based on SAT, which is an efficient and trusted tool in the detection of the security protocol. Section III represents the detecting model of BAN modified Andrew Secure RPC protocol, in which, the protocol interactions, the initial state and the goal of the protocol are transferred into rewriting rules. And then the defect of the protocol is shown in section IV. Consequently, in section V, we reason about the founded man-in-the-middle attack, propose an amendatory BAN modified Andrew Secure RPC protocol and make contrasts with other similar protocols. Finally, the paper is concluded in section VI.

Supported by National Science Foundation of China under Grant No.60773002863, 863 Program under Grant No.2007AA01Z472, The 111 Project No.B08038.
Corresponding author: Weibo Liu.

© 2011 ACADEMY PUBLISHER
doi:10.4304/jnw.6.4.662-669
II. PROTOCOL MODEL-CHECKER BASED ON SAT

Model checking is a good and important method for formally verifying finite-state concurrent systems [4]. Specifications about the system are expressed as temporal logic formulas, and efficient symbolic algorithms are used to traverse the model defined by the system and check if the specification holds or not. With the rapid development of electronic and information technology, computers become more and more powerful. Model checking methods, whose accomplishment relies on the computer, also develop speedily. Now, the technique has been applied into many complex systems, and protocol verification is one of the applications.

Protocol model-checking is an important method to analyze security protocols, which is easier to be automated and programmed by computer. And there are many detecting tools based on protocol model-checking. Automated validation of security protocols based on SAT [6] is one kind of protocol model-checkers, which is one of the integrated tools in AVISPA (Automated Validation of Internet Security Protocols and Applications) project [7] sponsored by European Union to develop a push-button, industrial-strength technology for the analysis of large-scale internet security-sensitive protocols and applications.

In the process of automated detecting security protocol based on SAT, the states of honest agents and intruder are firstly described into a series of rewriting rules. To be more detailed, the actions of the honest agents are described according to the protocol’s instructions into rewriting rules, while the actions of the intruder are described according to the D-Y model [8]. By means of rewriting rules, the protocol and the secure goals are transformed into PIP (Protocol Insecurity Problem). In PIP, if there is a combination of rewriting rules of honest agents and intruder which can transform the initial states to the goal states, the protocol defect exists, and contrarily the protocol is safe. The process of seeking out the combination of rewriting rules, which is called AI (Artificial Intelligent) planning problem, can be altered into proposition formula. Then the formula can be further transformed into CNF (Conjunction Normal Form). And now we can use available sophisticated algorithms [9, 10, 11] to solve the CNF. If the proposition formula has the true values, the protocol has attack and the attack trace is returned, otherwise the protocol is safe [6]. The overall approach of the automated detecting security protocol based on SAT is described in Figure 1.

A. Modelling security protocols into planning problems

In the process of the detecting security protocol, we assume that the cryptography used in the security protocol is perfect and interaction message type of the protocol is strong, i.e. agents only accept type-correct messages and therefore type confusion is not allowed.

A protocol insecurity problem is a reachability problem \( \Xi = (\Gamma_H \circ \Gamma_I, B) \), where \( \Gamma_H \) and \( \Gamma_I \) are multi-set rewriting system, which specify the actions of the honest agents and the intruder respectively, and \( B \) is a set of states. The reachability of any state in \( B \) implies a violation of a security goal that the security protocol intends to guarantee. The more detailed translating rules of honest agent, intruder, initial state and protocol goal are introduced in section III.

Once the protocol is transformed into protocol insecurity problem, we convert the protocol insecurity problem into planning problem. The ideas of the two transformations are similar, which externalize a state by means of a set of atomic formulae of a first-order language as well as the feature of using a rule-based language for specifying the transition relation.

Figure 1. The overall approach of the automated detecting security protocol based on SAT.

B. Planning via SAT

Planning problem can be represented by the following formula until there is an assignment to satisfy the formula or the \( k \) reaches the maximum value. The process consists of encoding into formula and solving by some SAT algorithm.

\[
\left[ \prod_{i=1}^{k} \right] = I(f^0) \land \bigwedge_{i=1}^{k-1} (f^i, d^i, f^{i+1}) \land B(f^k)
\]

In the formula, \( k \) is the current encoding step that labels transformation into proposition formula of the planning system; \( I(f^0) \) means the initial states of honest agents and intruder; \( f^i \) represents the set of the state at step \( i \); \( d^i \) is action set and includes the actions of honest agents as well as the intruder; \( f^{i+1} \) is the subsequence of the actions \( a^i \) on \( f^i \), and \( B(f^k) \) represents the bad states of the protocol. Generally, planning via SAT uses linear encodings [18] or graph-based encoding [19].

At each step \( i \), the reachability of \( B(f^k) \) will be checked. If one bad state of \( B(f^k) \) is in \( f^{i+1} \), the goal is reachable, and the action sequence from initial states to the goal will be returned. And if there is no reachable bad state of \( B(f^k) \) until \( k \) reaches maximum value, the empty action sequence will be returned.
C. Retrieving Attack

Finally, according to the returned action sequence, we can judge whether the protocol safe or not. If the action sequence is not empty, we can get the attack according to the partial-order plan of the returned action sequence. And if the action sequence is empty, there is no violation to the protocol.

III. BAN MODIFIED ANDREW SECURE RPC PROTOCOL

As mentioned earlier, BAN modified Andrew Secure RPC protocol is an enhanced form based on Andrew Secure RPC protocol. The BAN modified Andrew Secure RPC protocol is intended to create a new session key to be used in the following communication, which can be known only by the honest agents participating the handshake. In addition, the protocol must guarantee the authenticity of the new shared session key in every session. The detailed interactions of the honest agents are described as follows:

where, $A$ and $B$ are honest agents, $Na$ and $Nb$ are nonces, $Kab'$ denotes the new session key generated by $B$, and $Nb'$ is an initial identifier used for the following communication.

As shown in Figure 2, the executing process can be divided into the following five phases.

- In sponsor phase, honest agent $A$ sponsors the session, generates a new nonce $Na$ and sends his identity and encrypted nonce. After having sponsored the session, $A$ is waiting the responder to verify the initial session.
- In respond phase, having received the message from $A$, the responder $B$ obtains the sender’s information according to the built-in identity. And then, $B$ generates a new nonce $Nb$, combines the nonce $Na$ with $Nb$, encrypts the combined message with the session key $Kab$, and send the encrypted message to $A$. Now $B$ has replied to the sponsor, and launches a new challenge with $Nb$.
- In authentication phase, $A$ receives the message from $B$, verify the identity of $B$, and accepts the new challenge. Then he encrypts the challenge and sends it to $B$, in order to make $B$ confirm his identity.
- In negotiation phase, $B$ receive the encrypt message, confirms the identity of $A$. Until now, the authentication between each other has finished. Then, $B$ generates a new session key $Kab'$ and a new nonce $Nb'$, combines the both with $Na$, encrypts the combined message with the former session key $Kab$, and sends it to $A$.
- In verification phase, $A$ decrypts the message from $B$, accepts the new session key $Kab'$ and the new nonce $Nb'$ of $B$. Until now, the overall interactions of the protocol have completely successfully.

The formal execution of the protocol should convince $A$ that he has talked with $B$, and they confer on the new shared session key $Kab'$. The protocol must guaranty the secrecy and the authenticity of $Kab'$. In addition, the new nonce $Nb'$ should be shared only by the honest agents, i.e. $A$ and $B$.

A. Rewrite Rules of Honest Agents

The multi-set rewriting system $\Gamma_H = \{ F_H, L_H, I_H, R_H \}$ models the behavior of the honest agents specified by the protocol interaction process, where $F_H$ is the fact of the protocol, such as the principal fact, sending messages, session number, step number, numeric value, encrypt key and so on; $L_H$ represents the label of the rewriting rules, which labels the transforming step; $I_H$ is the initial states of the honest agents and $R_H$ is the overall rewriting rules of the honest agent.

Firstly, we transform the instructions of the BAN modified Andrew Secure RPC protocol into rewriting rules of the honest agents [6], which are shown as below. The four messages in the interactions of the protocol are abbreviated to $mes_1, mes_2, mes_3$ and $mes_4$ in the following rewriting rules. And the five rewriting rules are consistent with the mentioned five phased.

(R1) $\text{state}(0, A, A, [A, B, Kab], Se)$

\[ \text{step}(1, [A, B, Kab], Se) \rightarrow \exists Na : \]

\[ \text{state}(2, B, A, [A, B, Na, Kab], Se) \cdot \]

\[ \text{msg}(1, A, B, mes_1) \cdot \]

\[ \text{witness}(A, B, na, Na, Se) \]

(R2) $\text{state}(1, A, B, [A, B, Kab], Se)$

\[ \text{step}(2, [A, B, Kab, Na, Nb], Se) \rightarrow \exists Nb \]

\[ \text{state}(3, A, B, [A, B, Kab, Na, Nb], Se) \cdot \]

\[ \text{msg}(2, B, A, mes_2) \cdot \]

\[ \text{request}(B, A, na, Na, Se) \cdot \]

\[ \text{witness}(B, A, nb, Nb, Se) \]
(R3) state(2, B, A, [A, B, Na, Kab], Se) ⋮
\[\text{msg}(2, B, A, \text{mes}2)\]
\[\text{step}(4, A, B, \text{Kab}, \text{Na}, \text{Nb}, \text{Se})\]
\[\text{state}(4, A, B, [A, B, \text{Kab}, \text{Na}, \text{Nb}], \text{Se})\]
\[\text{msg}(3, A, B, \text{mes}3)\]
\[\text{request}(A, B, \text{nb}, \text{Nb}, \text{Se})\]

(R4) state(3, A, B, [A, B, Kab, Na, Nb], Se) ⋮
\[\text{msg}(3, A, B, \text{mes}3)\]
\[\text{step}(4, A, B, \text{Kab}, \text{Na}, \text{Nb}, \text{Se})\]
\[\exists \text{Nb}^{'}, \text{Kab}^{'}\]
\[\text{state}(5, A, B, [A, B, \text{Kab}, \text{Kab}^{'}, \text{Na}, \text{Nb}, \text{Nb}^{'}, \text{Se})]\]
\[\text{msg}(4, B, A, \text{mes}4)\]
\[\text{witness}(B, A, k, \text{Kab}^{'}, \text{Se})\]
\[\text{witness}(B, A, \text{nb}', \text{Nb}^{'}, \text{Se})\]

(R5) state(4, B, A, [A, B, Kab, Na, Nb], Se) ⋮
\[\text{msg}(4, B, A, \text{mes}4)\]
\[\text{step}(5, A, B, \text{Kab}, \text{Na}, \text{Nb}, \text{Nb}', \text{Se})\]
\[\text{state}(6, B, A, [A, B, \text{Kab}, \text{Kab}^{'}, \text{Na}, \text{Nb}, \text{Nb}^{'}, \text{Se})]\]
\[\text{request}(A, B, k, \text{Kab}^{'}, \text{Se})\]
\[\text{request}(A, B, \text{nb}', \text{Nb}^{'}, \text{Se})\]

where \(\text{msg}(J, P, Q, M)\) represents that agent \(P\) sends message \(M\) to agent \(Q\) at step \(J\). \(\text{state}(J, P, Q, [T_1, T_2, \cdots, T_k], \text{Se})\) means that when \(P \neq Q\) holds, \(Q\) is waiting for a message from \(P\) at step \(J\) of session \(\text{Se}\) while \(P = Q\). \(P\) responds a handshake at step \(J\) of session \(\text{Se}\), where \(T_1, T_2, \cdots, T_k\) are the knowledge of the agent \(Q\) at step \(J\) of session \(\text{Se}\). In particular, \(\text{witness}(P, Q, O, M, \text{Se})\) means that the honest agent \(P\) wants to execute the protocol with \(Q\) by using \(M\) as the value for the authentication identifier \(O\) in session \(\text{Se}\), whereas \(\text{request}(Q, P, O, M, \text{Se})\) means that agent \(Q\) accepts the value \(M\) and now relies on the guarantee that agent \(P\) exists and agrees with \(Q\) on this value for the authentication identifier \(O\) in session \(\text{Se}\). Thus, the situation in which \(\text{request}(Q, P, O, M, \text{Se})\) holds and the corresponding \(\text{witness}(P, Q, O, M, \text{Se})\) does not hold represents a violation of an authentication property. \(\text{step}(i, \ldots, \text{Se})\) is the \(i\)-th action label, which represents the uniqueness of action and rewriting rules.

The symbols such as \(A, B, \text{Kab}, Na, Nb, \text{Kab}\), which are written in capital letters or whose first letter is written in capital letter, represent free variables. When the protocol is checked step by step, the variables will be instantiated by concrete values.

B. The Behaviors of the Intruder

The multi-set rewriting system \(\Gamma_i = \{F_i, L_i, I_i, R_i\}\) models the behaviors of the intruder, where the meaning of \(F_i, L_i, I_i, R_i\) and \(R_i\) is similar with \(F_H, L_H, I_H, R_H\) as mentioned, only different in the body of the rewriting rules.

Contrary to the specified executing of the honest agents as the protocol, the intruder has many degrees of freedom. According to the D-Y model [2], the intruder can control the whole communication network, and he can intercept any message in the network, compose and send fraudulent messages to honest agents and so on. The exhaustive behaviors of the intruder can also be expressed in the form of rewrite rules [6].

(R6) \(\text{msg}(I, A, B, M) \rightarrow_{\text{divert}(A,B,I,M)} ik(M)\)

(R7) \(\text{ik}(M) \rightarrow \text{ik}(K) \rightarrow_{\text{encrypt}(K,M)} \text{ik}(M) \cdot \text{ik}(K) \cdot \text{ik}(\{M\}_k)\)

(R8) \(\text{ik}(M) \rightarrow \text{ik}(N) \rightarrow_{\text{pairing}(M,N)} \text{ik}(M) \cdot \text{ik}(N)\)

(R9) \(\text{ik}(M,N) \rightarrow_{\text{decompose}(M_1,M_2)} \text{ik}(M) \cdot \text{ik}(N)\)

(R10) \(\text{ik}(\{M\}_k) \rightarrow \text{ik}(K^{-1}) \rightarrow_{\text{decrypt}(K,M)} \text{ik}(\{M\}_k) \cdot \text{ik}(K^{-1}) \cdot \text{ik}(M)\)

(R11) \(\text{ik}(M) \rightarrow \text{ik}(A) \cdot \text{ik}(B) \rightarrow_{\text{fake}(A,B,M,1)} \text{ik}(M) \cdot \text{ik}(A) \cdot \text{ik}(B) \cdot \text{msg}(I, A, B, M)\)

where, (R6) states that, if a message has been sent out on the communication channel, the intruder can intercept the message. The rewriting rules (R7), (R8), (R9) and (R10) specify how the intruder performs encryption, pairing, decomposition, decryption of message, respectively. The rule (R11) represents that the intruder can send arbitrary messages possibly faking somebody else’s identity to any other agent. The behaviors above can conclude the ability of the intruder basically.

In the process of protocol executing, the intruder uses any one or the combination of the six abilities to destroy the normal interaction of the honest agents. Once the intruder obtains the message that the honest agents do not want the intruder to known or the intruder is successful to fake one honest agent to cheat other honest agent, the protocol is vulnerable to attacks. The former can destroy the secretive privacy of the protocol, and the later one destroys the authentication of the protocol. The both are important properties of security protocols.

C. Initial States

The initial states of the intruder and honest agents are completely described by their knowledge. The knowledge of the intruder is modeled by facts of the form \(\text{ik}(M)\) that the intruder knows message \(M\). The initial states of the intruder on the BAN modified Andrew Secure RPC protocol are represented by \(\text{ik}(a) \cdot \text{ik}(b) \cdot \text{ik}(i)\), where \(a\) and \(b\) are the honest agents participating the handshake, and \(i\) is the intruder who wants to destroy the communication between honest agents.

The initial states of the honest agents on the protocol involve two concurrent sessions.
state(0, a, [a, b, kab], 1) \hspace{1cm} (A.1)
state(1, a, b, [a, b, kab], 1) \hspace{1cm} (A.2)
state(0, b, b, [a, b, kab], 2) \hspace{1cm} (A.3)
state(1, b, a, [a, b, kab], 2) \hspace{1cm} (A.4)

(A.1) represents the initial state of $a$ in session 1 in which $a$ plays the role of initiator; (A.2) represents state of $a$ who is waiting for the responding message from $b$ in session 1; (A.3) represents the initial state of the agent $b$ as an initiator in session 2; and (A.4) represents state of $b$ who is waiting for the responding message from $a$ in session 2.

D. Goals

Bad states are the states whose reachability implies a violation of a secure property which the protocol seeks to guarantee. If $M$ is a secret shared by the honest agents, the reachability of the state $ik(M)$ means that the secret property is destroyed, i.e. the intruder knows a secret message $M$ that he should not know. Furthermore, the reachability of the fact request($Q, P, O, M, Se$) but not containing the corresponding the fact witness($P, Q, O, M, Se$) denotes the authentication property of the protocol is destroyed, and the intruder maybe fake an honest agent to trespass on the formal interactions between honest agents.

Firstly, the BAN modified Andrew Secure RPC protocol wants to negotiate a new session key $kab'$, which should be secret to the intruder. And in the negotiation process, the honest agents use the old session key $kab$, the nonces $Na, Nb$ and $Nb'$, which should also be known by the intruder. The bad privacy goals of the BAN modified Andrew Secure RPC protocol are represented in (B.1) to (B.5).

\begin{align*}
\text{ik}(kab) \quad & \quad (B.1) \\
\text{ik}(kab') \quad & \quad (B.2) \\
\text{ik}(Na) \quad & \quad (B.3) \\
\text{ik}(Nb) \quad & \quad (B.4) \\
\text{ik}(Nb') \quad & \quad (B.5) 
\end{align*}

Not only the privacy, the BAN modified Andrew Secure RPC protocol should also guarantee the authentication between $a$ and $b$ to make sure the protocol executes formally. The bad authentication goals are shown in (C.1) to (C.8).

\begin{align*}
\text{request}(a, b, k, kab, 1) & \land \neg \text{witness}(b, a, k, kab, 1) \quad (C.1) \\
\text{request}(a, b, k, kab, 2) & \land \neg \text{witness}(b, a, k, kab, 2) \quad (C.2) \\
\text{request}(b, a, na, Na, 1) & \land \neg \text{witness}(a, b, k, Na, 1) \quad (C.3) \\
\text{request}(b, a, na, Na, 2) & \land \neg \text{witness}(a, b, na, Na, 2) \quad (C.4) \\
\text{request}(a, b, nb, Nb, 1) & \land \neg \text{witness}(b, a, nb, Nb, 1) \quad (C.5) \\
\text{request}(a, b, nb, Nb, 2) & \land \neg \text{witness}(b, a, nb, Nb, 2) \quad (C.6) \\
\text{request}(a, b, nb', Nb', 1) & \land \neg \text{witness}(b, a, nb', Nb', 1) \quad (C.7) \\
\text{request}(a, b, nb', Nb', 2) & \land \neg \text{witness}(b, a, nb', Nb', 2) \quad (C.8)
\end{align*}

The reachability of any one among the mentioned bad goals means that the protocol is unsafe.

IV. DETECTING THE BAN MODIFIED ANDREW SECURE RPC PROTOCOL

In the following, we uses the automated validation tool based on SAT to detect the BAN modified Andrew Secure RPC protocol. In detecting result tables, “summary” states whether the protocol is safe or not; “Total time” denotes the time cost by the detecting process; “Number_Atom” and “Number_CNF” represent the number of the variables and CNF in the proposition formula; “Time_encoding” and “Time_solving” signify the spending time in encoding and solving proposition formula respectively.

A. Detecting privacy property

We detect the mentioned privacy bad goals (B.1) to (B.5) and get the detailed detecting results as illustrated in Table I.

<table>
<thead>
<tr>
<th>Goal</th>
<th>(B.1)</th>
<th>(B.2)</th>
<th>(B.3)</th>
<th>(B.4)</th>
<th>(B.5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Summary</td>
<td>Safe</td>
<td>Safe</td>
<td>Safe</td>
<td>Safe</td>
<td>Safe</td>
</tr>
<tr>
<td>Total time</td>
<td>0.016s</td>
<td>0.0s</td>
<td>0.015s</td>
<td>0.0s</td>
<td>0.0s</td>
</tr>
<tr>
<td>Number_Atom</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Number_CNF</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Time_encoding</td>
<td>0.0s</td>
<td>0.0s</td>
<td>0.0s</td>
<td>0.0s</td>
<td>0.0s</td>
</tr>
<tr>
<td>Time_solving</td>
<td>0.0s</td>
<td>0.0s</td>
<td>0.0s</td>
<td>0.0s</td>
<td>0.0s</td>
</tr>
</tbody>
</table>

From Table I, we know the privacy bad states of goal (B.1) to goal (B.5) are unreachable, i.e. the protocol is safe in protecting the secret messages.

B. Detecting authentication property

We also detect the authentication bad goals (C.1) to (C.8) using the protocol model-checker based on SAT. Table II shows the detecting results in authentication goals.

<table>
<thead>
<tr>
<th>Goal</th>
<th>(C.1)</th>
<th>(C.2)</th>
<th>(C.3)</th>
<th>(C.4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Summary</td>
<td>Attack</td>
<td>Safe</td>
<td>Safe</td>
<td>Safe</td>
</tr>
<tr>
<td>Total time</td>
<td>842.141s</td>
<td>0.0s</td>
<td>0.0s</td>
<td>0.015s</td>
</tr>
<tr>
<td>Number_Atom</td>
<td>553</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Number_CNF</td>
<td>616</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Time_encoding</td>
<td>0.0s</td>
<td>0.0s</td>
<td>0.0s</td>
<td>0.0s</td>
</tr>
<tr>
<td>Time_solving</td>
<td>829.953s</td>
<td>0.0s</td>
<td>0.0s</td>
<td>0.0s</td>
</tr>
</tbody>
</table>

From Table II, we know the authentication bad states of goal (C.1) to goal (C.8) are unreachable, i.e. the protocol is safe in authentication.
From Table II, we know that the bad goal (C.1) reaches and other goals do not, i.e. the goal (C.1) is unsafe and other goals are safe. The reachability of goal (C.1) means that \( a \) has accepted the new session key \( kab \) proposed by \( b \) in session 1, however, \( b \) does not launch the suggestion that they use \( kab \) as their new session key in session 1. Therefore, the authentication of the protocol is destroyed and the protocol is vulnerable to the intruder.

C. The retrieving attack trace

According to the returned attack path to the goal (C.3), we find a new man-in-the-middle attack on the BAN modified Andrew Secure RPC protocol, whose exhaustive attack steps are shown as below.

1. \( A \rightarrow i(B) \): \( A \{ Na \}_Kab \)
2. \( i(B) \rightarrow A \): \( A \{ Na \}_Kab \)
3. \( A \rightarrow i(B) \): \( \{ Na + 1, Na' \}_Kab \)
4. \( i(B) \rightarrow A \): \( \{ Na + 1, Na' \}_Kab \)
5. \( A \rightarrow i(B) \): \( \{ Na' + 1 \}_Kab \)
6. \( i(B) \rightarrow A \): \( \{ Na' + 1 \}_Kab \)
7. \( A \rightarrow i(B) \): \( \{ Kab', Na', Na \}_Kab \)
8. \( i(B) \rightarrow A \): \( \{ Kab', Na', Na \}_Kab \)

where \( i, j \) denotes the executing step \( j \) in session \( i \). \( i(B) \) denotes that the intruder \( i \) fakes the honest agent \( B \). Session 1 is the interaction between the honest agent \( A \) and the intruder \( i \), while session 2 is the interaction between the intruder \( i \) faking the honest agent \( B \) with \( A \).

In the light of exhaustive attack traces, we can get that the intruder \( i \) deceives \( A \) into believing \( A \) is talking with \( B \), whereas in the fact, \( A \) is talking with \( i \), not \( B \). Firstly, \( A \) starts the protocol with the message 1.1; The intruder \( i \) intercepts the message, and starts a second session with \( A \) by sending the message 2.1; \( A \) receives the message faked by \( i \), considers that she has another session with the honest agent \( B \), and sends back message 2.2; \( i \) intercepts the message 2.2 and sends message 1.2 to \( A \); considering the message 1.2 as the feedback in session 1, \( A \) sends out the message 1.3; as well, the message 1.3 is intercepted by \( i \) and is sent to \( A \) in message 2.3; confirming the message 2.3, \( A \) sends message 2.4 to \( B \); intercepting the message 2.4, \( i \) sends the message to \( A \) in message 1.4.

Until now, both the session are completed and it is successful for the intruder \( i \) to fake the honest agent \( B \) to deceive the honest agent \( A \), and the authentication of the protocol which is expected to guarantee breaks.

V. THE AMENDATORY FORM OF BAN MODIFIED ANDREW SECURE RPC PROTOCOL

We analyze the found defect and get the key point to the attack. The intruder institutes the identity in message 1 and then transforms the message 2 to honest agent \( A \), because there is no identity to label the message sender, the intruder succeeds to fake honest agent \( B \) to cheat \( A \) to complete the two different sessions.

In order to protect the BAN modified Andrew Secure RPC protocol from being attacked utilizing the found defect, we can add \( B \)'s identifier to the sending message in the second step. The identifier guarantees that the second message is sent out by the honest agent \( B \). As a result, the intruder cannot substitute the identifier to achieve the attack goal (C.1) and other goals. The following Figure 3 demonstrates the interactions of amendatory BAN modified Andrew Secure RPC protocol.

![Figure 3. The interactions of amendatory BAN modified Andrew Secure RPC protocol.](image-url)

In order to check the new amendatory protocol, we also detect the new protocol by the automated validation tool based on SAT. As mentioned in section III, the amendatory BAN modified Andrew Secure RPC protocol should also be described with rewriting rules, initial states and bad goals. And the initial states and bad goal of the new protocol are complete accord with the BAN modified Andrew Secure RPC protocol. There is only a small change in \( (R2) \), which is labeled by \( (R2') \). We assume that he four messages in the interactions of the new amendatory protocol are abbreviated to \( Mes1, Mes2, Mes3 \) and \( Mes4 \).

\[ (R2)\]

\[
\text{state}(1, A, B, [A, B, Kab], Se) \cdot
\text{msg}(1, A, B, mes1) \cdot
\text{step}(2, [A, B, Kab, Na, Nb], Se) \Rightarrow \exists Nb
\]

\[
\text{state}(3, A, B, [A, B, Kab, Na, Nb], Se) \cdot
\text{msg}(2, B, A, Mes2) \cdot
\text{request}(B, A, na, Na, Se) \cdot
\text{witness}(B, A, Nb, Na, Se)
\]

And then, we use automated validation of security protocols based on SAT to detect the amendatory BAN modified Andrew Secure RPC protocol, and find it is safe and robust. The detailed result in privacy and authentication properties is illustrated in Table III and IV separately.
Finally, we contrast many different protocols which is vulnerable to man-in-the-middle attack, such as SPLICE/AS protocol [12], NSPK protocol [13] and Helsinki protocol [14], which have the same defect in authentication as BAN modified Andrew Secure RPC protocol. Take NSPK protocol for example, the Alice-Bob instructions of NSPK protocol are described as below:

1) $A \rightarrow B: \{ A, Na \}_{K_B}$
2) $B \rightarrow A: \{ Na, Nb \}_{K_A}$
3) $A \rightarrow B: \{ Nb \}_{K_B}$

There is a man-in-the-middle attack on the above NSPK protocol. In 1995, Gavin Low fixed the defect of the protocol [13] by adding the sender’s identity in message 2), and proposed NSPK-fix protocol to prevent the attack, whose criterions are presented as follows:

1) $A \rightarrow B: \{ A, Na \}_{K_B}$
2) $B \rightarrow A: \{ B, Na, Nb \}_{K_a}$
3) $A \rightarrow B: \{ Nb \}_{K_B}$

Similarly, in other protocols which are vulnerable to man-in-the-middle attack, we find the enhanced measurements to remedy the defect are uniform. It is by adding the identifier of the sender to mark the sending message that the intruder can not fake one honest agent to deceive another honest agent. Therefore, it is essential that there be some a message in the sending message which can identify the sender's identity, which is of great importance in the analysis and design of the security protocols to defend the man-in-the-middle attack.

VI. CONCLUSION

In the paper, we present a new man-in-the-middle attack on the BAN modified Andrew Secure RPC protocol, and then propose an amendatory protocol by adding an identifier to overcome the found defect. Finally we take a comparison with some protocols which are vulnerable to man-in-the-middle attack, and suggest a sender’s identifier should be added into authentication message, which can protect the protocol from man-in-the-middle attack.

In addition, security protocols are used on heterogeneous communication channel characterized by different properties, so detecting different protocols require different intruder models. In the future, we will make the further research on different intruder models in different protocol circumstances. And we will also add more properties to the model-checker based on SAT and utilize it to analyze more security protocols.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers of this paper for their carefully reading of the manuscript as well as their many constructive comments. This paper is supported by the National Science Foundation of China under Grant No.60773002863, the 863 Program under Grant No.2007AA01Z472, and the 111 Project under Grant No.B08038.

REFERENCES

[18] Robinson, Nathan; Gretton, Charles; Pham, Duc-Nghiia; Sattar, Abdul,A compact and efficient SAT encoding for planning, 18th International Conference on Automated Planning and Scheduling, ICAPS 2008, pp. 296-303, Sydney, Australia.

Weibo Liu is born in Dezhou, Shandong, P.R. China, 1986. He received his bachelor in Mathematics and Applied Mathematics from Xidian University in 2008. Now he is a graduate, and major in Communication and Information System at Ministry of Education Key Lab of Computer Network & Information Security (CNIS), Xidian University. His research interests include automatic detecting of insecurity protocol and information security.

Wenping Ma received his Ph.D. degree in Communication and Information System form Xidian University in 1999. Now he is a professor in School of Telecommunications Engineering, Xidian University. His research interests include security protocol, information security and communication theory. He has published over 80 papers in some international conferences and journals.

Yuanyuan Yang received his M.S. degree in Communication and Information System form Xidian University in 2007. Now he is a doctor student in at Ministry of Education Key Lab of CNIS, Xidian University. His research interests include security protocol, information security.
Proactive Location Service in Mobility Management

Yu Liu, Qiang Shen, ZhiJun Zhao, Hui Tang
High Performance Network Laboratory, Institute of Acoustics, Chinese Academy of Sciences,
Email: liuy@hpnl.ac.cn

Abstract—This paper proposes a new proactive mobility management mechanism. Instead of traditional mobility management system that when a user wants to connect another it needs to lookup the current location of the called user from the remote home location register (HLR) of the called user, the proposed method provides a special proactive location service to the close friends, who satisfy certain characteristics, of a user. The proactive location service pushes the location information of a user to the current locations of its close friends and is cached there. By doing so, the location lookup of its close friends in calling process is localized resulting in lower lookup delay and signaling cost. The distinct relations between different user pairs are explored, which can be used to identify the close friend group from other friends of a user. The close relation is defined as a function of parameters of calling and mobility characteristics. These parameters are estimated adaptively which are used to update the close relation friends. System performance evaluation and analysis demonstrate that the proposed proactive mobility management mechanism can reduce location lookup delay and signaling cost significantly compared with the traditional mobility management scheme.

Index Terms—mobility management, proactive location service, push-and-cache, relation mining

I. INTRODUCTION

Mobility management is an important aspect of mobile systems, which influences the performance of mobile systems significantly. Adapting to large and diverse mobility characteristics to improve the performance of mobile systems poses challenges to mobility management as mobile users increasing and the widely popularity of mobile services. Location is one of the two basic issues, and another aspect is handoff, in mobility management [1]. In this paper, we focus on the location issue aiming at providing better location service for users and improving the system performance using a proactive location service mechanism.

Two major areas of mobility management include cellular communication and IP-based Internet. In cellular communication system, researchers put majority of their efforts on location update, terminal paging design and the tradeoff between them. A. Bar-Noy et al. gave three location update schemes, distance-based, movement-based and time-based in [2]. In [3], the author analyzed movement-based mobility management and the tradeoff involved between location update and paging. Vincent Wong et al. in [4] gave a survey review on location management, listing eight location update schemes. In IP-based Internet, I.F. Akyildiz et al. provided a survey paper [5], including different layer mobility management schemes, micro, macro schemes and cross-layer schemes. Paper [6] presented a distributed dynamic regional location management scheme in which the gateway foreign agent was dynamically selected to maintain signaling burden balance among foreign agents (FAs) and most of the signaling traffic was restricted locally. A dynamic hierarchical mobility management scheme was developed in [7] and the signaling burden was evenly distributed in the network therein. Meanwhile, the authors gave an algorithm to determine optimal FA chain length. In [8], the authors analyzed and compared mobile IP (MIP) series. However, in all above schemes whenever a user wants to call another user, it needs to lookup the current location of the called user at the called user’s home location register (HLR) and the HLR responses this lookup request reactively. The relations between communication users are useful information which is neglected before in mobility management. This kind of information is explored in the proposed proactive mobility management. To the best of our knowledge this is the first time the proactive location service based on the relations between user pairs is proposed to improve the mobile system performance.

Relations between users are valuable information, however, not being made use of efficiently before. The terms of user and user terminal are used interchangeable in this paper. We have the experience that in our telephone contact book frequent contact people is less, while the other large number of people contacted us rarely. This phenomenon conforms to 80/20 Principle [9] that roughly majority of effects is created by minority of causes. Intuitively, providing better service for these minor but important people will improve the overall performance significantly, which motivates our work.

In the proposed proactive mobility management scheme, the friend users of a user, say Bob, are classified into close relation group and far relation group. The location information of Bob is proactively pushed to the location servers nearby the users within Bob’s close relation friend group and cached there whenever Bob’s location changes. Bob and any of its close friends seem as if they are always connected by a hose. Thus, whenever Bob’s close relation friend wants to lookup the current location of Bob, the location lookup query can be processed locally at the nearby location server rather than a remote location server globally, which reduces the location lookup delay and signaling transport cost. The architecture used to support this mechanism is proposed and the criterion used to determine the close relation friend group is developed in this paper. The characteristics of user mobility and calling frequency are
estimated with least square technology, the information of which is employed in the relation classifying computation. The computation classifying the users and the proactive location service are all implemented on the network side, which can facilitate to save the power of user terminals. We show that through this new proactive location service mechanism the performance of the mobile system is improved significantly in terms of location lookup delay and signaling cost of the system.

The rest of this paper is organized as follows. The proactive mobility management system model is proposed in section II. The following section presents the proactive location service including work flow, user classifying criterion and parameters estimation. System performance is investigated in section IV. Conclusion is drawn in the last section.

II. SYSTEM MODEL

To show the difference between the traditional mobility management scheme and the proposed proactive scheme, we first briefly introduce the Signaling System 7 (SS7) architecture and its mobility management scheme referring to [1].

A. SS7 Mobility Management Architecture

SS7 system is comprised of two mobility management entities, home location register (HLR) and the visitor location register (VLR), as shown in Fig.1. When a user, say Bob, moves from VLR 2 to VLR 3, it updates its location to its HLR and the HLR informs the old VLR, VLR 2, of this change which can make VLR 2 reclaim the resources allocated to Bob before. When another user, say Alice, wants to connect Bob, via its mobile switching center (MSC) the connecting message is relayed to the HLR of Bob, after the HLR exchanging information with current VLR of Bob, VLR 3, HLR replies the MSC connected by Alice with the location information of Bob, upon which the MSC can connect Bob directly. We indicate that these processes are participated by MSCs, and for conveniently exposition we omit the functions of MSCs, routing and allocating temporary identifier called temporary local directory number (TLDN) to users [1], in order to focus on the location service of HLR and VLR architecture. These MSC functions are replaced by directly connecting via users themselves if they obtain the location information of the called user. Mobile terminal (MT) is replaced by user and its friend to reflect user relation. We can see that each time when a user wants to connect another it needs to check the remote HLR and wait for the reply to get the location information of corresponding communication user. We propose the proactive mechanism to provide users with close relations better location lookup service.

B. Proactive Mobility Management Architecture

For easily understanding, we use an example to illustrate the proactive mobility management scheme. A user, say Bob, usually has many friend users. However, the relations between Bob and them are different, some are close, some not. Normally, users with close relation contact each other more frequently than others. The friends of Bob are divided into close relation group and far relation group according to certain criterion which will described in next section. For the time being, we first present the scheme, which is a push-and-cache process. Suppose that the close friend group of Bob includes some
users, denoted by $G$, which is a user set. As shown in Fig.2, each user has a corresponding HLR and whenever it moves to a new VLR it updates its location information to its corresponding HLR. The close relation group of Bob is $G = \{Alice, Jane\}$. The HLRs correspond to Bob, Alice, Jane are HLR 1, HLR 2, HLR 3, respectively. In the proactive mobility management system, HLRs with close relation users exchange user location information between them to facilitate the mobility management process.

Initially, Bob is at VLR 1, Alice at VLR 3, Jane at VLR 5. Since the users in $G$ are Bob’s close friends, i.e. they contact Bob more frequently than other users, HLR 1 pushes the location of Bob to HLRs corresponding to its close friends in $G$, i.e., HLR 2, HLR 3, and the location information of Bob is cached by them. HLR 2 and HLR 3 will push the location information of Bob to the current location of Alice and Jane, VLR 3 and VLR 5 respectively and cached by VLR 3 and VLR 5. It needs to be pointed out that the close relation from the calling direction point of view between two users may not be symmetric. Alice calling Bob frequently does not necessarily mean Bob calling Alice frequently. If it is, the two users have mutually close relation. The users have close relation with a given user plays an important role in the system design and this influences the system performance significantly. Now when Alice wants to connect Bob she just needs to lookup the location information of Bob from VLR 3 locally rather than previously from HLR 1 globally. The location lookup delay is reduced significantly. The signaling traffic is also reduced under certain circumstances which will be analyzed in the sequel. In proactive management system, when a user moves, say Bob in Fig.2, his current location will not only be updated to his own HLR, but also the HLRs of the users in his close relation group, HLR 2 and HLR 3, and further the VLRs at the current location of the users in his close relation group, VLR 3 and VLR 5, all will be updated. The purpose is to keep users within one’s close relation group always being able to acquire Bob’s location in their nearby VLRs locally.

III. PROACTIVE MOBILITY MANAGEMENT

In the previous section, we introduced the system model, based on which the details of the proposed proactive mobility management scheme will be presented in the following.

A. Proactive Location Service

Firstly, we will give the overview of the proactive location service. The terms used in the description are listed in Table I. In the proactive mobility management when user moves it needs to update its own HLR as in the traditional scheme. In addition, its close friends’ HLR and VLR also should be updated. The message sequence of the new scheme is shown in Figure 3. In this figure, just one close friend’s HLR and VLR are drawn as illustration. Nevertheless, other close friends in the close friend group are dealt with similarly. The close friend group of a user is computed in the user’s HLR, which will be proposed in the next subsection. The computed close friend group is stored in the HLR of the user and the HLR of the user informs the close friends’ HLRs of this relation. Notice that the proactive location service is provided exclusively for the close friend group while other users follow the traditional mobility management process.

In the following, when a user moves the proactive location update process involving its close relation
friends is presented as,

(1) When a user detects it has moved into the region of another VLR, it initiates the location update process and send location update message to its HLR via its current VLR.

(2) The current VLR of the user relays the location update message to the HLR of the user.

(3) The HLR of the user replies the location update message to the user upon the received location update message via the VLR from which the update message came.

(4) The VLR relays the reply message to the user.

(5) The HLR informs the previous VLR of the user to reclaim the resources allocated to the user.

(6) The previous VLR acknowledges the informing message.

(7) The HLR of the user checks the close friend group of the user stored on it. If the close friend group is not empty, it proactively pushes the location update information to all the HLRs of the close friends and is cached there.

(8) The close friends’ HLRs reply the location update information.

(9) The location update information of the user is pushed to the current VLRs of the close friends.

(10) The current VLRs of the close friends acknowledge this message.

From above we can see that the user’s HLR and the user’s close friends’ HLRs and VLRs are all updated during the location update process. The advantages of the scheme will be analyzed later in this paper.

When the close friends of a user moves, the location update process is presented below, as shown in Figure 4.

(1) When the close friend of a user moves, it initiates its location update process as in the traditional mode to its HLR via its current VLR.

(2) The current VLR relays this location update information to its HLR.

(3) The HLR of the close friend updates the location information of the close friend and checks if it is any user’s close friend. If it is the close friend of some user, the corresponding user’s location information is sent to the current VLR of the close friend and cached there.

(4) The current VLR relays the location update acknowledgment to the close friend.

(5) The HLR informs the previous VLR of the close friend to reclaim the resources allocated to the close friend.

(6) The previous VLR of the close friend acknowledges this message.

From the above scheme, when a close friend of some one wants to connect the user, it only need to obtain the location of the user from the local VLR rather than remotely from the user’s HLR, which significantly reduces the location lookup signaling cost and location lookup time delay. Of course, these advantages are at the cost of the more operations for the proactively pushing and caching processes. The benefits this scheme achieves depend on the close friend selection which will be analyzed in the following. We indicate that the terms user and close friend of a user are relative concepts. Each user may be a close friend of other users.

### B. The Criterion to Classify The Friends

The computation of determining the close friend is implemented on the HLR of a user. The terms used in the following are listed in Table II. For easy to analyze, it is assumed that \( d_{vh} = d_{hv}, d_{uv} = d_{vu}, c_{uv} = c_{vu}, c_{vh} = c_{hv} \).

The parameters, \( f_i, m_a \), and \( m_1 \), can be estimated in the corresponding users’ HLRs which will be described later.

We will compare the signaling cost of the traditional mobility management scheme and the proactive push-and-cache mobility management scheme to derive the criterion classifying the friends of a user.

In the traditional scheme, the signaling cost of mobility management is composed of two parts, user mobility causing location update signaling cost and location lookup signaling cost when the friends of a user call it. The signaling cost between a user and one of its friends pair, corresponding to a called user and a calling user, respectively, in a unit time interval in the traditional scheme is

\[
C_{i,j} = m_a \cdot \left( 2c_{uv} + 2c_{vh} \right) + \frac{f_i \cdot (c_{uv} + 2c_{vh})}{\text{called user location update}} + \frac{f_i \cdot (c_{uv} + 2c_{vh})}{\text{calling user location lookup}}
\]  

(1)

Where \( c_{uv} \) is the cost from the user to its serving VLR, \( c_{vh} \) in the called user location update term is the cost from the serving VLR of the called user to its HLR while in the calling user location lookup term is from the serving VLR of the calling friend to the HLR of the called user. The signaling cost involved in the previous VLR reclaiming message is omitted for it is the same for both of the schemes.

<table>
<thead>
<tr>
<th>Table II. TERMS 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>( d_{vh} )</td>
</tr>
<tr>
<td>( d_{uv} )</td>
</tr>
<tr>
<td>( d_{vu} )</td>
</tr>
<tr>
<td>( c_{uv} )</td>
</tr>
<tr>
<td>( c_{vu} )</td>
</tr>
<tr>
<td>( c_{vh} )</td>
</tr>
<tr>
<td>( c_{hv} )</td>
</tr>
<tr>
<td>( f_i )</td>
</tr>
<tr>
<td>( m_a )</td>
</tr>
<tr>
<td>( m_i )</td>
</tr>
</tbody>
</table>
In the proactive push-and-cache mobility management scheme, the signaling cost of mobility management is also composed of two parts: user mobility causing location update signaling cost and location lookup signaling cost when the friends of a user call it. The difference between the two schemes is the content of the two components. The signaling cost between a user and one of its friends pair in a unit time interval in the proposed proactive scheme is,

\[ C_{p,i} = m_u \cdot (2c_{uv} + 2c_{wh} + 2c_{hh} + 2c_{hv}) \]

update the HLR of itself

\[ + \quad f_i \cdot c_{uv} \]

update its friend

Where \( c_{hv} \) in the update its friend term is the signaling cost to update its friend from its friend’s HLR to the current VLR of its friend. The differences from the traditional scheme are obvious: in the called user location update term there is another update its friend term while in the calling user location lookup term the cost is reduced to only the local cost from the calling user to the calling user’s current VLR. It is noticeable that the term, \( 2c_{hv} \), in update its friend term is underlined. Combined with the multiplier \( m_u \), this term can be rewritten as,

\[ c_f = m_u \cdot 2c_{hv} \]

Actually, the location update to a friend process from the friend’s HLR to the friend’s VLR can be piggybacked by the friend’s own location update message. The reduced amount relates to the friend’s mobility characteristic, which is represented as,

\[ c_{r,i} = a \cdot m_i \cdot 2c_{hv} \]

Where \( a \) is the related factor, the value of which is limited within \((0, 1)\). Hence, formula (1) can be modified as,

\[ C_{r,i} = C_{p,i} - e_i \]

Where,

\[ e_i = \min(c_{r,i}, c_f) \]

The parameter \( m_i \) can be estimated at friend’s HLR and exchanged with the corresponding user’s HLR. Thus far, the signaling costs of the two schemes between a pair of a user and one of its friends have been formulated. When the cost of the new scheme is lower than the traditional scheme, the friend in this pair is classified as close friend and provided with the proactive location service, or it is a far relation friend and is provided with the traditional location service. The criterion to classify the friend of a user can be represented as,

\[
\begin{align*}
C_{p,i} &< C_{r,i}, \quad i \text{ is a close friend} \\
C_{p,i} &\geq C_{r,i}, \quad i \text{ is not a close friend}
\end{align*}
\]

The parameter \( m_i \) can be estimated at friend’s HLR and exchanged with the corresponding user’s HLR. Thus far, the signaling costs of the two schemes between a pair of a user and one of its friends have been formulated. When the cost of the new scheme is lower than the traditional scheme, the friend in this pair is classified as close friend and provided with the proactive location service, or it is a far relation friend and is provided with the traditional location service. The criterion to classify the friend of a user can be represented as,

\[
\begin{align*}
C_{p,i} &< C_{r,i}, \quad i \text{ is a close friend} \\
C_{p,i} &\geq C_{r,i}, \quad i \text{ is not a close friend}
\end{align*}
\]

Following above criterion, providing the proactive location service to the close friend group definitely reduces the overall system signaling cost.

### C. Parameters Estimation

In the computation of the classification process, three parameters related with the user mobility characteristics and calling characteristic need to be estimated. The mobility characteristic of users is represented as mobility frequency, which is estimated in the HLR of each user. The calling characteristic is described by calling frequency, which is counted at the called user and is reported to the called user’s HLR, such as by being piggybacked in the location update message. The reason putting the estimation task at HLR side is to simplify the user side computation.

We first show how to estimate the mobility frequency of a user from the history data. An observation window is set to count the movement of a user at its HLR, as shown in the upper of Figure 5. The total length of the window is \( kT \) time unit. In each time interval \( T \), the mobility frequency is calculated using (8).

\[ m(j) = \frac{\text{The number of movements in } j\text{th } T}{T} \]

In a window, \( m(1), \cdots, m(k) \) are calculated from the history data. The least square is used to estimate the \((k+1)\)th time interval mobility frequency \( m(k+1) \). The minimum square prediction error is used as the optimization criterion. This linear estimator is figured with \( I \) taps. The relation between window length and the number of taps is constrained by \( k = 2I \), which aims at computing conveniently. The mobility frequency in the \((k+1)\)th interval, \( m(k+1) \), can be estimated,

\[ \hat{m}(k+1) = \sum_{i=1}^{I} w_i m(k+1-i) \]

The prediction error is,

\[ e_n = m(n) - \hat{m}(n) \]

The summation of the square error is,

\[ e = \sum_{n=1}^{I} e_n^2 = \sum_{n=1}^{I} (m(n) - \hat{m}(n))^2 \]
These weights $w_1, \cdots, w_l$ can be calculated using the minimum square error criterion,

$$
\min_{w_1, \cdots, w_l} (e) \tag{12}
$$

This is a standard least square problem [10] and there exist many ways to solve it. With this least square estimator, $m(k+1)$ can be estimated using the history data set $\{m_1, \cdots, m_k\}$. This is a self-adaptive estimator as the observation window sliding forward with time.

As to the calling frequency estimation a similar way can be applied. A user reports its friends’ calling information to its HLR within the location update message. With these history data, the calling frequency of a friend can also be estimated in the least square manner. The data used to estimate the calling frequency of a friend is shown in the lower of Figure 5.

**IV. PERFORMANCE EVALUATION**

In this section, the performances of the proposed proactive mobility management scheme are evaluated from different perspectives. The location lookup delay and the signaling cost regarding to different parameters in a calling and called user pair perspective are investigated. The performances of the lookup delay and signaling cost are also viewed from the overall friend set of a user collectively. These performances are compared with the traditional scheme. Some of the parameters used in the performance evaluation are given here: $c_{uv}=5$, $c_{vu}=5$, $c_{vh}=15$, $c_{hv}=15$, $c_{hh}=10$, $d_{uv}=2$, $d_{vh}=5$, $\alpha=0.8$.

**A. Location Lookup Delay from a Calling and Called User Pair Perspective**

In the traditional scheme, when a user wants to connect another one, it has to lookup the location of the called user from its HLR. The location lookup delay is calculated as (referred to [1]).

$$
D_t = d_{uv} + 4d_{vh} \tag{13}
$$

With the proposed proactive push-and-cache mobility management system the location lookup delay for the close friend is,

$$
D_p = d_{uv} \tag{14}
$$

Obviously, $D_p < D_t$ holds for users in close friend group. We indicate that location lookup of a given user mainly happens on the users in the close friend group, such that the proposed proactive push-and-cache location service will largely improve the overall system location lookup performance.

**B. Signaling Cost from a Calling and Called User Pair Perspective**

For a calling and called user pair, the signaling costs under the traditional and the proactive schemes regarding to calling frequency of the calling friend are compared in Figure 6. There are three cases in the comparison. The traditional scheme line is the signaling cost in a unit time interval under the traditional mobility management scheme. The proactive scheme line without friend classification is the case that treats all the friends with proactive scheme. Comparing these two cases, when the calling frequency is lower the traditional scheme performs better over the proactive one. However, after some point, the classifying point actually, the proactive scheme shows better. With the friend classification, when the characteristics of a friend fall in the close friend the proactive scheme is provided to it, or it is provided with the traditional mobility service, which always makes the lower cost. It needs to be pointed out that, if a friend is close friend or not is dynamically determined by the calling frequency and mobility characteristics. The classifying criterion is actually with three parameters $f', m_u, m_c$.

With the calling frequency as the major classifying factor, we will see how the mobility characteristics impact the scheme. In Figure 7 the influences are drawn. As the mobility of a user increases, the signaling cost increases, which is a common consequence for both the traditional and the proactive schemes. As the mobility of the calling friend increases, the signaling cost between the user and the friend decreases from the called user’s perspective. It is worth to notice that the classifying change point, across which a friend enters into the close
friend group, advances from $f_{11}$ to $f_{12}$ for larger $m_i$ value. This is because updating the friend involved VLR can be piggybacked in the friend’s location update process. The three classifying change points, which correspond to three sets of mobility characteristic parameters, are $f_{11} = 3.6, f_{12} = 4.4, f_{13} = 5.2$, respectively. It can be concluded that, providing a friend with high calling frequency the proactive location service can lower the system signaling cost largely.

### C. Observing Location Lookup Delay from The Overall Friends Perspective

In this subsection and the next subsection we assume that there are totally $N$ friends of the given user and totally $F$ amount of callings in a unit time. Assume that $p_1$ percentage of friends in $N$ are close friends, $p_2$ percentage of calling amount out of $F$ comes from the close friend group. The total location lookup delay in a unit time over all the friends of the user is,

$$D = p_1 \cdot F \cdot D_p + (1 - p_1) \cdot F \cdot D_f$$

(5)

The total location lookup delay over the whole friend set of the user is shown in Figure 8. It shows that with more calling are with proactive location service the overall location lookup delay goes down. The location lookup delay always benefits from the proactive location service. It needs to be pointed out that the proactive location service is provided to close friend rather than individual callings. The calling amount coming from the close friend group of a user often dominates the whole calling amount of the user. Hence the proactive location service reduces the system location lookup significantly.

### D. Observing Signaling Cost from The Overall Friends Perspective

The total signaling cost related to a user under traditional mobility management scheme in a unit time interval is,

$$C_i = m_i \cdot (2c_{uv} + 2c_{vh}) + F \cdot (c_{uv} + 2c_{vh})$$

(6)

The first term is the location update cost and the second term is the location lookup cost when there are callings.

The total signaling cost under the proactive scheme is different from a calling and called user pair case in (5). The case that the proactive location service is simultaneously provided for multiple close friends of a user needs to be considered. Taking account of that different schemes are provided to different friend groups, with the proactive scheme to close friend group and the traditional scheme to other friends, the total signaling cost related to a user in a unit time interval is,

$$C_{total} = m_i \left[ 2c_{uv} + 2c_{vh} + p_1 \cdot N \cdot (2c_{uv} + 2c_{vh}) \right] \cdot \text{close friend group part}$$

$$+ p_1 \cdot F \cdot c_{uv}$$

$$+ \left[ (1 - p_1) \cdot F \cdot (c_{uv} + 2c_{vh}) \right] \cdot \text{non-close friends part}$$

$$- \sum_{i=1}^{N} c_{fi}$$

Set that there are totally $N = 200$ friends of a user, the total calling amount to the user is $F = 800$ in a unit time interval. The total signaling cost related to a user in a unit time interval is drawn with the friend group distribution and calling amount distribution in Figure 9. The region with lower $p_1$ and large $p_2$ has the lowest cost which represents the concentrated calling traffic distribution. Via providing the minority friends who occupy large calling traffic with the proactive location service the whole system performance is improved largely. The regions with simultaneous larger $p_1$ and larger $p_2$, or simultaneous lower $p_1$ and lower $p_2$ indicate that the calling traffic scatters among the friends, under which case the system cost exhibits higher. The region with higher $p_1$ and lower $p_2$ represents the worst dispersion of calling traffic among all the friends which resulting the highest cost. The friend classifying criterion is designed to find the close friend group. The close friend group has the property that small percentage of friends make large percentage of calling traffic, to which the proactive location service is provided can reduce the system signaling cost significantly.

![Figure 8. Total location lookup delay related to a user (F = 50).](image-url)

![Figure 9. Total signaling cost according the friend and calling traffic distribution.](image-url)
IV. CONCLUSION

In this paper, a new proactive mode mobility management scheme is proposed, which proactively pushes and caches a user’s location information to its close friends. The close friend group consisting of close friend has the property that small number of friends occupies majority of the calling traffic. Via this proactive service the location lookup process is localized which leads to lower location lookup delay and lower signaling cost. The criterion used to classify the close friends from other friends is developed. From the performance evaluation, we proved that the proposed proactive mobility management scheme can improve the system performance significantly in terms of location lookup delay and signaling cost. The location lookup delay and signaling cost are view separately in this paper and how to combine these two factors together to develop a more comprehensive close friend classifying criterion and a more comprehensive proactive mobility management scheme considering multiple factors or parameters are the future work.

ACKNOWLEDGMENT

The authors would like to thank the support of 863 Program (China), No. 2008AA01A322, China Postdoctoral Science Foundation funded project under Grant 20100470568.

References


Yu Liu received the BS degree in Electronic Engineering from Xidian University, Xi’an, China, in 2006. He is currently a PhD candidate in Institute of Acoustics, Chinese Academy of Sciences. His research interests include wireless communication, mobile computing, distributed computing, and signal processing.

Zhijun Zhao received the BS degree in photo-electronics and PhD degree in condensed matter physics from University of Science and Technology of China, Hefei, China, in 1996 and 2002, respectively. He is currently an associate researcher in Institute of Acoustics, Chinese Academy of Sciences. His research interests include wireless communication, mobile computing, and distributed computing.

Qiang Shen received the B.S. degree in computer science in 2004, Ph.D. degree in communication engineering in 2009 from Southwest Jiaotong University, Chengdu, China. From April 2007 to April 2008, he was a Visiting Student with the Department of Electrical and Computer Engineering, University of Florida, Gainesville. Then, he joined the High Performance Networks Laboratory, Institute of Acoustics, The Chinese Academy of Sciences, Beijing, China, as a postdoctoral research fellow. He has, to his credit, over 20 high-quality research papers in journals and conference publications, including IEEE Transactions on Vehicular Technology and IEEE Transactions on Wireless Communications et.al. His research interests include protocol design and data fusion in Internet of Things and wireless communication.

Hui Tang received the BS degree in computer science in Lanzhou University, Lanzhou, China, in 1992, the MS degree in computer science in Institute of Computing Technology, Chinese Academy of Sciences, in 1995, and the PhD degree in signal and information processing in Institute of Acoustics, Chinese Academy of Sciences, in 1999. He is currently a researcher in Institute of Acoustics, Chinese Academy of Sciences. His research interests include wireless communication, future Internet and peer to peer network.
Soft-Handoff in WLAN Realized by Dual Link

Hong-yan Qian
College of Info. Sci. & Tech., Nanjing Univ. of Aero. & Astro., Nanjing, China
qhy98@nuaa.edu.cn

Wei-xin Cai
Information Technology Center, Guodian Nanjing Automation Co., Ltd., Nanjing, China
caiwx@sac-china.com

Xiao-lin Qin
College of Info. Sci. & Tech., Nanjing Univ. of Aero. & Astro., Nanjing, China
qinxcs@nuaa.edu.cn

Abstract—WLAN has been widely deployed in many aspects such as city hotspots and video surveillance system. When the mobile station roams in WLAN, it will switch among different access points (AP). This handoff is hard-handoff which is “break-before-make”. Therefore the latency may result in serious problems in some real-time applications because of losing important data. This paper proposes an efficient dual-soft-handoff (DSH) scheme in WLAN based on the analysis of wireless signal channel allocation. In DSH, STA maintains two active data links to forward AP and backward AP respectively, and they don’t switch at one time. It achieves continuous data transmission during fast motion. Simulation analysis and test results indicate that it can provide seamless handoff and efficient data transfer in WLAN.

Index Terms—Soft Handoff; Seamless handoff; WLAN; Mobile Communication; Roaming; Channel

I. INTRODUCTION

The wireless local area network (WLAN) and mobile communication have been penetrated into all aspects of our life with the rapid development of kinds of wireless technologies. Mobile technology can be integrated into WLAN to provide mobility for stations (STA). When a mobile STA roams in WLAN, it will switch among access points (AP) frequently for the limitation of sending and receiving power.

Handoff can be classified as hard-handoff and soft-handoff. The former is “break-before-make” using in IEEE 802.11 and cellular systems, with time division multiple access (TDMA) and frequency division multiple access (FDMA). The latter is “make-before-break” adopted by CDMA [1].

In IEEE 802.11, the standard of WLAN, the STA disconnects with the original AP (OAP) at first, and then connects to new AP (NAP), so it can’t send or receive any data during the handoff interval. There are many studies on diminishing the interval or buffering data, but the always existing handoff latency may be intolerable for some real-time applications such as video surveillance system, voice over IP (VoIP) and kinds of alarm systems.

This paper introduces our solutions for providing seamless data transmission during STA roaming with high speed. The rest of this paper is organized as follow. Section II describes the related works of mobile handoff in WLAN. Section III proposes and analyses the channel allocation scheme of WLAN. Section IV presents the Dual-Soft-Handoff (DSH) scheme. Section V describes the simulation and test in real system, analyzes the results. At last, section VI summarizes the paper, discusses the applicability and presents the future works.

II. RELATED WORK

When a STA roams, it will access the network through different AP, which can be realized at medium access control (MAC) layer (L2) or network layer(L3) [2]. L3 mobility is classified as macro-mobility and micro-mobility [2]. Macro-mobility deals with STA’s moving in networks with distinct access technologies. Mobile IP [3] and its derivatives [4-6] are proposed to it. Micro-mobility deals with STA roaming in networks with the same access technology. Cellular IP and HAWAII are solutions for it [7].

L2 solutions pay attention to the handoff of underlying radio system, such as GPRS and WLAN, and the data rate of WLAN is much higher than GPRS or 3G. The L2 handoff of WLAN is hard-handoff and divided into three phases: scanning, authentication, and re-association [8]. Researches on hard-handoff concentrate on accelerating the scanning phase which can reduce handoff latency more than 80% [9, 10]. Data may be lost during STA’s hard-handoff no matter how efficient it does, whereas soft-handoff realizes seamless handoff without losing data. So soft-handoff is fit for the application with high requirement of reliability. The studies on soft-handoff focus on CDMA system and need the support of both base stations (BS) and STA [2]. In WLAN, soft-handoff can be realized at L2, but mainly using L3’s technology.

L2’s soft-handoff solutions have two kinds, needing the cooperation of STA and AP, or ameliorating only the STA. Inter-Access Point Protocol (IAPP) proposed in IEEE 802.11f mends the process of handoff. After receiving STA’s connection request, the new AP (NAP)
notifies the original AP (OAP) instead of replying to STA directly. When OAP sends all information of the STA to NAP, NAP sends re-association reply to STA to complete the handoff [11]. Though IAPP is “make-before-break” and decreases packet loss, its re-authentication adds handoff latency. Another handicap is current standard AP not supporting IEEE 802.11f.

Soft-handoff in L2 can also be realized by ameliorating STA only. MultiNet updates the data link layer settings and drivers of STA’s wireless network card (NC), adding a middle layer between MAC and network layer. Its multi network interfaces are virtually realized by TDMA. Virtual interfaces connect to real networks at one time to satisfy different access requirements. MultiNet doesn’t refer to STA’s fast moving and rapid handoff [12].

Dual-MAC maintains both connections with the current and new AP simultaneously using two different MAC addresses through only one wireless interface. MAC1 is used in data transfer by channel 1 and channel scan by channel 6, while MAC2 is used in authentication and re-association with NAP [13]. It requires larger coverage of AP. MultiScan develops SyncScan [14] with multi wireless interfaces on STA. The primary interface takes charge of data transfer with OAP while the secondary interface does the passive scan of NAP. It can be further improved by alternating the role of primary and secondary interfaces, that is, they transfer data by turns. It may lose some packets under heavy load or congestion. If the packets are in the queue of the primary interface, then the interface changes its role to the secondary interface, residual packets will be discarded [15, 16]. Pazzi et al. add a pre-handoff stage to collect candidate AP’s advertisement messages before actual handoff. If the NAP is among its candidates, the scanning process is waived and handoff is nearly seamless. If NAP isn’t pre-found, it’ll probe channels as normal [17].

Some protocols are proposed to solve the soft-handoff on higher layer such as at endpoint to avoid modifying the network infrastructure. Stream control transmission protocol (SCTP) [18] and endpoint centric handover (ECHO) [19] realize the end-to-end soft-handoff in heterogeneous wireless IP-based networks.

In this paper, we discuss the wireless channel allocation scheme and put forward Dual-Soft-Handoff scheme to support STA’s fast seamless roaming in WLAN. The handoff trigger time of dual channels is discussed.

### III. THE WIRELESS CHANNEL ALLOCATION SCHEME FOR SOFT HANDOFF

IEEE 802.11 operates in the unlicensed 2.4GHz industrial, scientific, and medical (ISM) band with the bandwidth of 83.5MHz (2.4-2.4835GHz). IEEE 802.11b/g offers high maximum data rate of 11Mbps/54Mbps at the band [20]. 14 channels are defined but only 3 channels haven’t any overlay, such as channel 1, 6, 11. Using channels without overlay can reduce the interference, but it has some limitations for soft handoff. At least two wireless links are needed for soft handoff in WLAN, so the allocation of wireless channel is greatly related with soft-handoff scheme based on dual link.

Firstly, the scheme of adopting channels without overlay is discussed. Fig.1 illustrates three channel allocation modes.

- **(a)** AP with omni antenna/STA with directional antenna
- **(b)** AP with directional antenna/STA with omni antenna
- **(c)** AP with directional antenna/STA with directional antenna

Figure 1. The allocation of wireless channels without overlap

In Fig. 1 (a) and (b), channel 1 is used by AP1, AP3, AP5, ..., AP2n+1; channel 6 is used by AP6, AP4, AP2, ..., AP1. N1 and N2 are two wireless NCs of the STA. The antenna direction of N1 is the same as that of its moving, and the antenna direction of N2 is opposite to that of its moving.

When the STA is at the position shown in Fig.1 (a) or (b), N1 and N2 receive signals of AP1. As it moves continuously along the direction, N1 and N2 will make handoff respectively as follows:

1. When the forpart of STA enters the signal overage of AP2, N1 detects the signal attenuation of AP1 and the signal increase of AP2. N1 will switch to AP2 automatically when its received signal strength (RSS) of AP2 reaches the threshold. The handoff fulfills during its forpart passing the shadow part of Fig. 1(a), and the communication of STA rely on the link between N1 and AP1.

2. When the STA continues moving, N2 detects AP1’s signal attenuation and AP2’s signal increase. It switches to AP2 automatically when its RSS of AP2 reaches the threshold. The handoff fulfills during its back-end passing the shadow part of Fig. 1(a), and the communication rely on the link between N2 and AP2.
The dual link’s handoff control is easy in the mode, but it requires enough space between the STA’s two wireless modules, that is, since $N_1$ enters the shadow and prepares the handoff until it finishes the handoff, $N_2$ can’t enter the shadow area. Otherwise, the interval they entering the shadow will be too small to complete the anterior handoff, so it can’t ensure a data link at all the time.

Supposed the handoff time of wireless channel is 200ms, if the STA’s speed is 40km/h, the distance needed is about 2 meters. The space between $N_1$ and $N_2$ will increase with the speedup of the STA, so the volume of STA can’t be small. Apparently, these two modes aren’t fit for mini STA which has little space of $N_1$ and $N_2$.

Fig. 1 (c) shows a different mode fit for mini STA. In Fig. 1 (c), directional antenna is used by AP and STA, while the adjacent APs use channel 1 and channel 6 respectively.

The antenna of AP has different direction: $AP_{2i+1}$ (i = 1, 2, …, n) ’s antenna direction is the same as STA’s moving orientation, and they use channel 1; $AP_{2i}$ (i = 1, 2, …, n) ’s antenna direction is opposite to STA’s moving orientation, and they use channel 6. The space of adjacent APs is even.

Though the APs signal channels are not overlapped, their signal coverage overlaps. The coverage of $AP_{2i+1}$ is denoted by real lines, and the coverage of $AP_{2i}$ is denoted by broken lines. The coverage isn’t overlapped for APs with the same antenna direction while it is overlapped for APs with diverse antenna directions.

It is required that $N_1$ connects only with $AP_{2i}$ and $N_2$ connects only with $AP_{2i+1}$. When the STA is at the position shown in Fig.1 (c), $N_1$ connects with $AP_2$ and $N_2$ connects with $AP_1$. As it moves continuously along the direction, $N_1$ and $N_2$ will make handoff respectively as follows.

1) When the STA arrives at the position of $AP_3$, $N_2$ goes out of the coverage of $AP_1$ and enters the coverage of $AP_3$; then $N_2$ switches from $AP_1$ to $AP_3$; the handoff will complete during the STA pass the shadow of vertical line. During this period, the signal strength of $AP_3$ boosts up, so $N_1$ keeps data connection with $AP_2$.

2) When the STA arrives at the position of $AP_4$, $N_1$ goes out of the coverage of $AP_4$ and enters the coverage of $AP_6$; then $N_1$ switches from $AP_4$ to $AP_6$; the handoff will complete during the STA pass the shadow of horizontal line. During this period, the signal strength of $AP_3$ is enough for $N_2$ to keep data connection with $AP_2$.

The mode shown in Fig. 1 (c) can meet the soft handoff requirement of mini STA, but the space of AP is only half of its signal coverage. Therefore, it requires double number of AP than that in Fig. 1 (a) or Fig. 1 (b).

In order to reduce the number of AP and lessen the setup work, partly overlapped mode (shown in Fig. 2) is proposed. In Fig. 2, each AP has two wireless NCs supporting IEEE 802.11, denoted as $AP_{i1}$ and $AP_{i2}$ for $AP_i$ (i = 1, 2, …, n). It is also fit for mini STA.

The antenna of AP has different directions: $AP_{12}$ (i = 0, 2, …, n) ’s antenna direction is opposite to STA’s moving orientation, and their coverage is denoted by the real line; $AP_{11}$ (i = 1, …, n) ’s antenna direction is the same as STA’s moving orientation, and their coverage is denoted by the broken line. Channel 1, 4, 8 and 12 are used by $AP_{11}$, $AP_{12}$, $AP_{21}$ and $AP_{22}$ respectively. On the whole, the channels of AP are partly overlapped.

Due to directional antenna, at the position shown in Fig.2, $N_1$ receives signals of $AP_{12}$, $AP_{21}$ and $AP_{31}$ (channel 4, 8, 1) while it needs to recognize only channel 1 and 8(without overlapped). $N_2$ receives signals of $AP_{12}$, $AP_{02}$ and $AP_{21}$ (channel 4, 12, 8) while it needs to recognize only channel 4 and 12(without overlapped).

In STA, the antenna direction of $N_1$ is the same as its moving, and the antenna direction of $N_2$ is opposite to its moving. It stipulates that $N_1$ can only connect with $AP_{11}$ and $N_2$ can only connect with $AP_{12}$.

When the STA is at the position shown in Fig. 2, $N_1$ connects with $AP_{21}$ and $N_2$ connects with $AP_{12}$. As it moves continuously along the direction, $N_1$ and $N_2$ will make handoff respectively as follows.

1) When the STA arrives at the shadow of vertical line, $N_1$ switches from $AP_{21}$ to $AP_{31}$; the handoff will complete before the STA goes out of the shadow. During this period, the RSS of $AP_{31}$ is higher than that of $AP_{31}$, so $N_1$’s handoff can’t be automatically fulfilled by checking RSS. It must be triggered by force, which requires modifying the driver of wireless NC. $N_2$ keeps data connection with $AP_{12}$ at this stage.

2) When the STA arrives at the shadow of horizontal line, $N_2$ switches from $AP_{12}$ to $AP_{22}$; the handoff will complete before the STA goes out of the shadow. $N_1$ keeps data connection with $AP_{11}$ at this stage.

This channel allocation mode requests that STA controls the handoff of $N_1$ and $N_2$ actually. It can satisfy the seamless handoff till the space of adjacent AP is 2/3 of its signal coverage, which is more effective than the mode shown in Fig. 1 (c). Therefore, this mode is adopted by the DSH scheme.

IV. DUAL-SOFT-HANDOFF SCHEME

The working environment of DSH discussed in this paper is shown in Fig. 3. $F_{net}$ is a fix network connected by switches and routers. STA is the mobile station transferring data to nodes in $F_{net}$ through APs. STA has two NCs ($N_1$, $N_2$) with directional antennas mounted back-to-back. Each AP has two NCs with directional antennas.
mounted back-to-back too. AP$_{ij}$ denotes the NC is at location L$_i$ with the direction j:

- j=1: it’s opposite to STA’s moving direction;
- j=2: it’s the same as STA’s moving direction.

### A. Dual-Soft-Handoff

In DSH, N$_1$ receives forward signal from AP$_{i,1}$ (i = 1, 2, …, n); N$_2$ receives backward signal from AP$_{i,2}$ (i = 1, 2, …, n). The adjacent AP adopts distinct channels to reduce interference, for example, AP$_{1,1}$, AP$_{1,2}$, AP$_{2,1}$ and AP$_{2,2}$ using channel 1, 4, 8, 12 respectively. RSS$_{ij}$ denotes STA’s the received signal strength of AP$_{ij}$.

When the STA moves from L$_1$ to L$_2$, it can receive signal from AP$_{2,2}$, AP$_{2,1}$, AP$_{2,2}$ and AP$_{3,1}$. During its moving, the RSS$_{2,1}$ strengthens while RSS$_{2,2}$ lessens continuously. However, after the STA passes L$_2$, the RSS$_{2,2}$ falls dramatically, and the RSS$_{2,1}$ is enough to keep the link in a period of time. Therefore, N$_1$’s handoff from AP$_{2,2}$ to AP$_{3,1}$ should be completed before arriving at L$_2$. N$_2$ takes charge of data transfer with AP$_{1,2}$ at this time.

When the STA arrives at L$_2$, RSS$_{2,2}$ reaches its maximum and N$_2$ can find AP$_{2,1}$. N$_2$ needs to switch to AP$_{2,1}$ before RSS$_{1,2}$ is under the threshold. The STA has connected with AP$_{3,1}$ by N$_1$, so data communication is held by N$_1$ and AP$_{3,1}$.

Fig. 4 describes the general processes of DSH during the STA roaming from L$_1$ to L$_{i+1}$. It includes two phases.

Phase 1 is the forward handoff, and the NAP is in front of the STA. It includes:

1) Data transfer between N$_2$ and AP$_{1,2}$;
2) N$_1$ switches from AP$_{i+1,1}$ to AP$_{i+2,1}$.

Phase 2 is the backward handoff, and the NAP is at the back of the moving STA. It includes:

1) Data transfer between N$_2$ and AP$_{i+2,1}$;
2) N$_2$ switches from AP$_{2,2}$ to AP$_{i+2,2}$.

In Fig.4, one NC’s handoff occurs while the other works normally, so the data link can’t be interrupted.

### B. Handoff triggering time in DSH

Fig. 5 describes one AP group’s coverage. It includes two back-to-back APs. With directional antennas, AP’s coverage is similar to a polygon, which is different from the omni-directional antenna.

Fig.6 describes the change of the RSS of APs during the STA’s moving. In Fig. 6, L$_i$ is the location of AP; RSS$_{ij}$ is the N$_i$ Received Signal Strength of AP$_{ij}$; $T_i$ is the time STA passing L$_i$; $t_1$ is the time N$_1$ can switch; $t_2$ is the earliest time N$_1$ finishing switch; $t_3$ is the time N$_2$ beginning to switch; $t_4$ is the time N$_2$ must finish the switch; $S_{min}$ is the threshold of N$_1$ to be able to probe a AP.

According to Fig.6, there are different policies to handle the handoff while passing L$_2$ from L$_1$:

1) STA finishes the handoff just before the original AP (OAP)’s signal reaches the connection threshold.
2) STA switches immediately when new AP (NAP)’s signal reaches the connection threshold.

If we adopt the former, it has some risk that N$_1$ doesn’t fulfil its handoff before N$_2$’s handoff. So we choose the latter: N$_1$ starts its handoff at $t_1$, just since probing AP$_{3,1}$’s signal; and N$_2$ also starts handoff at $t_2$($t_2 = t_3$) when receiving signal from AP$_{2,2}$. This policy ensures both the handoff and the data communication.

Using the immediate handoff policy discussed above,
it’s clear that the backward handoff to be triggered when passing the access point, whereas the triggering time of forward handoff is worthy researching.

C. Handoff Arimetic in DSH

Fig. 7 illustrates N₁ and N₂’s handoff.

![Figure 7. N₁’s forward handoff and N₂’s backward handoff](image)

In Fig. 7, N₁/N₂ begin to switch at P₁/P₂, and finish switching at P₂/P₃; the distance needed for handoff is d; the distance from the switching point to the OAP is dₒ; the distance between L₁ and L₂ is Dₒ; l is the AP’s effective coverage; θ is the maximum deviation angle of STA’s track.

Arith1 and Arith2 describe the major processes of Dual-Soft-Handoff. They record moment and position of each handoff, which are used to make later handoff trigger more accurately.

**Arith1: the forward handoff**

\[
\text{Dual_FW_Handoff (int D_AB, int i, st_time time_init)} \]

// N1 connects with AP[i,1] at beginning
\{ dis_th[i, 1] = 1cov - dis_AP[i, i+1]; //the distance between STA and AP[i,1] when N1 begins //handoff from AP[i,1] to AP[i+1,1]
if ( (sum_dis_AP[i] – dis_current( )) > dis_th[i, 1] / coe )
// Don’t trigger the handoff because STA is far away // the coverage of AP[i+1, 1]
exit(0);
else

RSS_cur = get_RSS(i+1, 2); //get AP[i+1,1]'s RSS if ( RSS_cur > RSS_min )
// AP[i+1,2]'s RSS is larger than the threshold, so the // communication can be steady
\{ time_h[i, 2] = get_time() - time_init; // The time N2begins to handoff
dis_his_th[i,2] = distance_bk(i ); // The distance between N2 and AP[i,2] when it // begins to handoff
Handoff(N2, i+1, BK);
// N2’s backward handoff
time_c[i, 2] = get_time() - time_init;
// The time N2 finishing the handoff
time_d[i, 1] = time_c[i, 2];
// N1 stops data transfer
data_handoff(N2);
// N2 takes over data transfer
i++;
\}

\]

**V. ARITHMETIC ANALYSIS AND PERFORMANCE TEST**

A. DSH Arithmetic analysis

Fig.8 is supposed that STA moves from station A to B, it goes through three stages: accelerating with the acceleration a₁; moving with even speed; decelerating with the acceleration -a₂.

![Figure 8. STA running at different speed in three stages](image)

If STA’s velocity is v(t) ( \( t_1 \leq t \leq t_2 \) ), we can get v(t):

\[
v(t) = \begin{cases} 
  v_0 + a_1t & (t \in [t_1, t_2], a_1 > 0) \\
  v_{max} + a_2t & (t \in [t_1, t_2], a_2 > 0) 
\end{cases}
\]

If the STA moves with the deviation direction β, we can get the time that STA is in each zone. It is supposed that \( t_i \) is the time used in acceleration stage, \( t_f \) is the time in uniform motion and \( t_j \) is the time used in deceleration stage.
\[ t_1 = \frac{v_{\text{max}}}{a_1} \quad (2) \]
\[ t_1 = \frac{v_{\text{max}}}{a_2} \quad (3) \]
\[ t_2 = D_{\text{AB}} \left( 1 + \frac{v_{\text{max}}}{a_1} \right) - \frac{v_{\text{max}}^2}{2a_1} \quad (4) \]

If \( a_1 = a_2 = a \), then
\[ t_2 = D_{\text{AB}} \left( 1 + \frac{v_{\text{max}}}{a_1} \right) - \frac{v_{\text{max}}^2}{2a_1} \quad (5) \]

According to the above analysis, we need to verify whether the DSH runs correctly. We design a program to simulate the moving mode of STA in Fig. 8.

It is supposed that the STA moves on the constant acceleration with the maximum velocity \( v_{\text{max}} \). To reduce the interference, channel 1, 8 are assigned to AP0,2 and AP1,2; channel 4, 11 are assigned to AP1,1 and AP2,1, and so on. Main parameters set in Tab. 1 and simulation results are shown in Tab. II.

### TABLE I. THE PARAMETERS OF SIMULATION

<table>
<thead>
<tr>
<th>Parameter</th>
<th>meanings</th>
<th>value</th>
<th>unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>( D_{\text{ab}} )</td>
<td>The distance between A and B</td>
<td>2000</td>
<td>m</td>
</tr>
<tr>
<td>( D_{ij} )</td>
<td>The distance between adjacent APs</td>
<td>170~230</td>
<td>m</td>
</tr>
<tr>
<td>( v_{\text{max}} )</td>
<td>STA’s maximum velocity</td>
<td>60/80/120</td>
<td>Km/h</td>
</tr>
<tr>
<td>( T_a )</td>
<td>Accelerating/decelerating time</td>
<td>30/40</td>
<td>s</td>
</tr>
<tr>
<td>( T_{\text{max}} )</td>
<td>Handoff time</td>
<td>300</td>
<td>ms</td>
</tr>
<tr>
<td>( T_{\text{hj},i} )</td>
<td>The time ( N_j ) begins to handoff from AP( i,j )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( T_{\text{c},i} )</td>
<td>The time ( N_j ) connects to AP( i+1,j )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( T_{\text{d},i} )</td>
<td>Data transfer between ( N_j ) and AP( i+1,j ) until ( T_{\text{d},i} )</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### TABLE II. THE MOMENTS OF SOFT-DUAL- HANDOFF:

(A) \( v_{\text{max}}=60\text{Km/h}, T_{\text{a}}=30s \)

<table>
<thead>
<tr>
<th>i</th>
<th>( T_{\text{h},i} )</th>
<th>( T_{\text{c},i} )</th>
<th>( T_{\text{d},i} )</th>
<th>( T_{\text{h},i+1} )</th>
<th>( T_{\text{c},i+1} )</th>
<th>( T_{\text{d},i+1} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>15.76</td>
<td>16.06</td>
<td>24.22</td>
<td>23.92</td>
<td>24.22</td>
<td>28.86</td>
</tr>
<tr>
<td>2</td>
<td>28.56</td>
<td>28.86</td>
<td>32.94</td>
<td>32.64</td>
<td>32.94</td>
<td>38.16</td>
</tr>
<tr>
<td>3</td>
<td>37.86</td>
<td>38.16</td>
<td>42.39</td>
<td>42.09</td>
<td>42.39</td>
<td>46.71</td>
</tr>
<tr>
<td>4</td>
<td>46.41</td>
<td>46.71</td>
<td>51.66</td>
<td>51.36</td>
<td>51.66</td>
<td>54.59</td>
</tr>
<tr>
<td>5</td>
<td>54.29</td>
<td>54.59</td>
<td>60.21</td>
<td>59.91</td>
<td>60.21</td>
<td>64.71</td>
</tr>
<tr>
<td>6</td>
<td>64.41</td>
<td>64.71</td>
<td>68.09</td>
<td>67.79</td>
<td>68.09</td>
<td>73.85</td>
</tr>
<tr>
<td>7</td>
<td>73.55</td>
<td>73.85</td>
<td>78.21</td>
<td>77.91</td>
<td>78.21</td>
<td>82.80</td>
</tr>
<tr>
<td>8</td>
<td>82.50</td>
<td>82.80</td>
<td>87.35</td>
<td>87.05</td>
<td>87.35</td>
<td>91.45</td>
</tr>
<tr>
<td>9</td>
<td>91.15</td>
<td>91.45</td>
<td>95.12</td>
<td>94.82</td>
<td>95.12</td>
<td>120.00</td>
</tr>
<tr>
<td>10</td>
<td>120.00</td>
<td>120.30</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(B) \( v_{\text{max}}=120\text{Km/h}, T_{\text{a}}=30s \)

<table>
<thead>
<tr>
<th>i</th>
<th>( T_{\text{h},i} )</th>
<th>( T_{\text{c},i} )</th>
<th>( T_{\text{d},i} )</th>
<th>( T_{\text{h},i+1} )</th>
<th>( T_{\text{c},i+1} )</th>
<th>( T_{\text{d},i+1} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>12.87</td>
<td>13.17</td>
<td>19.83</td>
<td>19.53</td>
<td>19.83</td>
<td>23.32</td>
</tr>
<tr>
<td>3</td>
<td>30.24</td>
<td>30.54</td>
<td>33.36</td>
<td>33.06</td>
<td>33.36</td>
<td>35.94</td>
</tr>
<tr>
<td>4</td>
<td>35.94</td>
<td>36.24</td>
<td>39.54</td>
<td>39.24</td>
<td>39.54</td>
<td>41.19</td>
</tr>
<tr>
<td>5</td>
<td>41.19</td>
<td>41.49</td>
<td>45.24</td>
<td>44.94</td>
<td>45.24</td>
<td>47.94</td>
</tr>
<tr>
<td>6</td>
<td>47.94</td>
<td>48.24</td>
<td>50.49</td>
<td>50.19</td>
<td>50.49</td>
<td>54.03</td>
</tr>
<tr>
<td>7</td>
<td>54.03</td>
<td>54.33</td>
<td>57.24</td>
<td>56.94</td>
<td>57.24</td>
<td>60.00</td>
</tr>
<tr>
<td>8</td>
<td>60.00</td>
<td>60.30</td>
<td>61.86</td>
<td>61.56</td>
<td>61.86</td>
<td>63.17</td>
</tr>
<tr>
<td>9</td>
<td>63.17</td>
<td>63.47</td>
<td>65.16</td>
<td>64.86</td>
<td>65.16</td>
<td>90.00</td>
</tr>
<tr>
<td>10</td>
<td>90.00</td>
<td>90.30</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

According to Tab. II, we can find that:

\[ T_{\text{c},i+1} < T_{\text{c},i} \quad (6) \]
\[ T_{\text{h},i+1} > T_{\text{h},i} \quad (7) \]

These mean that one NC’s data link maintains until another ready to build a new data link; and the handoffs won’t happen simultaneously. Therefore, the DSH can provide seamless connection during STA’s fast roaming.

The DSH scheme aims at providing high quality data link for the rapid motion nodes. It has high reliability, which is important for applications such as metro control, video or audio transmission. In DSH, if one link is broken-down, STA will switch to single network card mode and wait for resuming from failure automatically.

The DSH requires only the cooperation of STA without any modification of AP, that is, standard IEEE 802.11 serial AP can be adopted to save investment.

### B. Performance testing

To compare the performance, the test contrasts DSH and single data link. Fig. 9 shows the testing environment, and the software IxChariot is adopted in the test. PC in Fig. 9 installs the server of IxChariot and endpoint (the client). STA1 and STA2 install IxChariot endpoint. STA1 has only one link to AP, while handoff triggering time choosing and handoff buffering arithmetic are deployed in STA1 [21]. STA2 deployed DSH has two wireless NCs (supporting IEEE 802.11g) to provide dual links.

![Figure 9. Testing environment of DSH](image)

**Testing includes throughput and response time. Testing throughput using the script Throughput.scr of IxChariot.**

![Figure 10. Throughput testing](image)
Fig. 10 (a) and (b) record the throughput of STA1/STA2 and PC respectively. In Fig. 10 (a), the handoff of STA1 occurs at 46" and 1'18". Network throughput drops dramatically from 22Mbps to about 3Mbps due to the handoff and lasts about 200–300ms.

In Fig. 10 (b), the handoff of STA2 occur (each wireless NC switch twice) four times, but the throughput is equable. The maximum/average/minimum throughputs are 24.9 Mbps/20Mbps/11Mbps. There always exists a data link between STA2 and AP, for dual links make handoff in turn. Soft-handoff is realized by the cooperation of dual links.

Testing the response time of the network uses the script Response_time.scr of IxChariot. It aims at obtaining the minimum response time, so the network load isn’t heavy and the throughput can’t reach its maximum.

Fig. 11 (a) records the response time of STA1 and PC, and there is only a single wireless link between STA1 and AP. Fig. 11 (b) records the response time of STA2 and PC, and there are dual wireless links between STA2 and AP.

In Fig. 11 (a), the handoffs of STA1 occur at about 12" and 33". During the handoff, network response time prolongs greatly from 1ms to about 40–52ms. It is back to 1ms soon after the handoff.

In Fig. 11 (b), the handoff of STA2 occur (each wireless NC switch twice) four times, but the response time ranges from 8ms to 13ms. It is normally less than 10ms during the handoff and data transfer is continuous.

Test results show that DSH achieves higher steady performance than single link in WLAN. In DSH, if one link is failure for some reason, the dual links can adjust to single link mode and send warning message. It also improves the security of the data communication system.

VI. CONCLUSION

We discuss the wireless channel’s allocation for dual link and propose a Dual-Soft-Handoff scheme providing soft handoff in WLAN. The arithmetic analysis and the test results illustrate that it has high performance at STA’s fast roaming. It’s valuable to be used in many kinds of applications with high reliability, real-time, fast moving and frequent handoffs. It makes possible that all sensitive data are sent to control center without delay and urgent instructions are executed in time. It also ensures the real-time monitor of mobile nodes, helpful to public security. In urban infrastructure, it can be used in subway monitor and passenger information system. In military, it may be used in battlefield monitor.

Further work will focus on triggering the forward handoff more accurately according to history data, current velocity and distance between adjacent AP.

REFERENCES


Hongyan Qian, born in 1973, earned BS and MS degree from the department of Computer Science and Technology at Nanjing University of Aeronautics and Astronautics (NUAA), Nanjing, Jiangsu, China, in 1995 and 1998, respectively, both in computer engineering. She earned PhD degree at the College of Information Science and Technology at NUAA in June, 2010. She has worked for NUAA as a lecturer since 1998. Her main research interests are computer network, wireless communication and information security.

Weixin Cai, born in 1969, earned MS degree from the College of Mechanical and Electrical Engineering at Nanjing University of Aeronautics and Astronautics, Nanjing, Jiangsu, China, in 2000. He has worked for Guodian Nanjing Automation Co., Ltd. since 2000. Now he is a Technical Director in IT department. His main research interests are computer network, enterprise informationization and management.

Xiaolin Qin, born in 1953, is a Ph.D in computer application. He is a professor and Ph. D. supervisor in the College of Information Science and Technology at NUAA, Nanjing, Jiangsu, China. His main research interests include security databases, temporal-spatial databases and information security.
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
- 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 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/.
Mobile Platform and Secure Access Approach of UMTS Terminal Based on Trusted Computing
Yingyou Wen, Xuena Peng, Shuyi Chen, and Hong Zhao

A Network Architecture for Load Balancing of Heterogeneous Wireless Networks
Wenxiao Shi, Bin Li, Na Li, and Chuanjun Xia

An Identity Based Aggregate Signature from Pairings
Yike Yu, Xuefeng Zheng, and Hua Sun

Detecting Malware Variants by Byte Frequency
Sheng Yu, Shijie Zhou, Leyuan Liu, Rui Yang, and Jiaqing Luo

A Novel Timing and Frequency Synchronization Technology for OFDM System
Yi Sha, Mingming Li, and Jiafu Chu

Analysis on Stability of a Network Based on RED Scheme
Shengbo Hu, Heng Shu, and Tiancheng Lin

Analysis and Improvement of the BAN Modified Andrew Secure RPC Protocol
Weibo Liu, Wenping Ma, and Yuanyuan Yang

Proactive Location Service in Mobility Management
Yu Liu, Qiang Shen, ZhiJun Zhao, and Hui Tang

Soft-Handoff in WLAN Realized by Dual Link
Hong-yan Qian, Wei-xin Cai, and Xiao-lin Qin