Bond University

DOCTORAL THESIS

Design and Analysis for the 3G IP Multimedia Subsystem

Alam, Muhammad

Award date:
2007

DESIGN AND ANALYSIS FOR THE 3G IP MULTIMEDIA SUBSYSTEM

Presented

by

Muhammad Tanvir Alam

Bachelor of Science, North South University, Bangladesh

Master of Science, Oklahoma State University, USA

Graduate Student Member, IEEE

A dissertation submitted in fulfilment of the requirements of the degree of Doctor of Philosophy for the School of Information Technology, Bond University

Supervised

by

Dr. Zheng da Wu

Associate Professor of Computer Science

August 2007

DESIGN AND ANALYSIS FOR THE 3G IP MULTIMEDIA SUBSYSTEM

Statement of Originality

This thesis represents my own work and the material in this thesis has not been previously submitted for a degree or diploma in any university. To the best of my knowledge this thesis contains no material previously published or written by another person except where due acknowledgement is made in the thesis itself.

Signature:

Date:

Muhammad T. Alam

SID: 12793059

E-mail: muhamta123@gmail.com

School of Information Technology

Bond University

Gold Coast, QLD 4229

Australia

Thesis Abstract

The IP Multimedia Subsystem (IMS) is the technology that will merge the Internet (packet switching) with the cellular world (circuit switching). It will make Internet technologies, such as the web, email, instant messaging, presence, and videoconferencing available nearly everywhere. Presence is one of the basic services that is likely to become omnipresent in IMS. It is the service that allows a user to be informed about the reachability, availability, and willingness of communication of another user. Push to talk over Cellular (PoC) is another service in IMS that is intended to provide rapid communications for business and consumer customers of mobile networks. In order to become a truly successful mass-market service for the consumer segment, the only realistic alternative is a standardized Push-to-talk solution providing full interoperability between terminals and operators. Instant Messaging (IM) is the service that allows an IMS user to send some content to another user in near real-time. This service works under IETF's Message Session Relay protocol (MSRP) to overcome the congestion control problem. We believe the efficiency of these services along with the mobility management in IMS session establishment has not been sufficiently investigated.

In this research work, we identify the key issues to improve the existing protocols in IMS for better system behaviour. The work is centred on the three services of IMS: (1) Presence Service, (2) Push-to-Talk over cellular and, (3) Instant Messaging and over the issue of (4) IMS session set up. The existing session establishment scenario of IP Multimedia Subsystem (IMS) suffers from triangular routing for a certain period of time when an end IMS user or terminal is mobile. In this thesis, the performance of three possible session establishment scenarios in a mobile environment is compared by using an analytical model. The model is developed based on the expressions of cost functions, which represents system delay and overhead involved in sessions'

establishment. The other problem areas in optimizing presence service, dimensioning a PoC service and analysing service rates of IM relay extensions in IMS are identified. A presence server becomes overloaded when massive number of IMS terminals joins a network to request presence facility. Performance models are developed in this research to mitigate such load during heavy traffic for the presence service. Queuing analyses for different cases are provided while instant messaging chunks go through two consecutive relay nodes. The specific factors such as blocking probability, stability conditions, optimized subscription lifetime etc. in IMS environment parameters have been investigated. We have also elaborated models to dimension a PoC service for service providers with regards to controlling PoC session access, optimal PoC session timer, path optimization and number of allowable simultaneous PoC sessions for given network grade of service.

In a nutshell, the contribution of this dissertation are: (a) a proposed robust scheduler to improve performance of the IMS presence service, (b) several derived models to dimension IMS Push-to-talk over cellular service, (c) a new mechanism to reduce cost for the IMS session set ups in mobile environment and (d) evaluation of message blocking and stability in IMS Instant Messaging (IM) service by applying queuing theories. All of these analyses have resulted in recommendations for the performance enhancements with optimal resource utilization in IMS framework.

Acknowledgements

This PhD thesis would not have been possible without the support of many people and organizations to who I owe a great deal. I would like to thank particularly:

Dr. Zheng da Wu, for his belief in my abilities, his advice, his encouragement, his wisdom, his intellectual brilliance, and his endless kindness. Dr. Wu provided me the support I needed at every step. It is his careful supervision that brought me wherever I stand today in my research career.

The School of Information Technology, Bond University, for providing me with a scholarship, resources and the opportunity to pursue with a PhD, as well as introducing me to start my teaching career. Special thanks to Dr. Michael Rees and Stephanie Patching, who continually supported me and made all the difficult paperwork that much easier.

The Australian Government, for providing me with an International Postgraduate Research Scholarship Award during my PhD candidature.

The Australian Academy of Technological Sciences and Engineering (ATSE) for providing me with the Early Career Symposium Fellowship (ECEF) award that inspired me a great deal when I needed.

I would also wish to thank my fellow research students and friends who have ‘come and gone’ during my stay in the research room, room 5323 of IT School, Bond University.

Finally, I would like to thank my parents, who supported and encouraged me throughout this journey of pursuing my doctoral studies. I am ever grateful to their unselfish blessings and support. Every page of this thesis is indeed dedicated to them.

To

My Parents

and

Grandfather,

Late Shafiul A. Chowdhury

Publication Arising from this Thesis

Book Articles

1. M. T. Alam (2007). An Optimal Timer for Push-To-Talk Controller, In D. Taniar (Ed.), *Encyclopaedia of Mobile Computing and Commerce,* Vol: 2, pp: 724-728, Hershey, PA: Information Science Reference.

2. M. T. Alam (2007). Protocol Analysis Over 3G IP Multimedia Subsystem, In D. Taniar (Ed.), *Encyclopaedia of Mobile Computing and Commerc*e, Vol: 2, pp: 778-784, Hershey, PA: Information Science Reference.

Journal Papers

1. M. T. Alam, Z. D. Wu (2007). “Dimensioning and Optimization of Push-to-Talk over Cellular Server“, *International Journal of Network Management,* John Wiley & Sons, Ltd. (In press, www.interscience.wiley.com), DOI: 10.1002/nem.625.

2. M. T. Alam, Z. D. Wu (2007). “Optimal Routing for SIP-based Session Set up over IMS in Mobile Environment", *International Journal of Internet Protocol Technology*, (In Press) Inderscience (www.inderscience.com).

3. M. T. Alam, Z. D. Wu (2007). “End-to-End Delay Measurement for Instant Messaging Relay Nodes,” *Ubiquitous Computing and Communication Journal,* Vol 2 (2), pp: 1-11, ISSN: 1992-8424.

4. M. T. Alam (2006). "On Analysing Cost for Optimizing the Watcher Subscription Time in the IMS Presence Service", *Engineering Letters,* Vol 13(1), pp: 1-10, ISSN: 1816-093X.

5. M. T. Alam, Z. D. Wu (2006). “Admission Control Approaches in the IMS Presence Service“, *International Journal of Computer Science,* WASET, Vol 1(4), pp: 299-314.

   Additional Journal Publication Relevant but not Forming Part of the Thesis

6. M. T. Alam, Z. D. Wu (2007), “Asymptotic Analysis of Instant Messaging Service with Relay Nodes”, *International Journal of Computer, Information, and Systems Science and Engineering,* Vol 1(1), pp: 1-9, ISSN: 1307-2331.

Conference Papers (fully refereed)

1. M. T. Alam, Z. D. Wu (2007). “Proposed Techniques to Dimension a Push-To-Talk over Cellular Server“, *IEEE Consumer Communications & Networking Conference,* CCNC’07, 11-13 Jan, Las Vegas, Nevada.

2. M. T. Alam, Z. D. Wu (2006). “Efficient Scheduling for Reducing Load in the IMS Presence Service” *The IASTED International Conference on Wireless*

Networks and Emerging Technologies, July 3 – 5, Banff, Alberta, pp: 459-464, ACTA press, ISBN: 0-88986-563-9.

3. M. T. Alam, Z. D. Wu (2006). "Cost Analysis of the IMS Presence Service," *First IEEE International Conference on Wireless Broadband and Ultra Wideband Communications* AusWireless'06 Conference, 13-16 March 2006, Sydney, Australia, In CD, available at http://epress.lib.uts.edu.au/dspace/handle/2100/165.

4. M. T. Alam, Z. D. Wu (2005). "Comparison of Session Establishment Schemes Over IMS in Mobile Environment." *Fifth IEEE International Conference on Information, Communications and Signal Processing (ICICS 2005), ISBN:* 0-7803-9283-3, IEEE Catalogue Number: 05EX1118C, December 6-9, Bangkok, Thailand, pp: 638-642, Paper ID: 0581 ***(Registration fee waiver award)*** .

5. Muhammad T. Alam (2005). "An Optimal Method for SIP-Based Session Establishment Over IMS." *2005 International Symposium on Performance Evaluation of Computer And Telecommunication Systems (SCS 2005),* July 24-28, Hilton Cherry Hill/Philadelphia, Philadelphia, Pennsylvania, Sim Series., Vol 37, No. 3, pp: 692-698.

6. M. T. Alam, Z. D. Wu (2005). "Performance Analysis of SIP-Based Session Establishments Over IMS." *The IASTED International Conference on Wireless Networks and Emerging Technologies,* July 19 – 21, 2005, Banff, Alberta, Canada, pp: 178-183, ACTA press, ISBN: 0-88986-499-3, Paper ID: 474-022.

Additional Publication Relevant but not Forming Part of the Thesis

7. M. T. Alam, J. P. Thomas, I. Jonyer (2004). "Reducing Latency in Mobile Ad Hoc Networks by pre-fetching." *The 2004 International Conference on Pervasive Computing and Communications,* (ICWN'04, PCC'04) Las Vegas, Nevada, Vol 2, pp: 952-958, CSREA press

Contents

List of Figures

List of Tables

Abbreviations

ACK	Acknowledgement
AS	Application Server
BS	Base Station
BU	Binding Update
CBQ	Class Based Queuing
CDMA	Code Division Multiple Access
CN	Corresponding Node
Diffserv	Differentiated Services
FER	Frame Error Rate
GGSN	Gateway GPRS Support node
GoS	Grade of Service
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communication
HA	Home Agent
I-CSCF	Interrogating-Call/Session Control Function
IETF	Internet Engineering Task Force
IM	Instant Messaging
IMS	IP Multimedia Subsystem
IP	Internet Protocol
MIME	Multipurpose Internet Mail Extension
MIP	Mobile IP
MMD	Multimedia Domain
MN	Mobile Node
MSRP	Message Session Relay Protocol
MTU	Maximum Transmit Unit
NAT	Network Address Translator
OMA	Open Mobile Alliance
PA	Presence Agent
P-CSCF	Proxy-Call Session Control Function
PDP	Policy Decision Point
PoC	Push-to-Talk over Cellular
PRACK	Provisional Acknowledgement

PS	Presence Server
PTT	Push-to-Talk
PUA	Presence User Agent
QoS	Quality of Service
RAN	Radio Access Network
RLS	Resource List Server
RPID	Rich Presence Information Data Format
S-CSCF	Serving- Call/Session Control Function
SCTP	Stream Control Transmission Protocol
SDP	Session Description Protocol
SGSN	Serving GPRS Support node
SIP	Session Initiation Protocol
SMS	Short Messaging Service
TCP	Transmission Control Protocol
TRU	Transmit/Receive Unit
UA	User Agent
UDP	User Datagram Protocol
UMTS	Universal Mobile Telecommunications System
URI	Uniform Resource Identifier
UTRAN	Umts Terrestrial Radio Access Network
VoIP	Voice over IP
WCBQ	Weighted Class Based Queuing
XDMC	XML Document Management Client
XDMS	XML Document Management Server
XML	Extensible Mark up Language

Chapter 1 Introduction

In the past few years, the evolution of cellular networks has reflected the success and growth the Internet has experienced in the last decade. This leads to networks where IP connectivity is provided to mobile nodes. The result is third generation (3G) networks where IP services such as voice over IP (VoIP) and instant messaging (IM) are provided to mobile nodes (MN) in addition to connectivity. IP Multimedia Subsystem (IMS) is a new framework, basically specified for mobile networks, for providing Internet Protocol (IP) telecommunication services. It has been introduced by the Third Generation Partnership Project (3GPP) in few phases (release 5, 6, 7 and release 8 etc., [24-27], [205-206]) for Universal Mobile Telecommunications System (UMTS) networks. An IP multimedia framework was later introduced by 3GPP2 as the Multimedia Domain (MMD) for third generation Code Division Multiple Access 2000 (CDMA2000) networks, and finally harmonized with IMS. Real-time services can only be properly supported using the release 6 (or higher) IMS specifications. The IMS concept was introduced to address the following network and user requirements:

- Deliver person-to-person real-time IP-based multimedia communications (e.g. voice or video-telephony) as well as person-to-machine communications (e.g. gaming service).
- Fully integrate real-time with non-real-time multimedia communications (e.g. live streaming and chat).
- Enable different services and applications to interact (e.g. combined use of presence and instant messaging).
- Easy user setup of multiple services in a single session or multiple simultaneous synchronized sessions.

Therefore, the IMS is the technology that will merge the Internet (packet switching) with the cellular world (circuit switching). It will make Internet technologies, such as the web, email, instant messaging, presence, and videoconferencing available nearly everywhere. One of the reasons for creating the IMS was to provide the Quality of Service (QoS) required for enjoying, rather than suffering, real time multimedia sessions. The IMS takes care of synchronizing session establishment with QoS provision so that users have a predictable experience. Another reason for creating the IMS was to be able to charge multimedia sessions appropriately. Furthermore, the aim of IMS is not only to provide new services but also to provide all the services, current and future, that the Internet provides. In addition, users have to be able to execute all their services when roaming as well as from their home networks. To achieve these goals, the IMS uses Internet technologies and Internet protocols. So a multimedia session between two users on the Internet is established using exactly the same protocol. It is to be mentioned that, the IMS does not depend on the circuit-switched domain. This way, inter-working with devices with no access to this domain, such as laptops connected to the Internet using any videoconferencing software, becomes trivial.

Problem statement: The message-processing load of an IMS Presence Server (PS) needs to be mitigated since a PS can easily be over loaded while there are massive number of watchers and subscribers requesting for presence service at the same time. The Push-to-Talk over Cellular (PoC) server meeds to be dimensioned to optimize revenue for service providers. Initiation of RE-INVITE messages from SIP (Session Initiation Protocol) Redirect servers need to be avoided during session set up while the IMS terminals are mobile. Queuing characteristics need to be identified properly for IMS Instant Messaging relay nodes for varying service rates.

Thesis contribution: This thesis covers the topic of quality of service (QoS) in IMS, more specifically how to make the services in IMS efficient and robust. The goal is to optimize network in terms of latency, overhead and resource usage. The primary contributions of this dissertation are as follows:

1. A robust scheduler is proposed to improve performance of the IMS presence service.
2. Several models are developed to dimension IMS Push-to-talk over cellular service.
3. A new mechanism is introduced to reduce cost for the IMS session set ups in mobile environment.
4. Message blocking and stability in IMS Instant Messaging (IM) service are evaluated by applying queuing theories.

Thesis structure: The rest of the dissertation is organized as follows. In Chapter 2, the IMS architecture with its services is introduced. The literature review and problem statement of this thesis are investigated in Chapter 3. The thesis objective and methodology are also furnished in this chapter. Chapter 4 depicts the admission control methods with simulation for IMS presence service. Several models to dimension an IMS Push-to-talk service are derived in Chapter 5. A cost efficient method to reduce IMS session establishment delay is presented in Chapter 6. Analysis of a special scenario in Instant Messaging with two relay nodes is furnished in Chapter 7. Finally, Chapter 8 concludes the thesis with recommendations for future work.

Chapter 2 Background

Both 3GPP and 3GPP2 have standardized their own IP Multimedia Subsystem specifications. IETF (Internet Engineering Task Force) also collaborates with them in developing protocols that fulfil their requirements. We discuss the architecture and a few services of IMS in this chapter which are the area of interest of this thesis.

2.1 Overview of IMS Architecture

Figure 2-1 depicts an overview of the IMS architecture ([24, 206]).

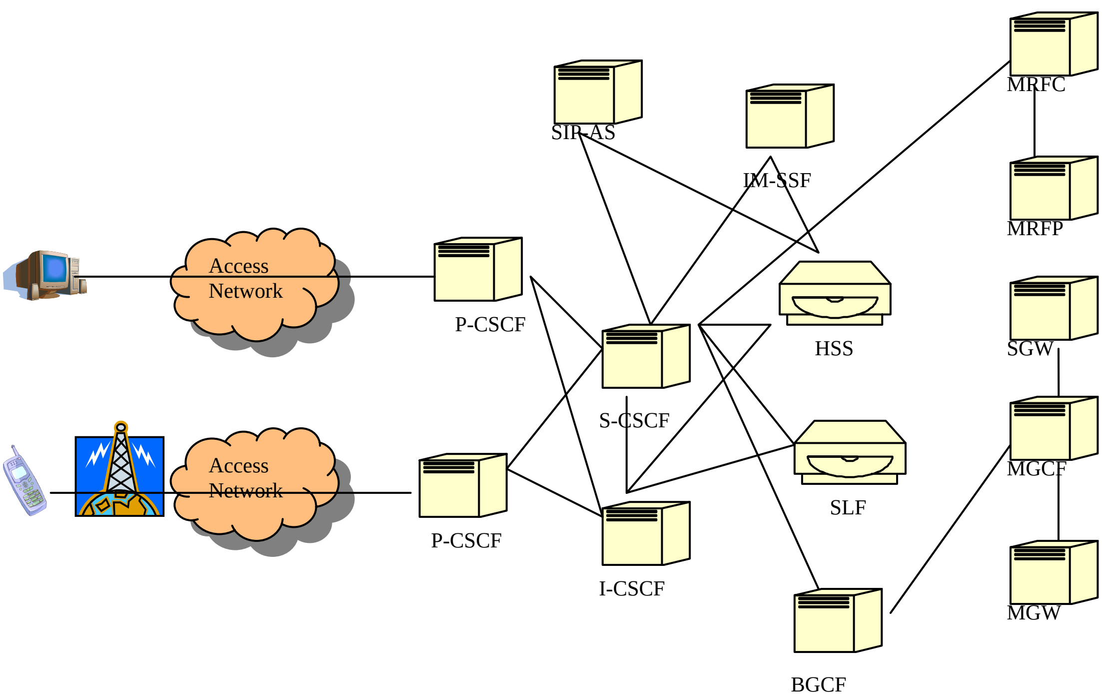


Figure 2-1: GPP IMS architecture overview

The common nodes included in the IMS are as follows:

1. CSCF (Call/Session Control Function): CSCF is a SIP (Session Initiation Protocol) server which processes SIP signalling in the IMS. There are three types of CSCFs (discussed below) depending on the functionality they provide.

2. P-CSCF (Proxy-CSCF): The P-CSCF is the first point of contact between the IMS terminal and the IMS network. All the requests initiated by the IMS terminal or destined to the IMS terminal traverse the P-CSCF. This node provides several functions related to security. The P-CSCF also generates charging information toward a charging collection node. An IMS usually includes a number of P-CSCFs for the sake of scalability and redundancy. Each P-CSCF serves a number of IMS terminals, depending on the capacity of the node.

3. I-CSCF (Interrogating-CSCF): The I-CSCF provides the functionality of a SIP proxy server. It also has an interface to the SLF (Subscriber Location Function) and HSS (Home Subscriber Server). This interface is based on the Diameter protocol (RFC 3588 [41]). The I-CSCF retrieves user location information and routes the SIP request to the appropriate destination, typically an S-CSCF.

4. S-CSCF (Serving-CSCF): The S-CSCF is a SIP server that performs session control. It maintains a binding between the user location and the user's SIP address of record (also known as Public User Identity). Like the I-CSCF, the S-CSCF also implements a Diameter interface to the HSS.

5. SIP AS (Application Server): The AS is a SIP entity that hosts and executes IP Multimedia Services based on SIP.

6. IM-SSF (IP Multimedia Services Switching Function): The IM-SSF acts as an Application Server on one side and on the other side, it acts as an SCF (Service Switching Function) interfacing the gsmSCF (GSM Service Control Function) with a protocol based on CAP (CAMEL Application Part, defined in 3GPP TS 29.278 [47]).

7. MRF (Media Resource Function): The MRF provides a source of media in the home network. It is further divided into a signalling plane node called the MRFC (Media Resource Function Controller) and a media plane node called the MRFP (Media Resource Function Processor). The MRFC acts as a SIP User Agent and contains a SIP interface towards the S-CSCF. The MRFC controls the resources in the MRFP via an H.248 interface. The MRFP implements all the media-related functions.

8. BGCF (Breakout Gateway Control Functions): BGCF a SIP server that includes routing functionality based on telephone numbers.

9. SGW (Signalling Gateway): SGW performs lower layer protocol conversion.

10. MGCF (Media Gateway Control Function): MGCF implements a state machine that does protocol conversion and maps SIP to either ISUP (ISDN User part) over IP or BICC (Bearer Independent Call Control) over IP. The protocol used between the MGCF and the MGW is H.248 (ITU-T Recommendation H.248 [48]).

11. MGW (Media Gateway): The MGW interfaces the media plane of the PSTN (Public Switched Telephone Network) or CS (Circuit Switched) network. On one side the MGW is able to send and receive IMS media over the Real-Time Protocol (RTP). On the other side the MGW uses one or more PCM (Pulse Code Modulation) time slots to connect to the CS network. Additionally, the MGW performs trans-coding when the IMS terminal does not support the codec used by the CS side.

12. The Home Subscriber Server (HSS) contains all the user related subscription data required to handle multimedia sessions. These data include, among other items, location information, security information (including both authentication and authorization information), user profile information and the S-CSCF allocated to the user. The *SLF* (Subscription Location Function) is a simple database that maps users' addresses to HSSs. Both the HSS and the SLF implement the Diameter protocol.

There are plenty of ways to improve the existing infrastructure and protocols in IMS. The signalling overhead reaches its peak when massive number of IMS terminals joins the network at the same time. The services in IMS lack network optimization and efficient admission control mechanisms. In this thesis, we identify a few key issues to improve the existing protocols in IMS for better system behaviour.

2.2 SIP in IMS

Session Initiated Protocol (SIP) is a prominent protocol (RFC 3261, [11]) today in the third generation network. It facilitates mainly multimedia data transfer. SIP has been chosen in IMS to play the key role for setting up the session while inter-working with other protocols. Originating in 1996 as part of the development of multicasting, SIP came to prominence once the direction of VoIP (Voice over IP) technology development moved from "low cost" to "value add". It is a lightweight, text-based protocol that is easily programmed, highly flexible and readily scalable. As the name implies, SIP is about initiating interactive communications sessions between users. It also handles termination and, most interestingly, modifications of sessions in progress as well. Once the user has been located, the correct session for the type of terminal he is using at the time needs to be established. SIP achieves all of this.

Wherever there is a requirement for real-time sessions to be established, SIP can reside in the communications device and handle these sessions. There are many more potential areas of use: IP Centrex, instant messaging, presence management, desktop call management and unified messaging, web commerce, on-line gaming application; to name a few.

3GPP, which is setting the standards for Universal Mobile Telecommunications System (UMTS), has now standardised on SIP for call control and signalling on third generation mobile networks. All IP voice and multimedia call signalling in IMS will be performed by SIP, end to end, providing a basis for rapid new service introductions and integration with fixed network IP services (such as streamed content) once the basic platform is in place.

2.3 Session Establishment Scenario for a Mobile Terminal

Every mobile node must register with the visited network in IMS. The SIP INVITE request is sent from the UE (user equipment) to S-CSCF#1 (serving call session control function) via P-CSCF#1 (proxy call session control function) by the procedures of the originating flow to initiate a session between two nodes. This message may contain the initial media description in the SDP (session description protocol). S-CSCF#1 performs an analysis/filter criteria and passes the request to I-CSCF#1 (Interrogating CSCF) and so on. Thus the intermediate nodes analyse and forward the request to the next node till it reaches the destination node. The detail of IMS SIP session set up procedures with MIPv6 can be found in [24]. Figure 2-2 and Figure 2-3 depict the serving to serving procedure for same and different operators respectively.

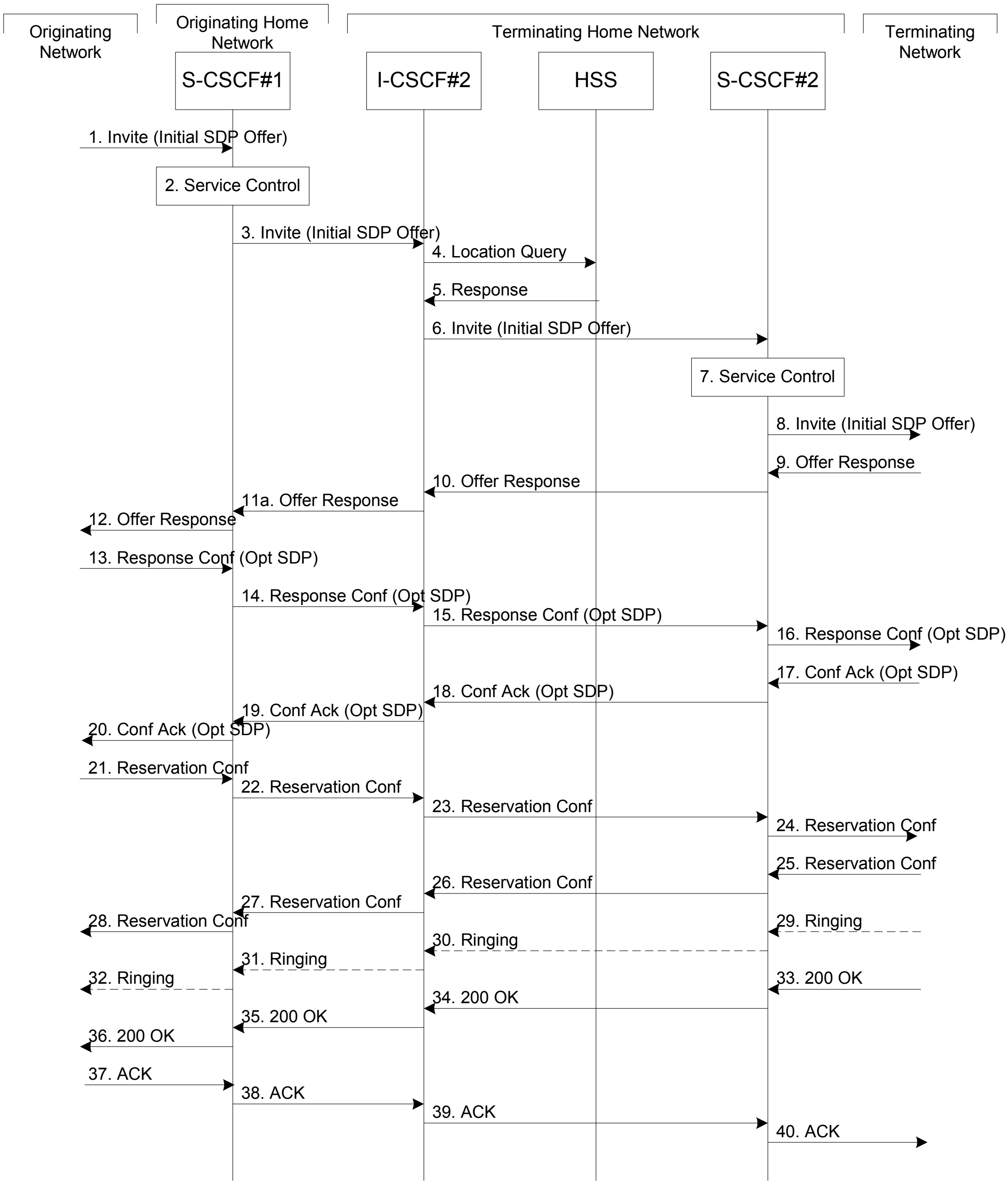


Figure 2-2: Serving to serving procedure - same operator [24]

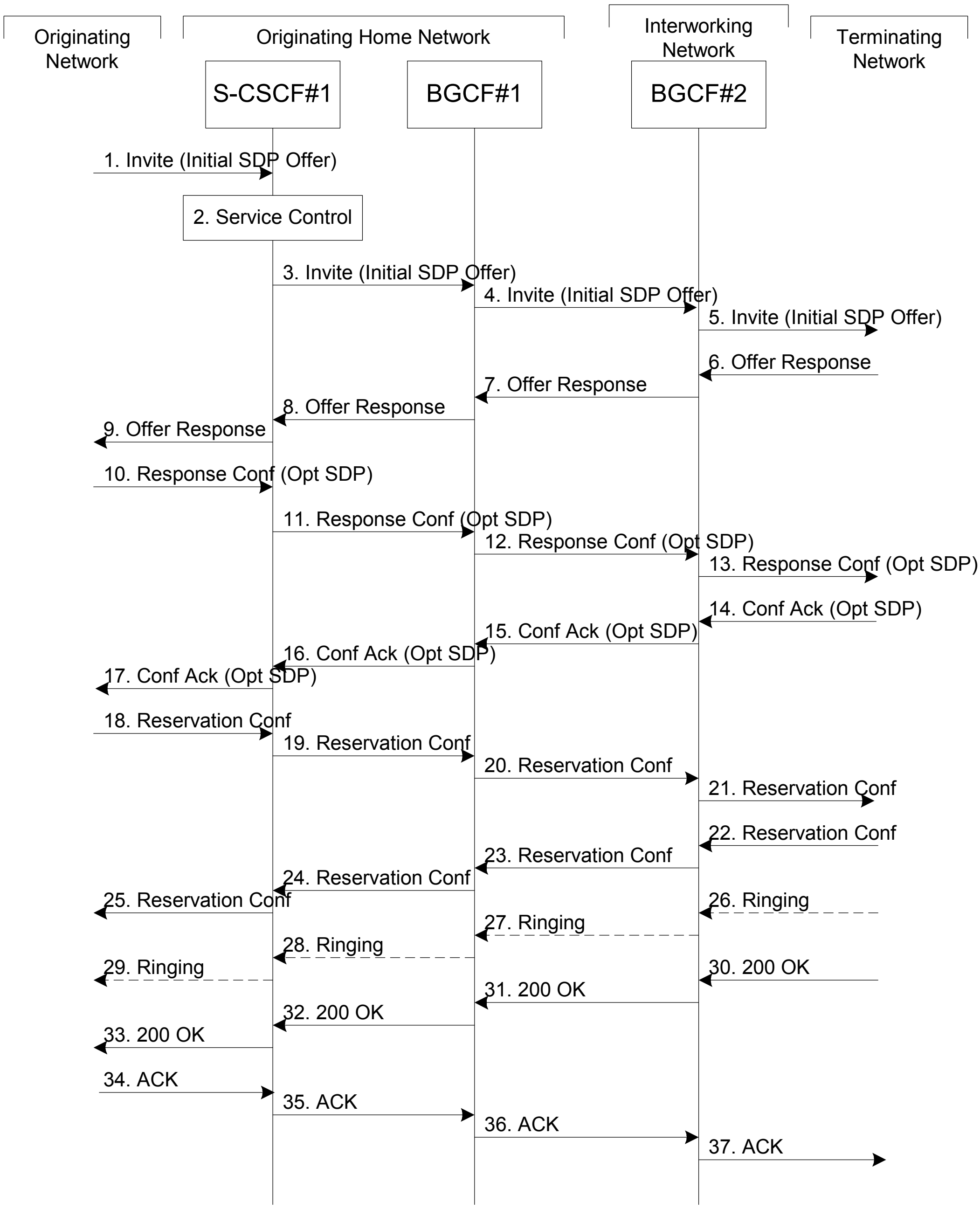


Figure 2-3: Serving to PSTN procedure - different operator [24]

If a mobile terminal moves away from its current visited network, it needs to send Binding Update (BU) message to the corresponding node. It may move away during the session set up. The issue of sending BU to achieve better mobility management needs to be addressed thoroughly. The fact that MIP (Mobile IP) mobility is preferred over SIP, introduced the adoption of SIP services with MIP in the latest releases of IMS. 3GPP has selected IPv6 as the IP version supported by the IMS in order to benefit from the advantages of IPv6. Both SIP and MIP support mobility of the MN (mobile node). However, the two types of mobility are rather different. When MIP is used, the MN has two addresses: the HoA (home address) and CoA (care-of-address). MIP supports node mobility by allowing applications to be unaware of a change in node address. Therefore, the addresses used by the MN for SIP communications is the HoA. However, the MN’s current point of attachment corresponds to the CoA, so to avoid tunnelling of SIP signalling through the HA (home agent), the CoA should be used to exchange SIP signalling. An additional aspect to consider is the IP address used by the MN as source address in IP packets containing the SIP messages sent to the Proxy-CSCF, and the security mechanisms required to ensure SIP signalling security. The IMS has defined a security mechanism to verify that the source IP address of SIP messages from the MN corresponds to the IP address in the SIP messages. Hence, this requires the MN to use the same address (i.e., either the HoA or CoA) for the source address and the address provided at the SIP level.

In the existing scenario, the mobile node sends BU to the corresponding node after the session is set up. The MN receives packets from the CN (corresponding node) tunnelled though the HA, and initiates the route optimization procedure. This implies that traffic will be routed through the HA before being routed directly to the MN, even if for a limited amount of time. This can have implications on quality of service (QoS), since QoS is initially established only for the route from the MN to the HA and to the CN, whereas QoS for the optimized route is not established.

2.4 Registration Scenario in IMS

The application level registration is initiated after the registration to the access is performed, and after IP connectivity for the signalling has been gained from the access network. For the purpose of the registration information flows, the user is considered to be always roaming. For user roaming in their home network, the home network shall perform the role of the visited network elements and the home network elements.

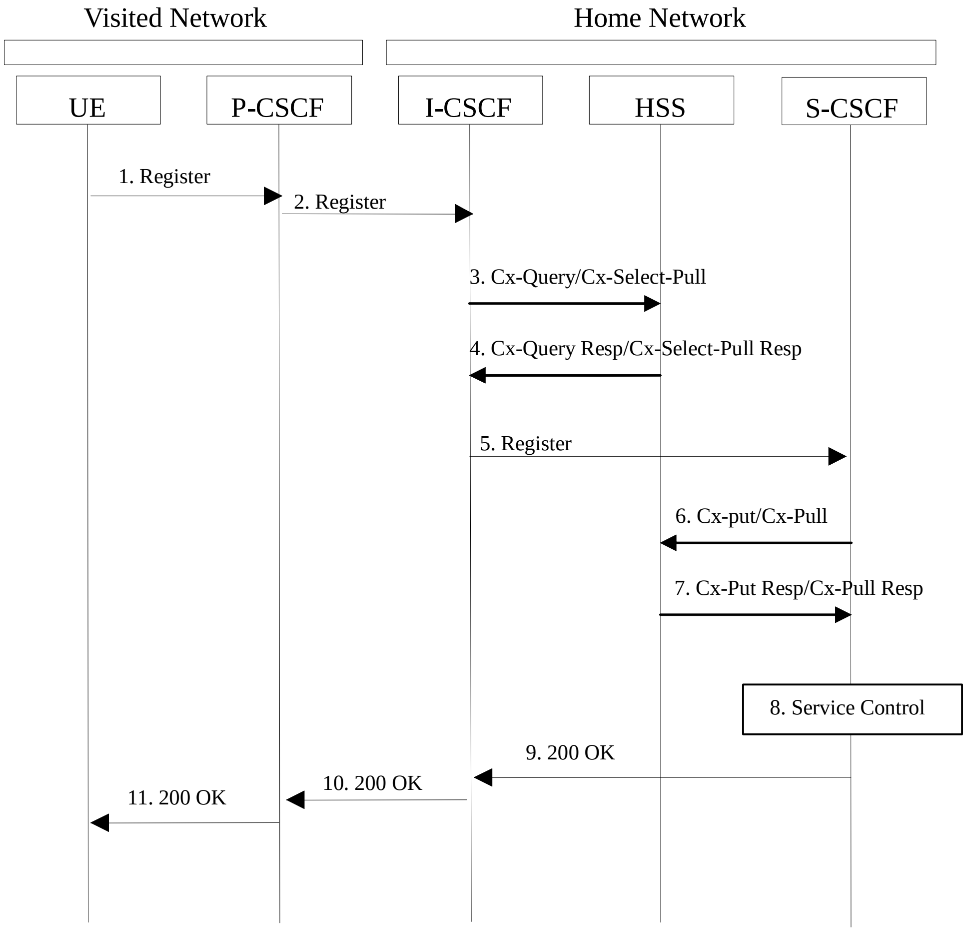


Figure 2-4: Registration – User not registered [24]

Figure 2-4 depicts the registration process of a mobile node that registers with the home network for the first time. Re-registration follows the same process of registration in IMS (Figure 2-5). When initiated by the UE (User Equipment), based on the registration time established during the previous registration, the UE shall keep a timer shorter than the registration related timer in the network. If the UE does not re-register, any active sessions may be deactivated. Prior to expiry of the agreed registration timer, the UE initiates a re-registration. To re-register, the UE sends a new

REGISTER request. The UE sends the REGISTER information (public user identity, private user identity, home network domain name, UE IP address) flow to its proxy.

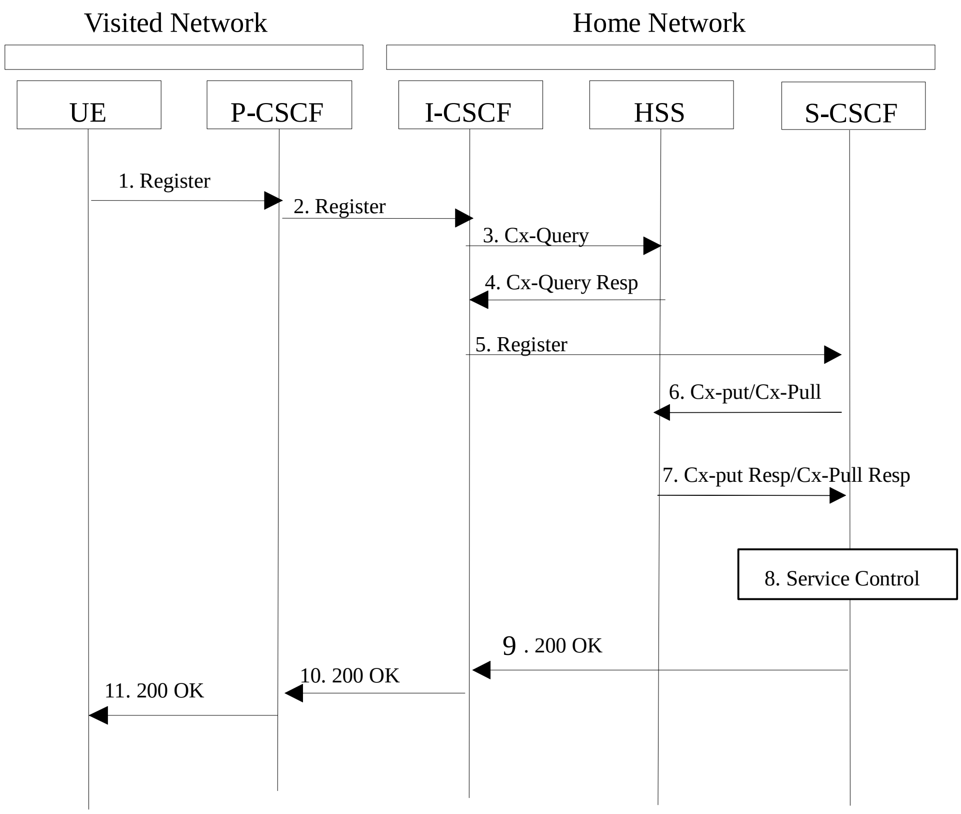


Figure 2-5: Re-registration - user currently registered [24]

When the UE wants to de-register from the IMS then the UE shall perform application level de-registration. De-registration is accomplished by a registration with an expiration time of zero seconds [24].

2.5 Presence Service in the IMS

Presence is one of the basic services that is likely to become omnipresent in IMS. It is the service that allows a user to be informed about the reachability, availability, and willingness of communication of another user. The presence service is able to indicate whether other users are online or not and if they are online, whether they are idle or busy. Additionally the presence service allows users to give details of their communication means and capabilities.

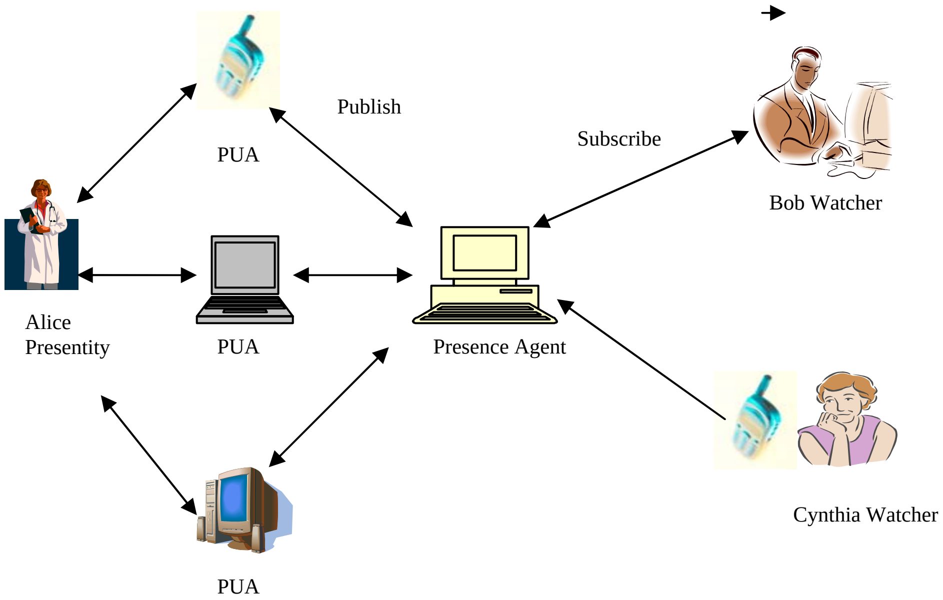


Figure 2-6: SIP Presence system

The presence framework defines various roles as shown in the above figure (Figure 2-6). The person who is providing presence information to the presence service is called a presence entity, or for short a presentity. In the figure, Alice plays the role of a presentity. The presentity is supplying presence information such as status, capabilities, communication address etc. A given presentity has several devices known as Presence User Agents (PUA) which provide information about her presence. All PUAs send their pieces of information to a presence agent (PA). A presence Agent can be an integral part of a Presence Server (PS). A PS is a functional entity that acts as either a PA or as a proxy server for SUBSCRIBE requests. Figure 2-6 also shows two watchers: Bob and Cynthia. A watcher is an entity that requests (from the PA) presence information about a presentity or watcher information about his/her watchers. A subscribed watcher asks to be notified about future changes in the presentity’s presence information, so that the subscribed watcher has an updated view of the presentity’s presence information.

End-users benefit from the presence service since they decide what information related to presence they want to provide to a list of authorized watchers. Presentities can decide the information they want to publish, such as communication address, capabilities of the terminals, availability to establish a communication. Watchers get that information in real time and decide how and when to interact with the presentity. The possibilities enrich both the communication and the end user experience of always being in touch with their relatives, friends and co-workers. On the other hand, presence information is not only available to end-users but also to other services. These other services can benefit from the presence information supplied. For instance, an answering machine server is interested in knowing when the user is online to send them an instant message announcing that they have pending voicemails stored in the server. A video server can benefit by adapting the bandwidth of the streaming video to the characteristics of the network where the presentity’s device is connected. For these reasons, the presence service is referred as the foundation for service provision.

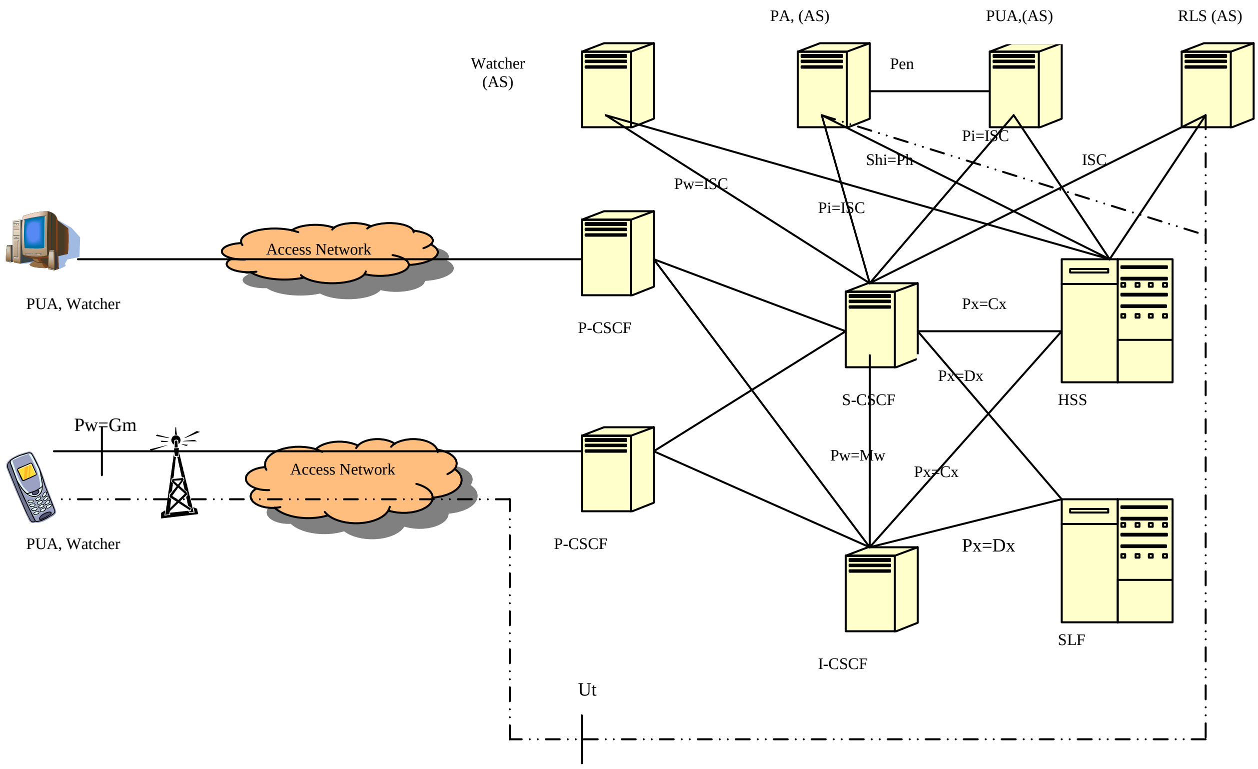


Figure 2-7: SIP-based IMS presence architecture

The presence architecture derived from Figure 2-1 is presented in Figure 2-7. 3GPP defined in 3GPP TS 23.141 [49] provides the architecture to support the presence service in the IMS. Most of the interfaces get a name starting with a "P" (e.g., *Pw, Pi, Px*), but most of them are existing IMS SIP or Diameter interfaces that map to a presence-oriented function. The *Pen* interface allows an Application server that is acting as a PUA to publish presence information to the presentity's PA. The PUA acquires the presence information from any possible source of information, such as the HLR (Home Location Register), the MSC/VLR (Mobile Switching Centre/Visited Location Register) in circuit-switched networks, the SGSN (Serving GPRS Support node), the GGSN (Gateway GPRS Support Node) in GPRS networks, or the S-CSCF through IMS registration. The other interface is the so-called *Ut* interface. This interface is defined between the IMS terminal and any application server, such as a PA or an RLS. The *Ut* interface allows the user to get involved in configuration and data manipulation, such as configuration of presence lists, authorization of watchers, etc. The protocol on this interface is XCAP (XML Configuration Access Protocol) with one or more specific application usages that depend on the particular application. XCAP [108] provides a client with the means to add, modify, and delete XML configuration data of any kind stored in a server, such as users in a presence list, authorization policies (e.g., list of authorized watchers), or a list of participants in a conference. Figure 2-8 shows the schematic representation of the protocol stack used by XCAP.

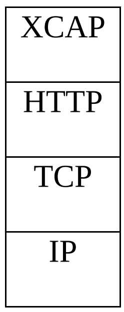


Figure 2-8: The XCAP protocol stack

XCAP defines conventions that map XML documents and their components to HTTP URLs. It also defines the rules that govern how modification of one resource affects another. Additionally, XCAP also defines the authorization policies associated with access to resources. It provides the client with the following operations: create a new document, replace an existing document, delete an existing document, fetch a document, create a new element in an existing document, replace an existing element in a document, delete an existing element in a document, replace an attribute in the document, delete an attribute from the document, and fetch an attribute of a document.

The watcher subscription flow is illustrated in the Figure 2-9. The watcher application residing in the IMS terminal sends a SUBSCRIBE request (1) addressed to her list for example sip:alice-list@home1.net. The request (2) is received at the S-

CSCF, which evaluates the initial filter criteria. One of those criteria indicates that the request (3) ought to be forwarded to an Application Server that happens to be an RLS (Resource List Server). A RLS can be implemented as an Application Server in IMS. The RLS, after verifying the identity of the subscriber and authorizing the subscription, sends a 200 (OK) response (4). The RLS also sends a NOTIFY request (7), although it does not contain any presence information at this stage. The RLS subscribes one by one to all the presentities listed in the resource list and, when enough information has been received, generates another NOTIFY request (13) that includes a presence document with the aggregated presence information received from the presentities' PUAs. Figure 2-10 shows the RLS subscribing to one of the presentities contained in the resource list.

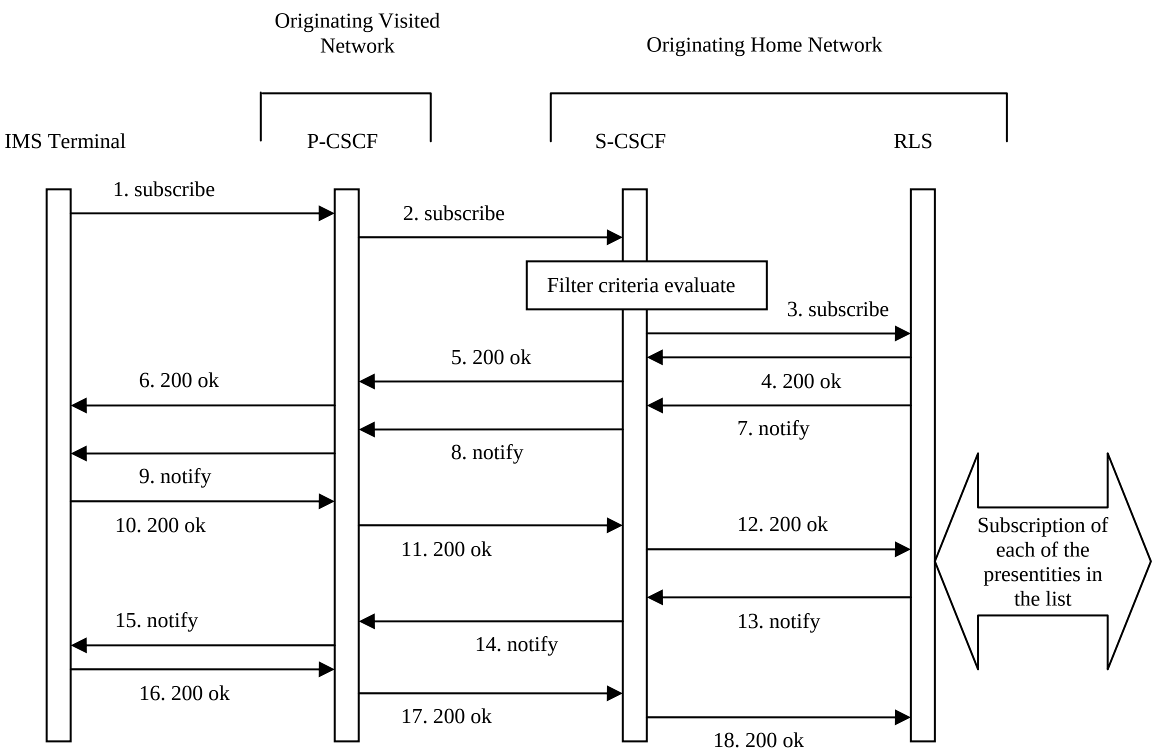


Figure 2-9: Watcher subscription to own list

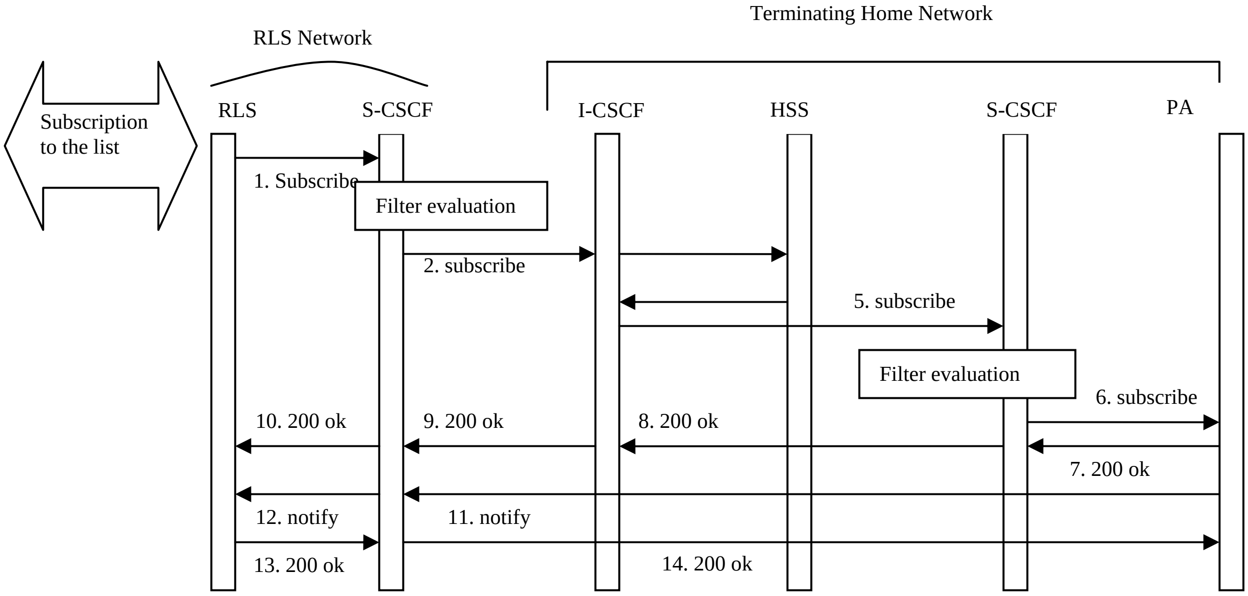


Figure 2-10: The RLS subscription to a presentity

When the IMS presence application starts, it publishes the current presentity's presence information. Figure 2-11 shows the flow. The IMS terminal sends PUBLISH request (1) that includes an Event header set of presence. The S-CSCF receives the request (2) that includes and evaluates the initial filter criteria for the presentity. One of the initial filter criteria indicates that PUBLISH requests containing an Event header set to presence ought to be forwarded to the PA/PS where the presentity's presence information is stored. So, the S-CSCF forwards the PUBLISH request (3) to that Application Server. The PA/PS authorizes the publication and sends a 200 (OK) response (4).

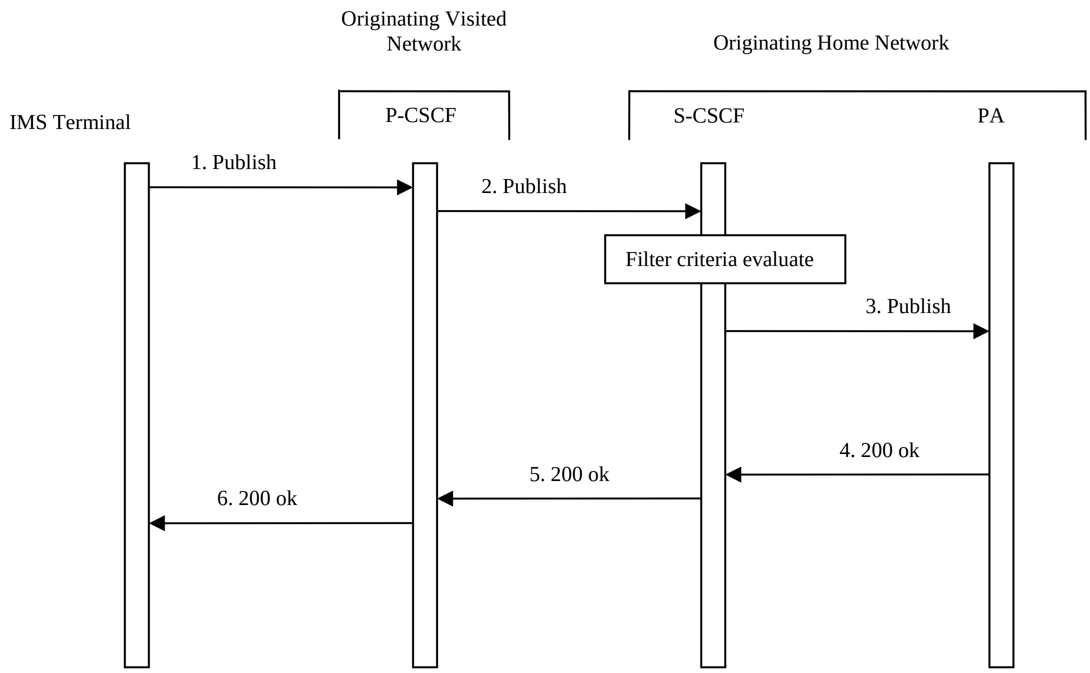


Figure 2-11: The IMS terminal publishing presence information

The above IMS presence architecture indicates that the flow of messages will be massive for large amount of publishers and watchers joining an IMS system.

2.6 Push-to-Talk Service in IMS

Push-To-Talk can be viewed as an Instant Messaging service, enhanced with voice functionality. Ericsson, Motorola, Nokia and Siemens were the first vendors to team up to develop the open Push-To-Talk industry standard called PoC (Push-To-Talk over Cellular) [109]. This jointly defined specification was submitted to OMA (Open Mobile Alliance, [112]) to facilitate multi-vendor interoperability for Push-to-Talk products. The specification is based on 3GPP's (Third Generation Partnership Project) IMS (IP Multimedia Subsystem, [24]) architecture and PoC is to bring the first commercial implementations of the IMS architecture into mobile networks. A discussion on strategic actions related to standardization, system architecture and service diffusion of PoC has been discussed in [111]. An exploratory discussion of

Voice over IP and CDMA usage in 2.5G/3G systems relating to Push-to-talk service has been furnished by DaSilva *et al* (2006) in [110].

Push to talk over Cellular (PoC) is intended to provide rapid communications for business and consumer customers of mobile networks. PoC will allow user voice and data communications shared with a single recipient, (1-to-1) or between groups of recipients as in a group chat session, (1-to-many) such as in Figure 2-12.

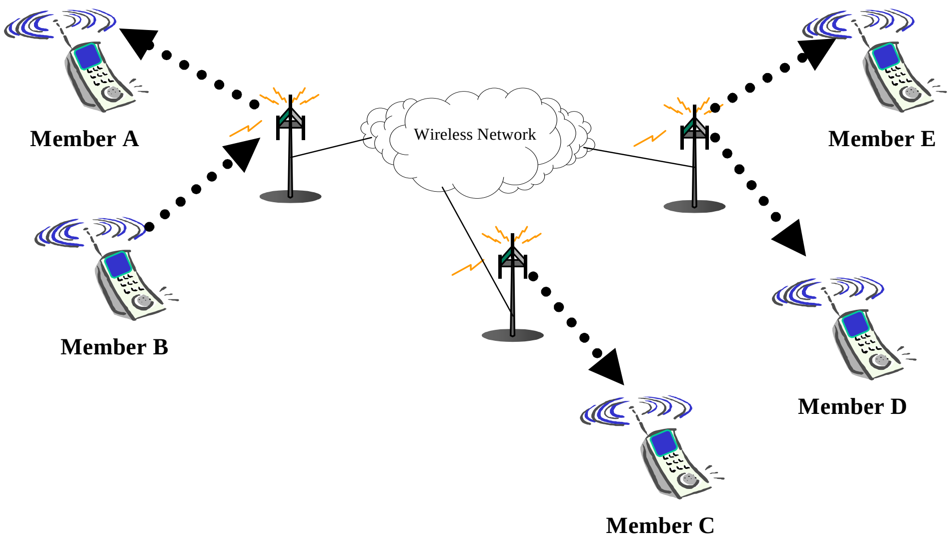


Figure 2-12: Example of a PoC 1-to-many Group Session (voice transmission) [112]

The PoC logical architecture is provided in Figure 2-13 according to the OMA release.

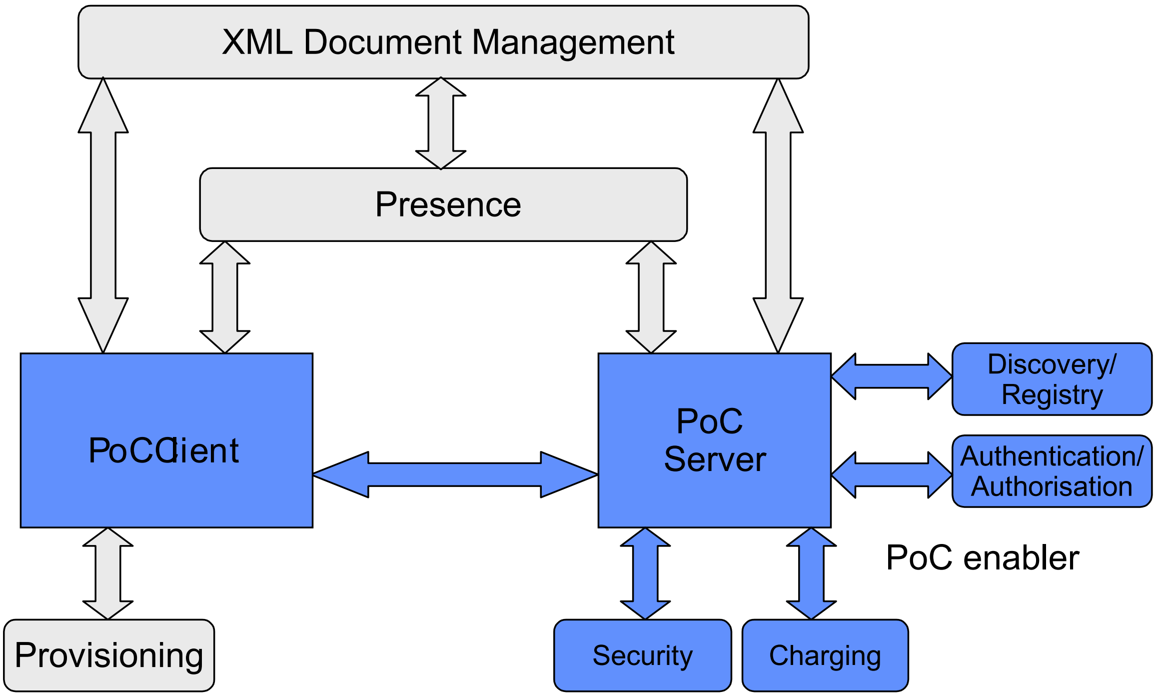


Figure 2-13: Logical architecture of PoC [112]

The XDM functional entities are the Aggregation Proxy and Shared XDMS (*"Shared XML Document Management Server (XDMS)"*). The Presence functional entities are the Presence Server, Presence Source, and Watcher. The PoC Server can assume the role of a Presence Source and/or Watcher, and interacts with the Presence Server. The physical architecture is presented in Figure 2-14. PoC utilizes SIP/IP Core based on capabilities from IMS as specified in 3GPP.

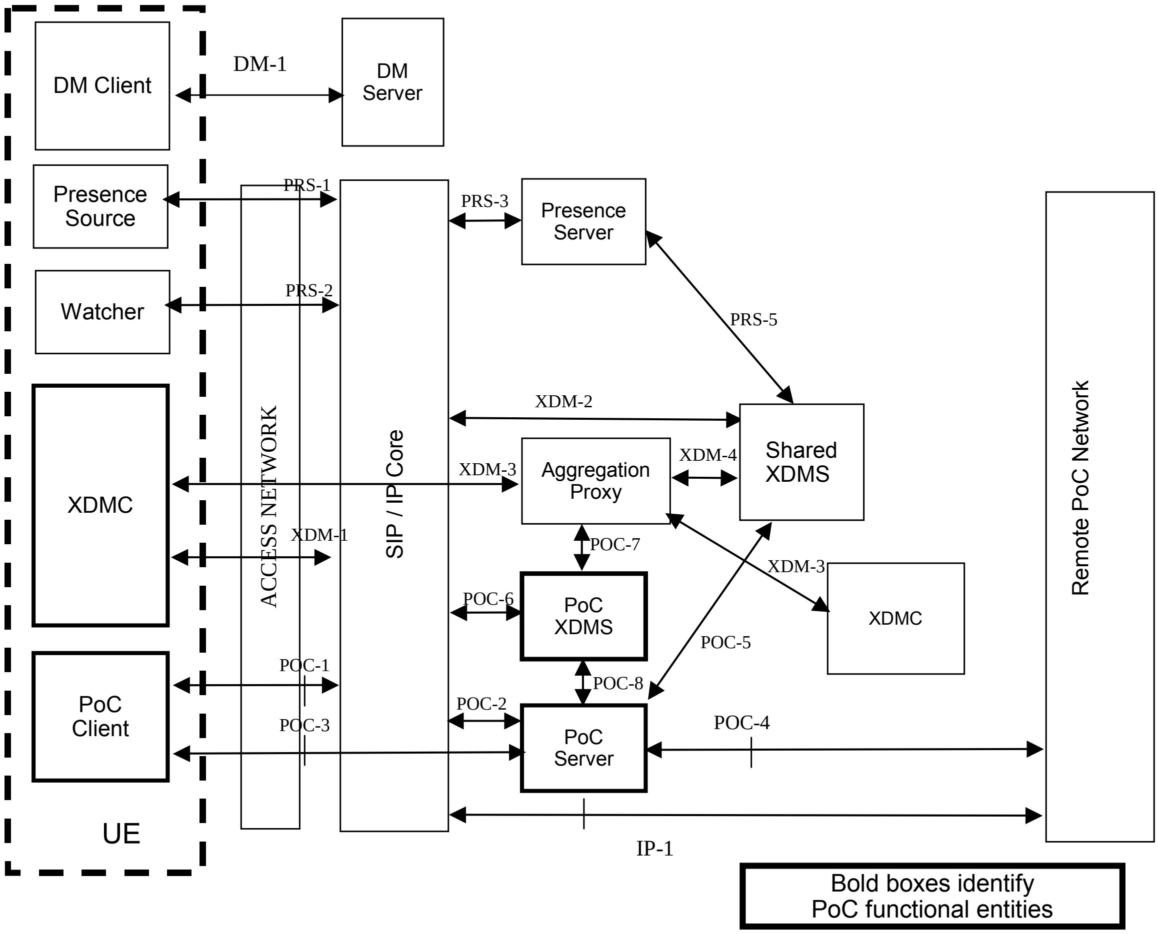


Figure 2-14: PoC architecture [112]

The PoC Client resides on the mobile terminal and is used to access the PoC service. The XML Document Management Client (XDMC) is an XCAP client which manages XML documents stored in the network (e.g. PoC-specific documents in the PoC XDMS, URI lists used as e.g. Contact Lists in the Shared XDMS, etc). Management features include operations such as create, modify, retrieve, and delete. The XDMC is also able to subscribe to changes made to XML documents stored in the network, such that it will receive notifications when those documents change. The XDMC can be implemented in a UE or fixed terminal.

2.6.1 PoC Server

The PoC Server implements the application level network functionality for the PoC service. The PoC Server performs a Controlling PoC Function and/or Participating PoC Function. The Controlling PoC Function and Participating PoC Function are different roles of the PoC Server.

The determination of the PoC Server role (Controlling PoC Function and Participating PoC Function) takes place during the PoC Session setup and lasts for the duration of the whole PoC Session. In case of 1-1 PoC Session and Ad-hoc PoC Group Session the PoC Server of the inviting User performs the Controlling PoC Function. In case of the Chat PoC Group and Pre-arranged Group Session the PoC Server owning/hosting the Group Identity performs the Controlling PoC Function.

The PoC Server performing the Controlling PoC Function normally also routes media and media-related signalling such as Talk Burst Control messages to the PoC Client via the PoC Server performing the Participating PoC Functioning for the PoC Client. However, local policy in the PoC Server performing the Participating PoC Function allows the PoC Server performing the Controlling PoC Function to have a direct communication path for media and media-related signalling to each PoC Client. Figure 2-15 shows the signalling and media paths in this configuration for a Controlling PoC Function, Participating PoC Function and PoC Client served in the same network. A PoC Server performing the Participating PoC Function has always a direct communication path with a PoC Client and a direct communication path with the PoC Server performing the Controlling PoC Function for PoC Session signalling.

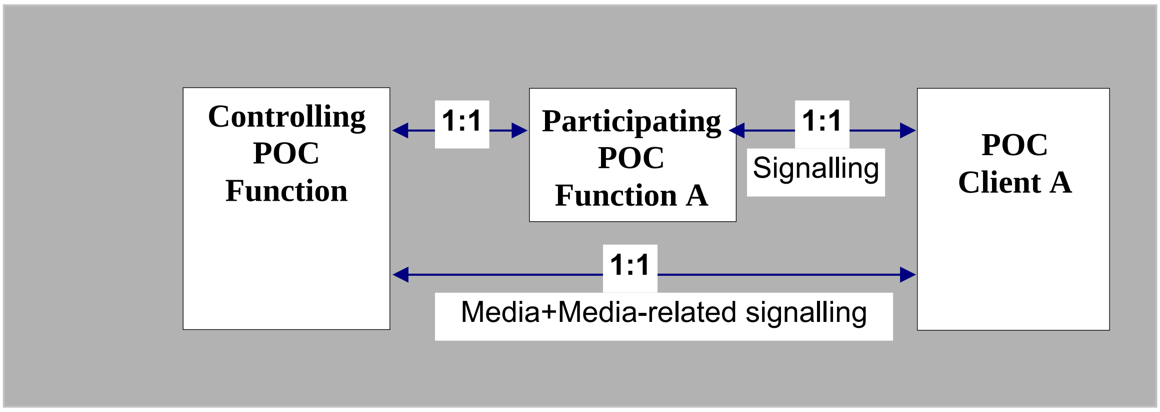


Figure 2-15: Direct media flow between Controlling PoC Function and PoC Client [112]

2.6.2 Controlling PoC Function

The PoC Server performs the following functions when it fulfils the Controlling PoC Function [112]:

a) Provides centralized PoC Session handling,

b) Provides the centralized media distribution,

c) Provides the centralized Talk Burst Control functionality including Talker Identification,

d) Provides SIP Session handling, such as SIP Session origination, release, etc.,

e) Provides policy enforcement for participation in Group Sessions,

f) Provides the Participants' information,

g) Provides for privacy of the PoC Addresses of Participants,

h) Collects and provides centralized media quality information,

i) Provides centralized charging reports,

j) Supports User Plane adaptation procedures,

k) Support Talk Burst Control Protocol negotiation.

2.6.3 Participating PoC Function

The PoC Server performs the following functions when it fulfils the Participating PoC Function [112]:

a) Provides PoC Session handling,

b) Supports the User Plane adaptation procedures,

c) Provides SIP Session handling, such as SIP Session origination, release, etc, on behalf of the represented PoC Client,

d) Provides policy enforcement for incoming PoC Session (e.g. Access Control, Incoming PoC Session Barring, availability status, etc),

e) Provides the Participant charging reports,

f) Supports Talk Burst Control Protocol negotiation,

g) Stores the current Answer Mode, Incoming PoC Session Barring and Incoming Instant Personal Barring preferences of the PoC Client,

h) Provides for privacy of the PoC Address of the Inviting PoC User on the PoC Session setup in the terminating PoC network.

The Participating PoC Function is performed once per PoC Client for all incoming/outgoing PoC Sessions. The Participating PoC Function may support Simultaneous PoC Sessions for the PoC Client. The Participating PoC Function may have 0 to M number of PoC Sessions for the PoC Client, where M is the maximum number of Simultaneous PoC Sessions permitted to a single PoC Client. The maximum number of possible Simultaneous PoC Sessions may be limited by the operator or the PoC Client configuration.

2.7 Instant Messaging in IMS

Instant Messaging (IM) is one of today’s most popular services. Many youngsters use this service to keep in touch with their relatives, friends, co-workers, etc.

Millions of instant messages are sent everyday. Thus, it is not a surprise that IMS already has this service well supported in its architecture.

IM is the service that allows an IMS user to send some content to another user in near-real time. The content in an instant message is typically a text message, but can be an HTML page, a picture, a file containing a song, a video clip, or any generic file. This service combines well with the foundation service of all services i.e., the presence service.

2.7.1 Modes of IM

There are two modes of operation of the instant messaging (IM) service, depending on whether they are stand-alone instant message, not having any relation with previous or future instant message. This mode of IM is referred to as “pager mode”. The model is also similar to the SMS (Short Message Service) in cellular networks. The other model is referred to as session based instant message that is sent as part of an existing session, typically established with a SIP INVITE request. Both modes have different requirements and constraints, hence the implementation of both models.

The IETF has created an extension to SIP that allows a SIP UA to send an instant message to another UA. The extension consists of a new SIP method named MESSAGE. The SIP MESSAGE method (RFC 3428 [207]), is able to transport any kind of payload in the body of the message, formatted with an appropriate MIME (Multipurpose Internet Mail Extensions) type. 3GPP TS 23.228 [24] already contains requirements for Application Servers (ASs) and S-CSCFs to be able to send textual information to an IMS terminal. 3GPP TS 24.229 [208] introduces support for the MESSAGE method extension. The specification mandates IMS terminals to implement the MESSAGE method [207] and to allow implementation to be an optional feature in

S-CSCFs and ASs. The flow is simple as depicted in Figure 2-16. The source IMS terminal sends the instant message via MESSAGE method and receives a 200OK response from the destination IMS terminal after the MESSAGE has been received. The diameter base protocol (RFC 3588, [41]) is used for the purpose of Authentication, Authorization and Accounting (AAA).

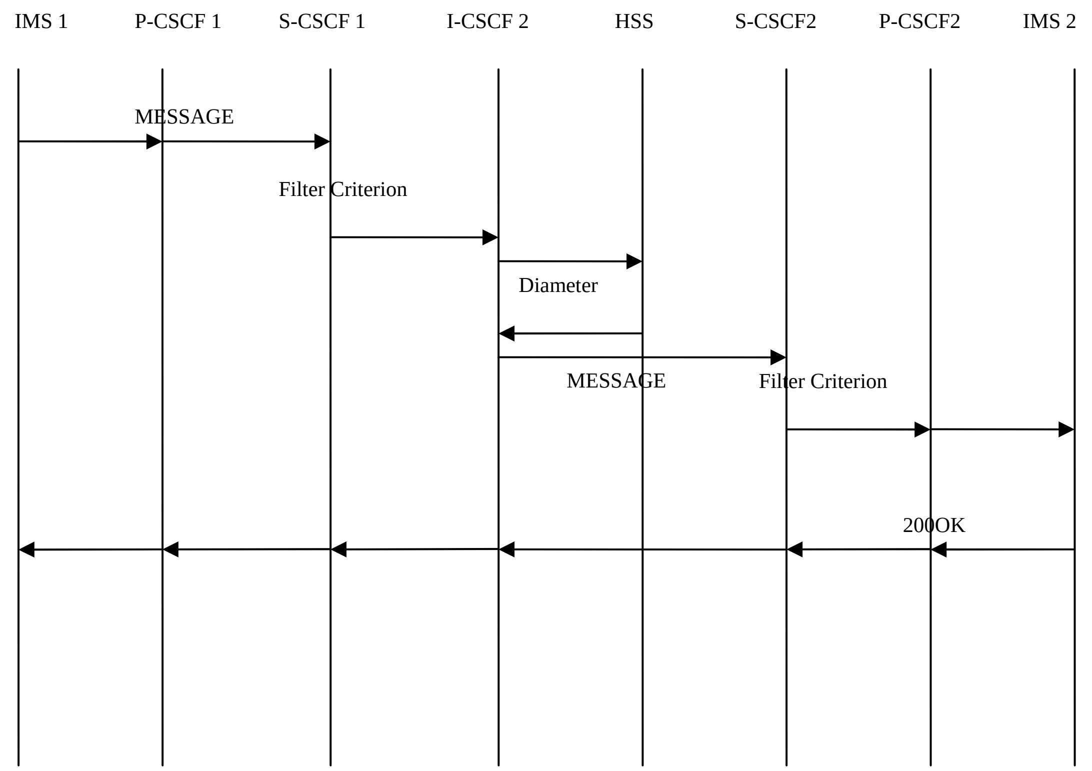


Figure 2-16: Pager-mode instant messaging in the IMS

Chapter 3 Literature Review

As stated before, the IP Multi-Media Subsystem (IMS) is defined by 3GPP and 3GPP2 standards and organizations based on IETF Internet protocols. The detail documentation on it is furnished in [24], [25], [26], [27], [205] and in [206]. This research is based on IMS stage 2, release 6 and release 7. IMS is access independent as it supports IP to IP session over wire-line IP, 802.11, 802.15, CDMA, packet data along with GSM/EDGE/UMTS and other packet data applications. It consists of session control, connection control and an applications services framework along with subscriber and services data. It enables new converged voice and data services, while allowing for the interoperability of these converged services between subscribers.

Some recent work on IMS can be found in the Bell Labs Technical Journal in [92-99, 134]. The crucial issues involved in these work are:

a. Providing seamless mobility for subscribers across the packet and circuit domains [92, 93, 95, 96];
b. Subscriber data management and data integration so that IMS applications can use single point of access for accessing user profile information inside a service providers network [97, 98, 134];
c. Lucent Technologies' SIPia BUS software architecture to maximize IMS server co-location [99];
d. Threats and vulnerabilities of IMS implementations as well as high level service provider security requirements to provide the desired level of security for IMS deployments [94].

We provide a break down of the literature review related to our research next.

3.1 Quality of Service Issues

Over the past few decades various Quality of Service (QoS) issues have evolved on wireless communication. Scheduling/queuing is one of the areas that drive researchers to ameliorate performance analysis of wireless network. The scheduling can be found in top to bottom layers of the network in order to achieve efficient network admission control for instance, in terms of reliability, energy efficiency or resource utilization etc. Two different approaches may be distinguished as far as admission control is concerned: reservation-based and measurement-based. In the first approach, new flows specify their QoS requirements along with their traffic descriptors through a signalling protocol such as Resource Reservation Protocol (RSVP). The amount of resources to be allocated to an incoming flow is computed accordingly. In the measurement-based approach, resources are not dedicated to a given flow. Hence, the admission criterion does not depend on the amount of reserved resources, but on their real utilisation for instance, a link. Recently there has been a growing interest in applying admission control to elastic flows of classified traffic.

Network Dimensioning is another key capacity planning discipline, which has a direct impact on a network's cost base. An excess of deployed capacity will result in wasted capital, while a dearth of capacity will adversely impact service level agreements, potentially incurring service penalties. The key features of the Network Dimensioning service include determining the appropriate sizing approach required for the network, defining key inputs for the dimensioning exercise, such as a traffic demand matrix, routing configuration files, network topology, resilience requirements, etc., optimising the routing of traffic by using features such as traffic engineering, defining a bandwidth augmentation strategy and characterisation of network workload etc.

Mobility management is a significant field where researchers are putting much effort. Perkins was among the first few who introduced mobility support in Internet

Protocol (RFC 3344, [3]). Mobility support in IP worked like a tonic for other prominent protocols in communications. It is still a significant area for the researchers. The mid call mobility management has been observed in quite a few work. Reducing message overhead and location based performance analysis are other key factors today in mobile environment.

Application of queuing theories is yet another aspect in the field of multimedia communications. Our work in this dissertation is centred on these aforementioned issues of admission control mechanisms, dimensioning services, pre-session mobility management and application of queuing theories.

3.1.1 Message formats in IMS presence service

The Presence Information Data Format (PIDF) is a protocol-agnostic document that is designed to carry the semantics of presence information across two presence entities. The PIDF is specified in the Internet-Draft “Presence Information Data Format (PIDF)” (RFC 3863, [50]). The PIDF encodes the presence information in an XML (Extensible Mark-up Language) document that can be transported, like any other MIME (Multipurpose Internet Mail Extension) document, in presence publication (PUBLISH transaction) and presence subscription/notification (SUBSCRIBE/NOTIFY transaction) operations. The Rich Presence Information Data Format (RPID) is an extension to the PIDF that allows a presentity to express detailed and rich presence information to his/her watchers. Like the PIDF, RPID is encoded in XML. The RPID extension is specified in (RFC 4480, [52]).

A presentity like Alice for instance can set her rich presence information by manually operating on the appropriate setting of her presence software. However, RPID allows an automation that has access to the presentity’s presence information to set such information up automatically. For instance, a calendar application can automatically set the presentity’s presence information to “online- in a meeting” when the presentity’s agenda indicates so. A SIP phone can automatically update the presentity's presence information to indicate that the presentity is engaged in a call when the presentity answers the phone. The RPID contains one or more activity elements that indicate the activity the presentity is currently doing. The specification allows the activity element to express that the presentity is on the phone, away, has a calendar in a meeting, steering a vehicle, in transit, travelling, on vacation, sleeping, just busy, or on permanent absence. For instance, a place-type element in the RPID indicates the presentity currently in. the possible initial values are home, office, library, theatre, hotel, restaurant, school, industrial, quiet, noisy, public, street, public transport, aircraft, ship, bus, train, airport, station, mall or outdoors etc. The list of values is expandable for future extensions. Figure 3-1 shows an example of the presence information that Alice provides to her watchers.


```
<?xml version=”1.0” encoding=”UTF-8”?>
<presence xmlns=”urn:ietf:params:xml:ns:pidf”
        xmlns:es=”urn:ietf:params:xml:ns:pidf:rpid-status”
        xmlns:et=”urn:ietf:params:xml:ns:pidf:rpid-tuple”
        entity=”pres:alice@example.com”>

    <tuple id=”3bfua”>
        <status>
          <basic>open</basic>
          <es:activities>
                <es:activity>meeting</es:activity>
          </es:activities>
          <es:place-type until-“2006-01-17T11:30:00Z”>
                        Home</es:place-type>
          <es:privacy>quiet</es:privacy>
          <es:idle>2006-01-17T09:4600Z</es:idle>
          <es:sphere from=“2006-01-17T09:00:00Z“>work</sphere>
        <status>
        <et:class>sip</et:class>
        <et:contact-type>service</et:contact-type>
        <contact priority=”0.8”>
                sip:alice@laptop.example.com
        </contact>
        <timestamp>2006-01-17T10:32:16Z</timestamp>
    </tuple>
```

```
   <tuple id=”vusa44”>
      <status>
        <basic>open</basic>
        <es:privacy>quiet</es:privacy>
      </status>
      <et:class>phone</et:class>
      <et:contact-type>device</et:contact-type>
      <contact priority=”0.8”>
              im:user_public@dodo.com
      </contact>
      <timestamp>2006-01-17T10:32:15Z</timestamp>
  </tuple>

   <tuple id=”tan45”>
      <status>
        <basic>open</basic>
      </status>
      <et:class>mail</et:class>
      <et:contact-type>device</et:contact-type>
      <contact priority=”3.0”>
              mailto:malam@bond.edu.au
      </contact>
   </tuple>

      <note>I am working on IMS at home</note>
</presence>
```

Figure 3-1: Example of the RPID

The first tuple in Figure 3-1 indicates her own presence information to be active or open, but at the meeting etc. The second tuple conveys the presence information of her phone while the 3rd indicates a mail contact where she could be reached via email. After a presentity publishes its presence to its Presence Agent (PA) / Presence Server (PS) via RPIDs, the PS keeps the presentity’s watchers updated with NotifyPresUp messages (see Figure 3-2).

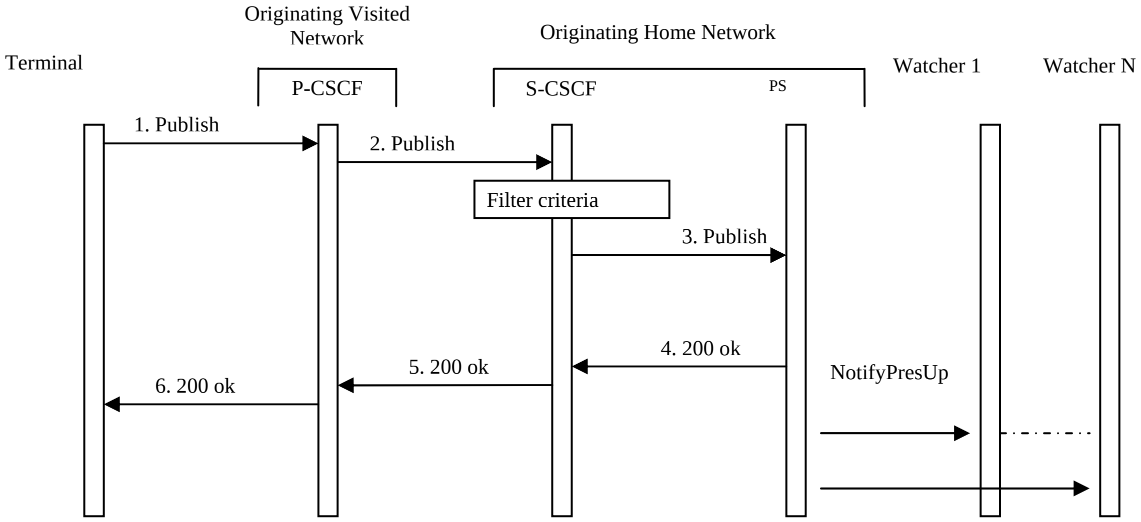


Figure 3-2: Publishing and notifying presence information [49]

A watcher receives NotifyPresUp messages from the PS based on the RPID, every time a presentity of its list changes state. These XML documents with presence information can be rich in data compared to the processing capacity of a small wireless device. Obviously, this mechanism does not scale well, particularly in wireless environment since the heavy transmission rate can easily overload an IMS network with message flows. Our objective is to propose an efficient scheduler for the PS in heavy traffic situation.

3.1.2 Subscription / Registration time

The detail of SIP and MIP registration can be located in (RFC 3261, [11]) and (RFC 3775, [12]) respectively. Related work can be found in [3] (RFC 3344), [87]. Multi-cast support for MIP with Hierarchical local registration has been presented by Omar *et al* (2000) in [89]. Several optimization schemes on location update procedure and sending binding lifetime in MIPv6 in terms of costs can be found in [32], [33], [34], [35] and in [90]. However, they do not mention optimizing the registration life time.

The Timed Presence extension is specified in RFC 4481, “Timed Presence Extension to the Presence Information Data Format (PIDF) to indicate Presence Information for Past and Future Time Intervals” [53] and allows a presentity to express what they are going to be doing in the immediate future or actions that took place in the near past. A timed-status element that contains information about the starting time of the event is added to the PIDF XML document. The starting time of the event is encoded in a ‘from’ attribute, whereas an optional ‘until’ attribute indicates the time when the event will stop. Figure 3-3 shows an example of the time status extension. Here, Alice is publishing that she will be offline from 13:00 to 15:00.

```
<?xml version=”1.0” encoding=”UTF-8”?>
<presence xmlns=”urn:ietf:params:xml:ns:pdf”
          xmlns:ts=”urn:ietf:params:xml:ns:pidf:timed-status”
          entity=”pres:alice@example.com”>

  <tuple id=”qoica32”>
     <status>
          <basic>open</basic>
     </status>
     <ts:timed-status from=”2004-02-15T13:00:00.000+02:00”
          Until=”2004-02-15T15:00:00.000+02:00”>
          <basic>closed</basic>
     </ts:timed-status>
    <contact>sip:alice@example.com</contact>
  </tuple>
</presence>
```

Figure 3-3: Example of the timed status extension

A subscription can last for a period of time. If watchers want to keep the subscription active they need to renew it prior to its expiration. The PS will keep the PUA/IMS user updated, using NOTIFY requests about changes in the list of watchers. That is, it will inform a presentity every time a new watcher subscribes or un-subscribes to the presentity’s presence information. Every time a watcher wants to subscribe to the presence information of a presentity, the watcher needs to exchange a SUBSCRIBE transaction and a NOTIFY transaction with the presentity's PUA, just to set up the subscription. Obviously, again this mechanism does not scale well, particularly in wireless environment for small devices.

3.1.3 Presence Optimizations by IETF

In order to solve these above-stated problems of frequently notifying watchers (via NotifyPresUp message) due to the presentities' state change and notifying Presentities (via NOTIFY message) due to the watcher subscription time expiration, the IETF has created a number of concepts as described below.

1. The concept of resource lists is one of the mechanisms to reduce excessive signals. A resource list is a list of SIP URIs that is stored in a new functional entity called the Resource List Server (RLS) as introduced in Figure 2-9 and in Figure 2-10 (section 2.5), sometimes known as an exploder for SUBSCRIBE requests. A SIP exploder receives a request from a user agent and forwards it to multiple users. SIP exploders used for subscriptions are described in RFC 4662 [54].

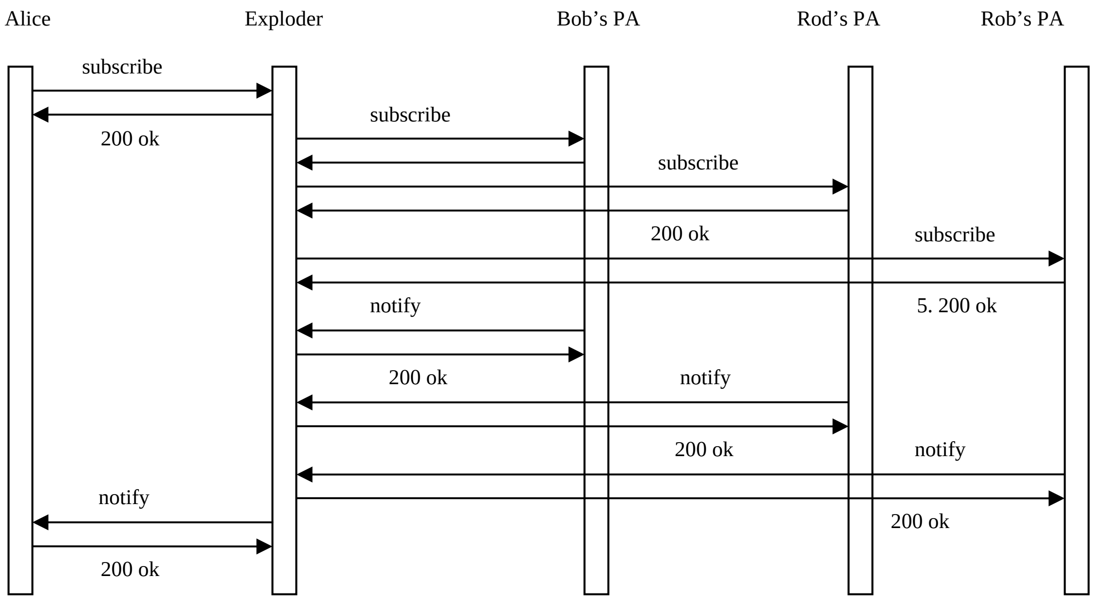


Figure 3-4: Resource list through an exploder

Figure 3-4 shows how this type of exploder works. Instead of sending a SUBSCRIBE request to every user in the presence list, Alice sends a single SUBSCRIBE request addressed to her presence list. The request is received by the SIP exploder, or RLS. Alice has previously provided the exploder, using an out-of-bound configuration mechanism of her choice, with her presence list. The exploder sends a request to every user in the list. Later when the exploder receives the NOTIFY requests from them, it aggregates the presence information and sends a single NOTIFY request to Alice. Although the mechanism saves bandwidth on a user's access network, the signalling impact is still there for massive number of publishers and watchers.

2. Event filtering (RFC 4660, [135]) is one mechanism on which IETF engineers are working to reduce the amount of presence information transmitted to watchers. A weight or preference is indicated through a SUBSCRIBE request. The mechanism defines a new XML body that is able to transport partial or full state. Thus, the document size is reduced at the cost of information transmitted. Sending less information in presence documents may lead to IMS users not getting a good experience

with presence systems used from wireless terminals. Also, the implementers need to be aware of the computational burden on the PS.

3. Event-throttling mechanism [136] allows a subscriber to an event package to indicate the minimum period of time between two consecutive notifications. So, if the state changes rapidly, the notifier holds those notifications until the throttling timer has expired. Usually, the PS will buffer notifications that do not comply with the throttle interval, and batch all of the buffered state changes together in a single notification when allowed by the throttle. The throttle applies to the overall resource list [54], which means that there is a hard cap imposed by the throttle to the amount of traffic the presence application can expect to receive. With partial-state notifications, the notifier will always need to keep both a copy of the current full state of the resource F, as well as the last successfully communicated full state view F' of the resource in a specific subscription. The construction of a partial notification then involves creating a difference of the two states, and generating a notification that contains that difference. When a throttle is applied to the subscription, it is important that F' is replaced with F only when the throttle is reset. Additionally, the notifier implementation checks to see that the size of an accumulated partial state notification is smaller than the full state, and if not, the notifier sends the full state notification instead. The disadvantage is that batching and matching will introduce additional processing delay in the PS. Currently, a subscription refresh is needed in order to update the throttle interval. However, this is highly inefficient, since each refresh automatically generates a (full-state) notification carrying the latest resource state. In addition, with this mechanism the watcher does not have a real-time view of the subscription state information. Moreover, holding the information will require additional buffer space. Nonetheless, this policy may be helpful for IMS terminals with low processing power capabilities, limited battery life or low bandwidth accesses. We will discuss the tradeoffs of such service later in this thesis.

4. Compression of SIP messages is another technique to minimize the amount of data sent on low-bandwidth access. RFC 3486 [55], RFC 3320 [56], RFC 3321 [57] defines signalling compression mechanisms. Usually these algorithms substitute words with letters. The compressor builds a dictionary that maps the long expressions to short pointers and sends this dictionary to the de-compressor. However, the frequency of data transmission is not reduced in such techniques.

3.1.4 Related work on PoC service

This section focuses on one of the other services in IMS, the Push-to-Talk over Cellular (PoC) service. The PoC application allows point-to-point, or point-to multipoint voice communication between mobile network users [137]. The communication is strictly unidirectional, where at any point of time only one of the participants may talk (talker), all other participants are listeners. In order to get the right to speak, listeners first have to push a “talk” button on their mobile terminals. Floor control mechanisms ensure that the “right to speak” is arbitrated correctly between participants. The PoC application may become a highly popular service for the mobile telecommunications market if its responsiveness and voice quality meet end-user expectations.

The value of Push-to-talk increases when it is well integrated with other available services and enablers. The integration effort decreases if common functionality, protocols and system principles can be applied across application borders. This speaks in favour of standardized solutions. In order to become a truly successful mass-market service for the consumer segment, the only realistic alternative is a standardized Push-to-talk solution providing full interoperability between terminals and operators.

Since, PoC is one of the emerging technologies the literature review on technical part of it is slim so far. The related work available today focuses on the performance analysis over PoC mostly. Parthasarathy implemented a prototype of a Push to talk Server as a Java application on 2.5 networks [140]. However, the prototype is not complete and does not support all the PoC features. Some strategic actions related to standardization, system architecture, vendor's product strategies, substitutes etc. are discussed in [111]. A solution for voice group communication in mobile ad hoc networks has been implemented and tested by Hafslund *et al* (2005) in [142]. Their system reuses the optimized flooding techniques from the OLSR (Optimized Link State Routing) protocol. This minimizes the number of forwarding nodes, and thus also the total network load. The group communication system is best suited for broadcasted voice traffic in dense mobile ad hoc networks. The system was implemented and tested for a real life test-bed, based upon Linux routers with 802.11b wireless LAN.

An architecture for enabling PoC services in 3GPP networks has been furnished by Raktale S. (2005) in [139]. The performance of PoC signalling transfer is been evaluated using NS2 simulator. The focus was on the impacts of PoC requirements on 3GPP UTRAN. The system simulation setup and findings are provided in Figure 3-5 and Table 3-1 below.

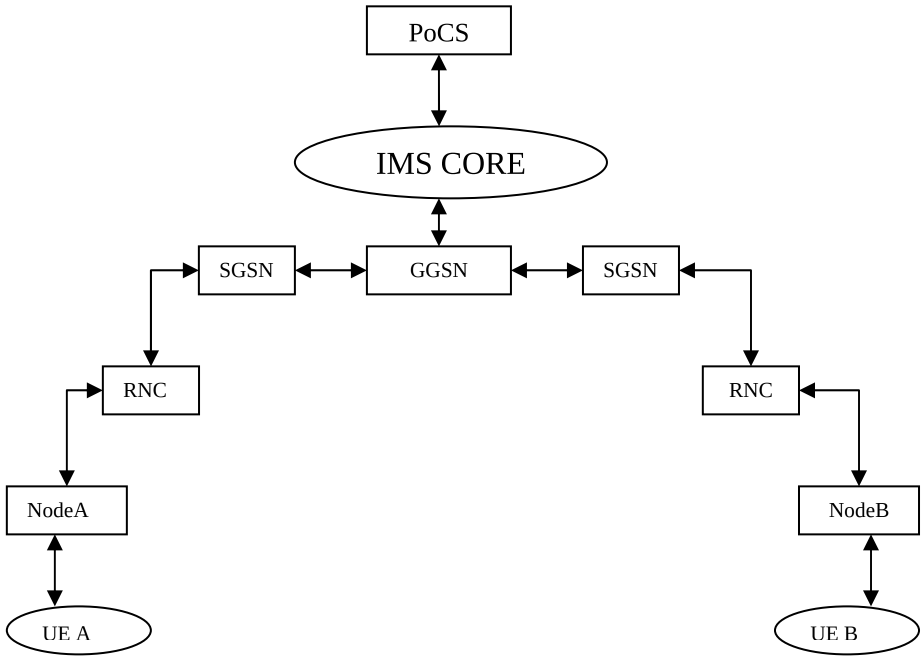


Figure 3-5: Simulation architecture of Raktale [139]

Table 3-1: PoC Call setup performance [139]

Session Type	**Total Delay (sec)**
On-Demand Session with opportunistic call setup	2.8
On-Demand Session with guaranteed call setup	3.5
Pre-established Session with opportunistic call setup and single PDP context	2.1
Pre-established Session with guaranteed call setup and dual PDP context	2.6
Pre-established Session with opportunistic call setup and dual PDP context	2.4
Pre-established Session with guaranteed call setup and dual PDP context	2.9

Raktale's (2005, [139]) work is an analysis of call set up performance of two kinds of session in PoC service: (1) on-demand and (2) pre-established session. The results of Table 3-1 suggest that on-demand session is out-performed by pre-established session in terms of set up delay. The pre-established PoC session provides a mechanism to negotiate media parameters such as IP address, ports and codecs, which are used for sending the media and Talk Burst Control messages between the PoC client and the Home PoC server. The mechanism allows the PoC client to invite other PoC clients or receive PoC sessions without negotiating again the media parameters. The pre-established session is established after the initial registration where as registration is performed at the same time of establishing on-demand session. This is the reason on-demand session consumes more time to set up which is evidence from Raktale's work. The Figure 3-6 presents the high level description of the pre-established session procedure.

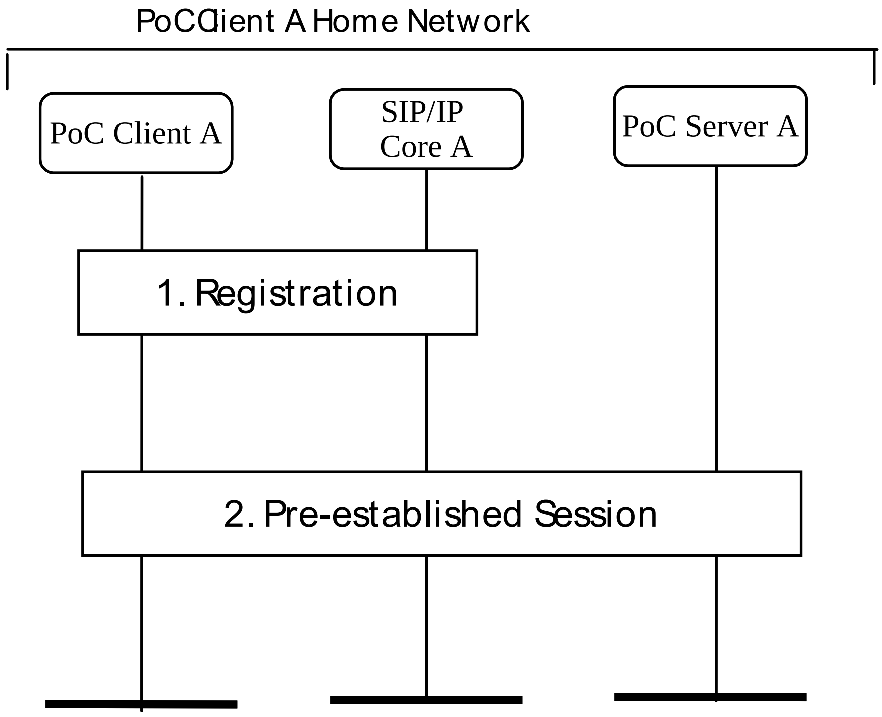


Figure 3-6: Pre-established Session [112]

The steps involved in the pre-established session are [112]:

1. The PoC client registers to the SIP/IP Core.
2. The pre-established session is a session establishment procedure between the PoC client and the PoC server to exchange necessary media parameters needed for setting up the media bearer. After the pre-established session is established the PoC client is able to activate media bearer whenever needed:
    - immediately after the pre-established session procedure or;

- when the actual SIP signalling for the PoC session is initiated.

In case of on-demand session, the session is usually very short and disconnected after the data flows. On the other hand, the pre-established sessions are long and these sessions maintain state change depending on whether sessions are active or not. The pre-established PoC sessions will generate more message flows than the on-demand PoC sessions in the long run though the former provides faster session initiation due to early registration. Also with pre-established session, a PoC is allowed to set up as many sessions as it wants which should be hard capped in busy time. We discuss the two session set up issues more in Chapter 5. The related works do not address the issue of controlling these two types of session access during busy traffic for a PoC service. The Northstream report suggests that the PoC server performance can be measured through number of TRUs (Transmit / Receive Unit) [138]. Thus the session access priority should be assigned based on available TRUs in a network cell.

The current available works on PoC services are also ignorant on issues like PoC session timer settings, optimizing number of simultaneous sessions and PoC traffic overflow etc. A PoC client prototype has been implemented based on OMA v.10 release by Lin-Yi Wu *et al* (2006) in [143]. The design of a PoC service operated over a GPRS/UMTS (General Packet Radio service / Universal Mobile Telecommunications System) network is depicted by Kim *et al* (2005) in [141]. The PoC performance is analysed over GPRS by Balazs (2004) in [137]. The impacts of mobile network elements are analysed in terms of delay and bandwidth along the end-to-end transport path of GPRS networks. Again, these works emphasize on performance analysis and lack the issue of dimensioning a PoC service completely.

3.1.5 Mobility management in IPv6

Neumann *et al* (2003) implemented a prototype and evaluated the performance of a QoS conditionalized handoff scheme for mobile IPv6 networks [6]. The work shows that QoS-enabled handoffs can be achieved with a small amount of introduced latency compared to Hierarchical Mobile IPv6, which is much less than that of Mobile IPv6. Although fewer packets were found to be lost, their scheme needs to interact with an end-to-end QoS signalling solution. Urien *et al* (2002) proposed a network management protocol by policies with Common Open Policy Services (COPS) for both macro and micro mobility [7]. It seems their architecture solves the mobility in IP network with a soft handover mechanism. However, the protocol needs to be validated to evaluate its performance. The performance of IPv6 network mobility is measured in [177]. A fast handover algorithm for Hierarchical Mobile IPv6 macro-mobility management is introduced in [8]. The algorithm minimizes the disruption delay that occurs in handover process. In this mobility management, MN acquires two new addresses, a new RCoA (regional care of address) and LCoA (link care of address). These addresses are registered at the HA or CN by sending a Binding Update to the HA or CN. The macro-mobility is provided by the MAP (Mobile Access Points) in the networks via multicasting technique that passes the information to the neighbouring nodes. Although the technique is very useful in macro mobility handover, it does not provide mobility support in micro-environment. The real-time applications will be one of the domain types of traffic transported through Mobile IPv6. Work on real-time traffic in differentiated services network has been done in [9]-[10]. The work of Yousof and Fisal (2003, [10]) provided the acceptable fairness of services for better QoS support for real-time traffic based on scheduling algorithm named Round Robin Priority Queuing (RRPQ). But the implementation of RRPQ in differentiated services is yet to commence.

Performance evaluation of network and application layer multicast over MIPv6 networks and IPv6 handover techniques over wireless LAN have been analysed in [16], [17]. Comparison between IP multicast and application layer multicast have been performed by Finney *et al* (2003, [17]) under a specific assumption: end hosts are wireless devices using MIPv6 protocol. Their work suggests that the advantage of using IP multicast grows stronger in mobile networks while the packet loss increases for application layer multicast. Nevertheless, the work was limited within the multicast technique only. The handover latency was calculated for basic MIPv6, the forwarding method of MIPv6, the anticipated method of MIPv6 and the tunnel-based of MIPv6 in [16]. The throughput and number of users were varied to get useful insight into the handover behaviours. Fast handover was found to offer shorter disruption times. However, duplicate address detection was not taken into account in their experiment which might introduce greater disruption time. Also, the test was performed for wireless LAN only.

The IETF mobile IPv6 (MIPv6) enables correspondent nodes (CNs) to directly send packets to a mobile node (MN) using care-of address of the MN. For this service, however, MNs always have to inform CNs and the home agent (HA) of its new location at each movement. To reduce this control signalling, the existing hierarchical scheme built on top of the MIPv6 separates micro-mobility from macro-mobility and exploit an MN's locality. The hierarchical scheme does not achieve real optimization of packet routing. Packets from CN to MN are delivered through an intermediate mobility agent. It brings needless delay on packet delivery and imposes heavy loads on the intermediate mobility agent. In [15], S. Hwang *et al* (2003, [15]) proposed a new hierarchical scheme that enables any CNs to send packets to an MN without helps of the intermediate mobility agent using a subnet residence time in the profile. This proposal can reduce delay in packet delivery and optimize packet delivery routing. Furthermore, it can alleviate heavy loads on the intermediate mobility agent. The research compared registration and packet delivery costs between Hierarchical Mobile IPv6 and their proposed mechanism. However, the registration cost becomes very high in their work if the probability to select a local care-of-address when receiving a BU (Binding Update) from an MN (Mobile Node) is high and if there are too many CNs communicating MN. Also, none of the above works emphasizes similar comparison on the session set up issue in MIPv6.

3.1.6 Mobility management in SIP

Considering the fact that mobile IP may not provide fast enough handoffs to support rich data communications, much work can be observed to be performed on other signalling protocols like SIP that may provide a better solution. Location management and handoffs over SIP have been key areas where researchers worked on lately. [28-30] investigate mobility support of SIP in different environments. Wedlund and Schulzrine (1999, [28]) proposed to use mobility support in the application layer protocol SIP where applicable in order to support real-time communication in a more efficient way. In their proposed architecture, a mobile policy table is used for deciding what source address to use (home or care-of address) whether it should be tunnelled, or even use a bidirectional tunnel. Moving the mobility handling to the application layer, eliminates the need for tunnelling of the data stream. Moreover, the fact that SIP mobility is at the application layer, means that it can be installed easily. They also described the traditional hierarchical registration mechanism in SIP (Figure 3-7).

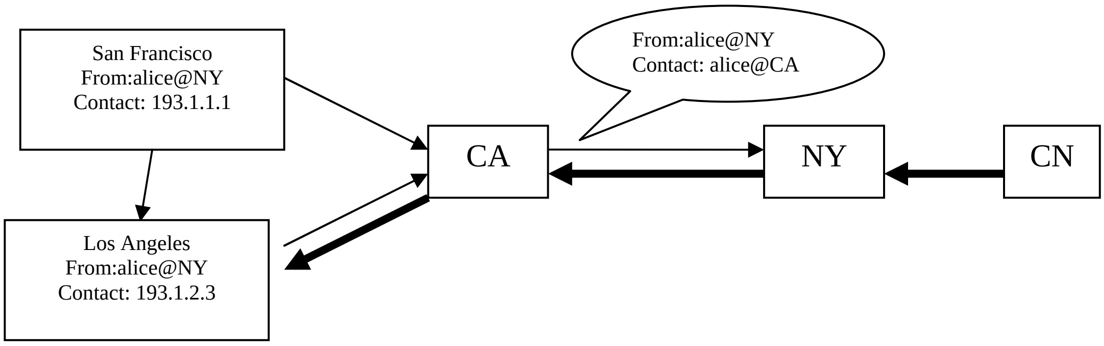


Figure 3-7: Hierarchical registration in SIP [29]

In Figure 3-7, Alice with a home in NY, visits CA. Each time she moves, she sends a REGISTER request towards her home register, through the out bound proxy in CA. For the first REGISTER, originating in San Francisco, the outbound proxy makes a note of the registration and then forwards the request to the normal home register, after modifying the Contact in the registration to point to it rather than Alice's mobile host. After Alice travels to LA, the REGISTER update hits the same register (CA). It recognizes that Alice is already in CA and does not forward the request. A call from anywhere first reaches the NY proxy server, which forwards the request to the CA proxy server, which in turn forwards it to Alice's MH (mobile host). The details of SIP proxy behaviour can be found in [31]. Moh *et al* (1999, [30]) emphasized the ability of SIP to compare with H.323 in the support of mobile telephony over the Internet addressing the issues of registration in roaming and location management. A similar SIP-based route optimization technique can be located in [178].

Much work has been done on the standard QoS part of SIP. QoS control by means of Common Open Policy Service (COPS) to support SIP-based applications has been demonstrated in [43]. COPS protocol was defined by IETF working group mainly to support policy control in an IP QoS environment. Salsano and Veltri (2002, [43]) proposed a COPS based model to provide admission control scheme in SIP-based IP telephony applications that can use Diffserv-based QoS network. A test bed implementation of the proposed solution was described. Issues related to secure remote appliance control using SIP was mentioned in [42]. A mechanism for Dynamic Resource Allocation (DRA) in 3GPP SIP overlay networks has been introduced in [44]. The mechanism can also be used in virtual SIP links. The aim of DRA is twofold. Firstly, it is a methodology to enable the QoS provisioning for the virtual SIP signalling network. Secondly, it achieves the dimensioning automatically on the fly. It uses capabilities that mixed services IP transport networks provide. Harris and Kist (2003, [44]) argued that since the DRA methodology allows the automated configuration of resources and ensures QoS for signalling, it enables the guarantee of QoS to customers in UMTS networks. Kueh, Tafazolli and Evans (2003) evaluated the performance of SIP-based session set up over satellite Universal Mobile Telecommunications Systems (UMTS) in [69]. Similar work needs to be performed in IMS environment.

An approach to replicate SIP call control functionality over a number of dispersed hosts has been proposed in [45]. SIP service users and providers require fault-tolerance with high service availability and reliability. In order to allow for mid-call fail-over, call states need to be replicated, but this may cause call state inconsistency. The trade-off relationship between SIP transaction inconsistency and read delay exploited the authors in [45] to derive the algorithm that is easily adapted by the SIP traffic networks. Kist and Harris (2003) argue to use virtual SIP links to enable QoS provisioning in SIP signalling overlay networks [51]. Their methodology includes the well-known leaky bucket concept to calculate the message loss probabilities. They also introduced a queuing scheme that reduces the required network resources. However, none of the above works proposes to optimize the cost for required resources in the network; neither they include the impacts by DiffServ environment.

3.1.6.1 HMSIP architecture

An efficient Hierarchical Mobile SIP (HMSIP) micro-mobility management support in SIP environment has been proposed by Vali *et al* (2003) in [63]-[64]. HMSIP aims at reducing handoff latency and minimizing signalling overhead in the backbone network, by restricting intra-domain handoff related signalling inside the roaming domain. All types of IP traffic are handled by HMSIP, including TCP flows. HMSIP relies on Mobile SIP functionality for inter-domain mobility, much like the various network layer micro-mobility schemes rely on Mobile IP for global roaming. Their proposal follows the general regional registration approach found in various proxy-Agent micro-mobility schemes (e.g. HMIPv6 in RFC 4140 [65], IDMP [66]) and builds on the SIP Hierarchical Registration proposed in [29]. A key entity in HMSIP architecture is the HMSIP Agent. It is a SIP Mobility Agent that is located at the domain border. The HMSIP Agent contains the necessary intelligence for localizing the intra-domain mobility related signalling and performing fast data path redirection to the current location of the mobile. Its functionality may be distributed across various domain border entities, as it is shown in Figure 3-8.

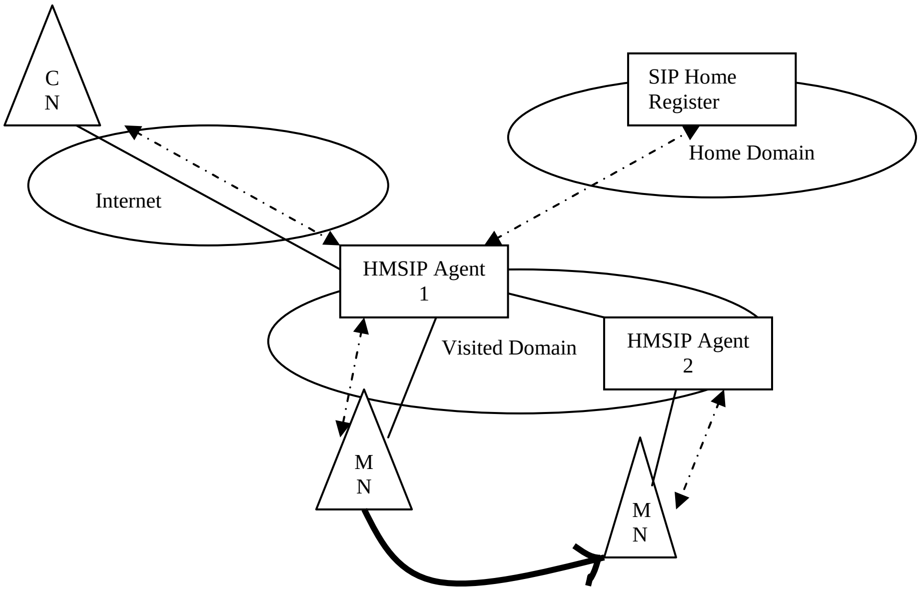


Figure 3-8: HMSIP architecture for intra-domain handoff

Similarly to other micro-mobility approaches, HMSIP allocates two IP addresses to a mobile node (MN) entering a visited domain, a Local Address (LA) and a global Domain Address (DA). The LA is an IP address that reflects the current point of attachment of the MN and is routable inside the visiting domain. It may even be a private address inside the domain. It is allocated to the MN by the serving access router. After a handoff to a new access router, the MN always gets a new LA. The DA is a globally routable IP address assigned to a MN that does not change as long the MN roams inside an access domain and has active sessions. The DA is allocated to the MN by the serving HMSIP Agent, drawing it from a pool of public IP addresses. The global IP address assigned to the mobile host remains unmodified throughout its active communication inside the roaming domain. The existence of a stable DA further allows for smooth inter-working with non-mobility aware protocols such as QoS enabling Resource Reservation Protocol (RSVP, RFC 2205, [67]).

Some standard QoS work over IMS can be located in [72], [73]. Borosa *et al* (2003, [72]) presents some aspects of provisioning QoS in the IMS environment based on DiffServ and scalable IP-based QoS technology. Their proposal works for both real time and non real time IMS applications. SIP call set up delay in 3G networks and hand-off delay in wireless networks have been investigated in [70], [71]. The research in [70] shows several simulations on answer-signal delay, call-release delay, post-dialling delay etc. in SIP static environment only. Das *et al* (2003, [71]) analysed the SIP based hand-off delay in wireless networks which suggests that SIP is not suitable for supporting streaming media with stringent delay requirements. Nonetheless, these works do not refer to the session set up delay when an end node is mobile.

3.1.7 MIP and SIP Interactions

Quite a handful work has been done on MIP and SIP interactions in [81-85]. The benefits for SIP and IPv6 interactions have been well depicted in [83]. The most obvious reason for using SIP with IPv6 is naturally the huge amount of available addresses. This is especially important when considering 3G architectures with millions of SIP based mobile phones all requiring their own IP addresses. But phones are not the only IP capable devices from the SIP point of view. Internet-capable gaming stations or even appliances are also thought to be triggered by SIP. Besides this obvious reason, IPv6 can be very helpful while SIP requiring dynamic configuration in a standardized manner. Also, for a user to start a communication session it might need to send all its SIP messages to a register or an outbound SIP proxy which might be responsible for the authentication of the user or controlling a firewall. Finding out the location of this registrar or outbound proxy might be statically configured in the user agents. A more flexible solution is to have all proxies with similar functionalities under the same anycast address. In this scenario the messages will get directed to the closest entity. Some useful simulation results have been presented by Nakajima *et al* (2003) in [88] in terms of hand-off delay analysis while SIP interacts with IPv6 in an IEEE 802.11b wireless LAN laboratory test-bed. IMS has adapted SIP and MIPv6 interactions in the latest releases.

Takahashi *et al* (2003, [82]) proposed two IPv4/IPv6 SIP interaction methods (IP-version dependent routing and register method) that eliminate unnecessary IPv4/IPv6 translations in duel-stack network. The IP version dependent method was found to perform the best among all. However the weak point is that it uses SIP application layer gateway (SIP-ALG) that performs special routing. The detail of SIP-ALG is described in [86]. A comparison between SIP and MIP shadow registration delays have been shown in [85]. The information of a mobile node sending its location information to the neighbouring servers while performing visited network registration is called shadow registration [87]. However, this methodology is expensive since messages are wasted. The issues related to SIP and MIP interactions (both architectural and security considerations) with a focus on 3GPP2 have been outlined by Faccin *et al* (2004) in [81]. But their work does not provide any analytical model to investigate the topics covered.

3.1.8 Constraints of Instant Messaging

The work over instant messaging [210, 211, 212] observed so far lacks a thorough analysis of the scalable behaviour of the nodes involved in providing the IM service. Unlike the RPID message that carries presence information, the messages of IM may be very large. Large instant messages have two important disadvantages: service behaviour is too slow on low bandwidth links and more importantly, messages get fragmented over some transport protocols and then look at SIP extension that resolves this issue. Even if messages are compressed, sometimes SIP messages can be loo large. If one of the fragments of this message gets lost, the sender needs to retransmit the whole message, which is clearly a quite inefficient way to perform packetloss recovery.

Moreover, some port-based firewalls and NATs (Network Address Translators) cannot handle fragments. This is because only the first fragment carries the port numbers of the datagram carrying the message. When a firewall or a NAT receives a fragment which is not the first one, they cannot find the port number of the datagram and simply discard the packet. So, in some situations e.g., an IMS terminal behind a NAT that cannot handle fragments, it might be impossible to transmit large SIP messages.

Another problem with SIP is that the fact that any proxy can change the transport protocol from TCP to UDP, SCTP, or other transport protocols and vice versa. The protocols other than TCP and SCTP are not famous for congestion control. If an IMS terminal is sending a large instant message over a transport protocol that does not offer congestion control, the network proxies can become congested and stop processing other SIP requests like INVITE, SUBSCRIBE, etc. Even if a terminal sends large SIP MESSAGE over a transport protocol that implements end-to-end congestion control e.g., TCP, SCTP, the next proxy can switch to UDP and congestion may occur.

3.1.9 Solutions from the literature on IM

To solve the issue of large message passing and congestion control in IM, a limit has been placed on the SIP MESSAGE method such that MESSAGE requests cannot exceed the MTU (Maximum Transmit Unit) minus 200 bytes. If the MTU is not known, this limit is 1300 bytes. Another solution to sending SIP MESSAGE requests with large bodies is to use the content indirection mechanism [209]. Content indirection allows replacing a MIME body part with an external reference, which is typically an HTTP URI. The destination IMS terminal fetches the contents of that MIME body part using the references contained in the SIP message. Content indirection is especially useful for optimal body parts. For instance, if Alice uses content indirection to indicate her photo in her INVITEs, the callees can choose whether or not they want to fetch it. A callee on a low bandwidth link can probably live without seeing Alice's photo, while another callee using a high-speed access will most likely enjoy seeing the photo.

Another solution to getting around the size limit problem with MESSAGE is to use session-based IM mode rather than pager mode. Session-based instant message mode uses the SIP INVITE method to establish a session. An IMS terminal establishes a session to send and receive instant messages via Message Session Relay Protocol (MSRP) [91]. MSRP is a simple text-based protocol whose main characteristic is that it runs over transport protocols that offer congestion control.

There are currently three methods defined in MSRP after the INVITE message is sent for an IM session set up:

SEND: sends an instant message of any arbitrary length from one endpoint to another.

VISIT: an endpoint connects to another end point.

REPORT: endpoint or a relay provides message delivery notifications.

MSRP does not impose any restriction on the size of an instant message. If an IMS user, Alice wants to deliver a very large message, she can split the message into chunks and deliver each chunk in a separate SEND request. The message ID corresponds to the whole message, so the receiver can also use it to reassemble the message and tell which chunks belong with which message.

Long chunks may be interrupted in mid-transmission to ensure fairness across shared transport connections. This chunking mechanism allows a sender to interrupt a chunk part of the way through sending it. The ability to interrupt messages allows multiple sessions to share a TCP connection, and for large messages to be sent efficiently while not blocking other messages that share the same connection, or even the same MSRP session. Any chunk that is larger than 2048 octets MUST be interruptible [91].

3.1.9.1 MSRP Relays

One of the characteristic of MSRP is that, MSRP messages no not traverse SIP proxies. This is an advantage, since SIP proxies are not bothered with proxying large instant messages. Also, MSRP does not run over UDP or any other transport protocol that does not offer end-to-end congestion control. It supports instant messages to traverse zero, one or two MSRP relays. The relay extension of MSRP is defined in [213]. A typical MSRP session is shown in Figure 3-9.

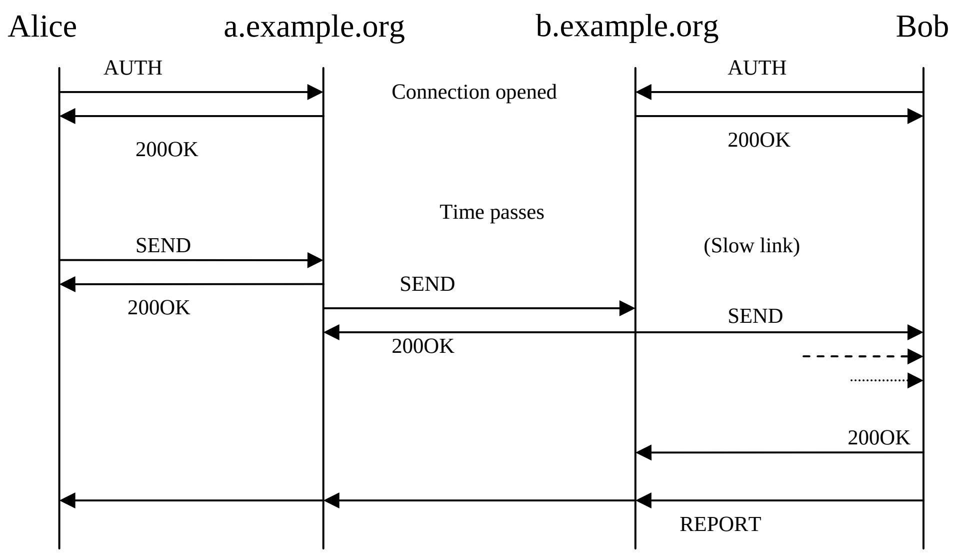


Figure 3-9: Typical MSRP session with relays [213]

The default is that SEND messages are acknowledged hop-by-hop. Each relay that receives a SEND request acknowledges receipt of the request before forwarding the content to the next relay or the final target. When sending large content, the client may split up a message into smaller pieces; each SEND request might contain only a portion of the complete message. For example, when Alice sends Bob a 4GB file called "file.mpeg", she sends several SEND requests each with a portion of the complete message. Relays can repack message fragments en-route. As individual parts of the complete message arrive at the final destination client, the receiving client can optionally send REPORT requests indicating delivery status. MSRP nodes can send individual portions of a complete message in multiple SEND requests. As relays receive chunks they can reassemble or re-fragment them as long as they resend the resulting chunks in order.

A series of papers [214-217] have studied the capacity scaling in relay networks. These works quantify the impact of large wireless relay networks in terms of signal-to-noise ratio. Most of the work focuses on characterizing one relay node only. The work of H. Bolcskei *et all* (2006) in [217] demonstrated that significant performance gain can be obtained in wireless relay networks employing terminals with multiple-input multiple-output (MIMO) capability. However, these works do not address the issue of characterizing traffic parameter in relay nodes. A significant challenge is to analyse system as the IM chunks go through a maximum of two relay nodes with the blocking probability and stability conditions.

3.2 Discussion of Problems based on Lit Review

There are several aspects in the IMS that require much attention and modifications. Some of the existing technologies are still underdeveloped. A few problem areas have been identified as part of this research work. The literature review discussed thus far strengthens the niche for these statements of problems stated below.

Aspect 1: Presence service is the foundation service among other services in the 3G IMS. Scalability is always an issue for massive number of watchers / presentities joining an IMS cell. The message processing load will be heavy for presentity movement. Every time presentities change state, messages will be generated to the PS and consequently the corresponding watchers will be updated by the PS. This will have direct impact on the performance of a PS. A presence application in an IMS mobile terminal device contains a list of 100 presentities. A watcher receives NotifyPresUp message every time any of its presentity changes state. Although the SIP (Session Initiation Protocol) event notification framework (RFC 3265, [148]) offers powerful tool, in some situations the amount of information that the Presence Server has to process might be large. When presence information reaches a small device that has constraints in memory, processing capabilities, battery lifetime and available bandwidth, the device may be overwhelmed by the large amount of information and might not be able to acquire or process in real time. So, there has to be tradeoffs between the amounts of information sent, the frequency of the notifications, and the bandwidth usage to send that information.

Methods to mitigate the message processing load from the PS with the balance in the real time view of the watcher notification are essential. In order to achieve this, efficient scheduling and reduction in bandwidth consumption in the admission control of a PS is required. There exist some mobile node presence optimization techniques like Partial Notification mechanism, Event Filtering etc (which are discussed in section 3.1.3). However, these are still under design phase of the IETF and a stable solution is under developed and required. Deriving efficient admission control mechanisms for a Presence server is under the scope of this research.

The IMS terminal subscription/registration time is another key issue while a mobile node registers with its home network Presence Agent / Presence Server as a watcher. The existing procedure in IMS allows a mobile node to publish its presence as a watcher to its Presence Agent (PA) either for a constant amount of time or for a period of time mentioned by watcher during the time of publishing. If the registration/publishing time is set too short compared to the mobility of the IMS mobile node, the mobile node will have to re-publish its presence soon with the same PS. The frequency of sending messages for such implicit registration/subscription will be increased. Thus the re-registration with the same information will introduce extra messages and redundant data in the cache. On the other hand, if the registration time is set too large and the mobile node does not re-register with the home network until the time out occurs, the actual position of the node becomes unavailable for the home network. This will lead packets to deliver from MN to CN inefficiently. As mentioned earlier in section 2.4, de-registration is accomplished by a registration with an expiration time of zero seconds. Again, excessive de-registration may introduce overheads in number of messages. Another problem of having the publishing time of a watcher node too long in IMS is that the system will have to periodically notify the watcher the information of the presentities (it is watching). Any location update at the presentity side will result in notification to the watcher by the system.

So, the constant time set may create bottleneck because of excessive message flow in the network. In other words, for a long timed extension, a PS will have to generate excessive NotifyPresUp messages to keep the watcher updated, where as for a short one, the watcher will have to subscribe frequently with the PS and accordingly increasing message flows in both cases. Thus an optimal procedure to set the timer of the registration/subscription life time for the mobile node with its home network in IMS is desirable.

Aspect 2: A crucial factor is dimensioning the PoC service in IMS. The related works on PoC service are completely ignorant about dimensioning PoC controller to optimize revenue for service providers. They also fail to identify the PoC session behaviour and optimal resource/hardware utilization. The performance analysis cannot be done without understanding the end-users requirements. Usage behaviour will also affect the performance requirements. It should be noted that the performance for PoC is highly dependent on tuning the service from an end-to-end perspective. Deployment requires expertise in the entire service delivery chain including service networks, core networks, radio networks, terminals and the service itself.

The capacity dimensioning of the radio access network is directly linked to the capital expenditure for the mobile network operators. For a new service offered to the end users, the service providers must offer sufficient capacity in the form of radio resources. Usually, radio resources imply time slots and frequency. These resources can be added in two ways (a) Capacity expansion of existing base stations or (b) Deployment of new base stations [138]. For the former case, cost as function of capacity follows a rather smooth line that is proportional to the amount of offered capacity. In the later case, a steep increase in cost is needed to offer the extra capacity. Part of this thesis (chapter 5) focuses on the optimal utilization of base station resources for a PoC service in IMS.

We discussed the evolution of Push-to-talk over Cellular (PoC) in IMS earlier. There are plenty of areas yet to be improved in the PoC service. A mechanism is required to assess the access of the early media session and on demand sessions for better network performance. Access should be restricted to the early sessions during heavy traffic. Route optimization via the PoC controlling function is a common issue that needs to be addressed. Also, there must be an upper limit for the number of simultaneous session set up for each PoC client and session length for each PoC session. This research addresses the aforementioned PoC issues by providing precise derivation and analysis based on available network resources and infrastructure.

Aspect 3: Another significant factor in IMS is the SIP session set up scenario. An immature choice may introduce significant delay. A mobile node has to send BU (Binding Update) to the corresponding node while session is being set up. The system needs to benefit from the session establishment in mobile environment. The key point is when to send Binding Update (BU) message and when to start the data transfer so as to benefit from the optimized route in IMS. Initiation of RE-INVITE message in SIP is not much of a desirable aspect when a mobile terminal changes its location in a session establishment scenario. This leads the niche to analyse the possible IMS session set up options in mobile environment. The existing session establishment scenario of IP Multimedia Subsystem (IMS) suffers from triangular routing for a certain period of time when an end SIP user or terminal is mobile. There may be other options to make the session set up more efficient while optimizing the cost at the same time.

It can be observed from the literature review that though large amount of work is available on the mobility management i.e., post session set up over SIP and IPv6 interaction, little is known on the pre-session set up issue in mobile environment over IMS. The possible scenarios need to be investigated in detail while a mobile node tries to initiate a session. In this thesis, we identify the optimal option in order to achieve better system performance and reduce latency in such situation.

Aspect 4: In any IMS network the capacity is large for Instant Messaging (IM) communication service. Large messages have to be broken down into chunks to overcome the fixed size limit fact. Real time service of IM is always desirable. However, issues arise if the relay nodes in between source and destination IMS terminals possess slow links with finite buffer. Therefore analysing service discipline of the chunks of IM is necessary. In an IM system with relay nodes, the buffer capacity and the service rate of the relay nodes may vary. Analysis of such system is not trivial. In this thesis, we explore queuing analysis with a special case for instant messages when the messages traverse via two relay nodes. Such analysis of IMS instant messages indeed requires much attention when the capacity and the service time of the relay nodes vary. Although, the study of the fundamental frameworks, namely Integrated and Differentiated services have a long history, defining queuing characteristic with blocking and stability of an IMS instant message traversing the relay nodes under MSRP is essential.

3.3 Objective & Methodology

In the context of the four issues discussed in the last section, this research investigates for the efficient admission control methods in the IMS presence service, optimal values to dimension an IMS PoC service and for the possible options to set up a SIP session in IMS along with the existing one. A queuing analysis for IM service with a maximum of two relay nodes is also provided.

The challenges identified for the IMS presence service in chapter 4 are (a) how to schedule the incoming messages from presentities to a PS efficiently, (b) which messages are to drop to reduce load from a PS, (c) how to derive an optimized message dropping time for a PS, (d) how to compute effective bandwidth consumption due to class based message generation from a PS, (e) how to analyse the performance of a PS in terms of message blocking probability, (f) how to derive the optimized watcher subscription time and (g) how to measure the cost of a Presence system. The challenges addressed for the IMS PoC service in chapter 5 are (a) how to control a PoC session access based on network resources, (b) how to derive expressions to optimize the route for a PoC controller (c) what is the maximum lifetime of a PoC session based on network resources and (d) what is the optimal number of allowable simultaneous sessions for a PoC client during rush hour. The issues addressed in chapter 6 are (a) how to compare the possible session set up options in terms of cost and delay and (b) when exactly the MN (Mobile node) sends the BU (Binding Update) to the CN (Correspondent node) in SIP session set up over IMS. In chapter 7, our objective is to (a) provide a discussion of the states of an IM system when SEND chunks go through two consecutive relay nodes and (b) to define the blocking probability and stability for varied service rates of the relays.

In a nutshell, the goal of this research is to reduce delay, message-overhead and above all improve performance while making some of the existing mechanisms in IMS efficient. The investigation areas are the three services of IMS: (1) Presence Service, (2) Push-to-Talk over Cellular, and (3) Instant Messaging, and (4) Over the issue of IMS session set up. This research basically relies on design, mathematical modelling and successive simulation work. Innovative applications of stochastic process and traffic theory ([113], [114], [115], [116], [117], [118], [119], [120], [121], [122], [123], [124], [125], [126], [127], [128], [129], [130], [131], [132], [133], [162], [221]) have been used to derive the models. Some of the proposed algorithms use heuristic method to gather values for parameters. We compute the time complexity for all the derived methods. We rely on Java simulation for topology independent implementation. The simulation tools used to simulate network prototype and topology in chapter 4 and 6 are OPNET 11.5 modeller with utilities: (a) Wireless module (b) Simulation runtime (c) IPv6 (d) Flow Analysis. The supported complier that is used for OPNET 11.5 is VC++.NET of VS.NET 2003 professional edition. The server behaviour is considered to be M/M/1 for implementation purpose.

In chapter 4, we first propose a weighted class based queuing (WCBQ) mechanism to drop the low priority pre-existing messages from the PS based on the optimal sojourn time. We also discuss the event-throttling presence optimization technique that is proposed by the IETF engineers and compare them with our WCBQ in terms of admission control for a PS. The goal is to save message generation as much as possible for the PS and at the same time keep the real time view for the watchers. In chapter 5, a few of the PoC dimensioning issues are resolved by modelling and the numerical results are generated out of the model equations. The possible IMS session establishment options are compared using the simulator in chapter 6. The cut off threshold to identify the best option is also simulation driven in this chapter. Finally, the IM relays in terms of varied capacity and service rates are modelled using the applications of queuing theories in chapter 7.

Chapter 4 Admission Control for Presence Server

4.1 Introduction

It is indeed important that the user gets accurate and rich presence information while the bandwidth usage is reduced in the IMS presence service. Recapping from the literature review, the flow of messages will be massive for large amount of publishers and watchers joining an IMS system. Every time a presentity changes state, the Presence Server (PS) has to notify all its associated watchers by generating NotifyPresUp messages (see Figure 4-1). Clearly each of the IETF works (discussed in section 3.1.3) has limitations and tradeoffs.

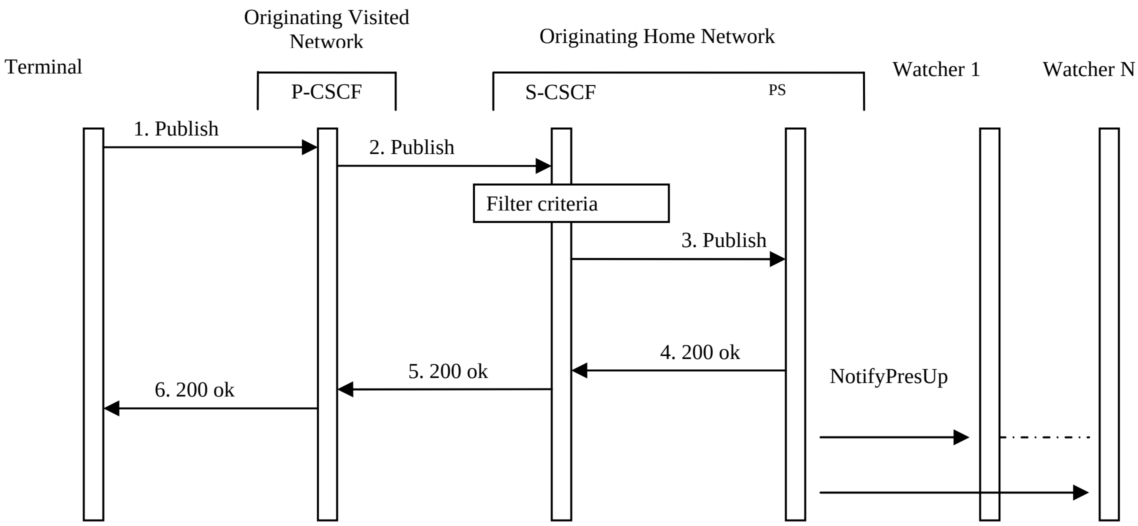


Figure 4-1: PS notifying watchers of a presentity's state change

The admission control technique with effective bandwidth allocation is a mature topic today [144, 145, 146, 152, 153]. Janevski and Spasenovski (2000) in [147] proposed a wireless class based flexible fair queuing algorithm that supports QoS demands of different traffic classes both at error free and error states for wireless IP networks. Marbach (2004) analysed the optimized pricing scheme with packet loss in a game theoretic priority servicing framework in [149]. Moorman and Lockwood (1999) developed and analysed a wireless scheduling algorithm in [150] to provide QoS bounds to the ATM traffic classes of [151]. However, an efficient scheduling of presentities publishing information to the PS has not been defined in the IMS environment yet.

In this chapter, we developed admission control mechanisms for a PS centred through a proposed weighted class based scheduling mechanism to reduce the load of the IMS Presence Server (PS) during heavy traffic. The contributions of our research work in this chapter are:

i. Introduce a Weighted Class-Based Queuing (WCBQ) system to reduce load from the PS;
ii. Compute a threshold timer based on which lower priority messages to be dropped from the PS when necessary;
iii. Derive mechanism for controlling service and performance to avoid starvation at the PS; and
iv. Develop a theoretical model to optimize the watcher subscription time.

The literature review on class-based queuing is hand-full [154-155]. Our Weighted Class Based Queuing (WCBQ) distinguishes classes according to message arrival rates and weighs flows inside them according to the number of watchers who are watching a presentity. An optimal sojourn time for the heavily weighted messages has been proposed. Pre-existing messages are dropped from low priority classes based on the timestamp of newly arrived messages and the derived optimal timeframe in order to achieve efficient system performance. We then compare our WCBQ mechanism with the First Come First Served (FCFS) queuing including different throttlers. We also derive expressions to quantify effective bandwidth for our WCBQ. Further we analyse the cost of a PS in terms of message generation and propose a watcher subscription time based on the analysed cost function. For the theoretical derivation of the admission control expressions we consider the PS as an M/G/1 system where as for the sake of implementation and performance analysis we consider the PS as the M/M/1 system, a special case of M/G/1. Justification has been provided as to the application of the M/M/1 system with the classified service time of RPID messages at the PS.

4.2 Overview of Class Based Queuing

Class based queuing (CBQ) is well defined in [154] by Floyd and Jacobson (1995). CBQ architecture (Figure 4-2) is based on a generic "fair" scheduler controlled by a generic link-sharing scheduler. Incoming traffic is inserted (classifier) into the appropriate queue according to a set of filtering rules. General scheduler (usually Weighted Round Robin (WRR)) extracts packets from queues and it guarantees each class to receive at least its nominal bandwidth.

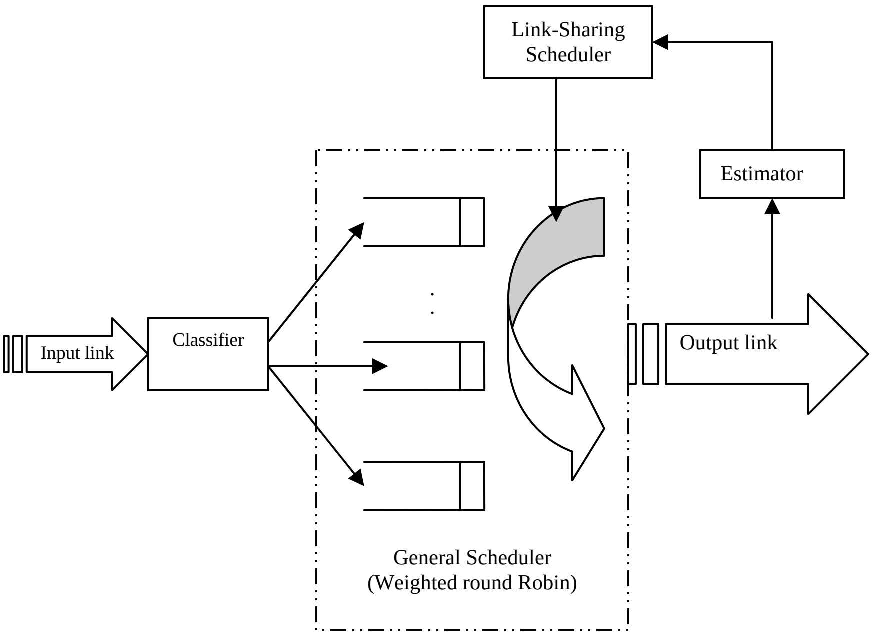


Figure 4-2: CBQ building blocks

The estimator measures the inter-packet departure time for each class and checks whether the class is exceeding its allocated rate *(overlimit* class). The link-sharing scheduler cooperates with this “feedback block” and distributes the excess bandwidth according to the link-sharing structure. Basically, the link-sharing scheduler keeps control and suspends (for a specific amount of time) classes that exceed their allocated rate. Suspension time is calculated in such a way to force the class being consistent with its allocated bandwidth. Generally speaking, it appears like the suspended class is no longer active so that the WRR does not give any service to it until the suspension ends. Link-sharing scheduler reconciles delay with link-sharing capabilities by allowing a "Priority Queuing"-like service without starvation for lower priority classes. F. Risso (2001, [155]) presented an enhanced version of CBQ, called Decoupled-CBQ (D-CBQ), whose main points are the improvement of the rules used to distribute bandwidth according to the link sharing structure and the decoupling of bandwidth and delay. D-CBQ is also able to guarantee tighter delay bounds and more precise bandwidth guarantees.

An implementation of the estimator, presented in [154] and in [201] is the following: Consider a specific leaf-class/queue. Let *s* be the size of the most recently transmitted packet in bytes, *b* the link-sharing bandwidth allocated to the queue in bytes per simulation unit, and *t* the measured inter-departure time between the packet that was just transmitted and the previous packet transmitted from that queue (Figure 4-3) Ideally, we would like inter-departure time to be *t=s/b***.** Let *diff=t-s/b* be the discrepancy between the actual inter-departure time and the allocated inter-departure time for that class for packets of that size. So *diff* is negative when the class transmits more often than its allocated bandwidth permit, and positive if it transmits less often than allowed. The estimator computes the exponential weighted moving *avg* of the *diff* variable using the equation *avg = (1-w)avg +w*diff* as shown in [154] (*w* determines the time constant of the estimator). A class is considered to be overlimit if *avg* is negative and underlimit if *avg* is positive. The value of *avg* computed by the estimator is also used to update the *time-to-send* field associated with each queue. This field indicates the next time that the server is allowed to send a packet from that queue. For a queue with positive *avg***,** the estimator sets the *time-to-send* field to zero, indicating that the class is under its limit. For a regulated class with negative *avg,* the link sharing scheduler sets the *time-to-send* field to *s/b* seconds ahead of the current time. This is the earliest time the queue will next be able to send a packet. Thus, a regulated queue is never restricted to less than its allocated bandwidth, regardless of the "excess" bandwidth used by that class in the past.

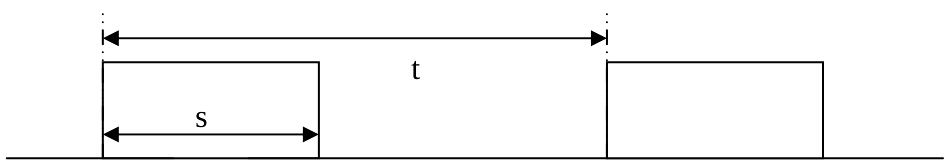


Figure 4-3: CBQ to estimate the throughput uses the rate of the bytes sent to calculate the inter-departure time

4.2.1 Dynamic Class-Based Queue Management (DCQM) [197]

In this section, we describe Dynamic Class-Based Queue Management (DCQM) model from [197]. In the proposed DCQM abstraction, input streams are aggregated into a fixed number of classes. Each class has an associated queue and consists of streams with similar loss tolerances that are aggregated together under a single class state. A scheme to allow flexibility between the two extremes of per-stream state information and per-class state information is required. In order to accomplish this, the notion of a group is defined. A *group* is defined to be a set of streams, *Str*, with loss characteristics under the constraint of $|L| < Grp,$ where $|L|$ is the cardinality of set *Str* and *Grp* is the maximum group size. State information is maintained on a per-group basis. Thus, by varying *Grp*, it is possible to achieve per-stream QoS (*Grp = 1*) or per-class QoS (*Grp* = ∞).

The goal of DCQM is to provide a scheme that adapts appropriately to system dynamics to balance scalability with QoS granularity. For a given media server, let *Grp* be the maximum group size, *Str* be the number of streams being scheduled by the server, *K* be the number of classes, and *Ftr* be a multiplicative factor that represents the additional amount of state information beyond *K* that can be maintained. Therefore, the general constraint on the system as defined in [197] is as follows:

$$\frac{Str}{Grp} \leq K * Ftr \tag{4-1}$$

Rearranging the equation yield the maximum group size *Grp* that is available as:

$$Grp \geq \frac{Str}{K * Ftr} \tag{4-2}$$

where, *K* is a fixed quantity and *Ftr* and *Str* vary upon server load and stream creation or termination, respectively.

4.2.2 Adaptive Group Size

An adaptive group sizing algorithm can be invoked periodically in DCQM. In an increasing only model, the algorithm would increase *Grp* until steady state occurs. For a server that can experience a dynamic load, it may be more desirable to allow the algorithm to decrease *Grp* (better QoS) during low loads and increase *Grp* (better scalability) during high loads. However, when selecting the frequency of group rearrangement, one must also consider the cost of group rearrangement which depends on the change in G as well as the relative distribution of the groups.

In a DCQM media server, two sets of information are maintained. First, there is a limited amount of per-state information. This per-state information is the appropriate routing (queue pointer) information detailing which group a stream belongs to. This information is fairly static throughout the duration of a stream and thus presents scalability concerns only in terms of storage capacity, not CPU bandwidth. Second, the prioritization information is maintained on a per-group basis. The amount of prioritization information can range from per-stream (variable) to per-class (fixed). Thus, an increase in scalability occurs due to a reduction in both required storage capacity and required CPU bandwidth. The decrease in required storage capacity occurs because state information is maintained on only a per-group basis rather than on a per-stream basis. The decrease in required CPU bandwidth arises from the decrease in the number of queues being checked for deadline expiration and being considered for scheduling. For a non-adaptive server, the server administrator must determine the appropriate trade-off between scalability and fine grain QoS. However, an adaptive server will appropriately adjust this trade-off at a cost of scalability (CPU bandwidth involved in invoking the adaptive algorithm). Other Class-Based QoS applications can be found in [198-200].

4.3 Proposed Queuing System

The goal of any Call Admission Control (CAC) function is to simultaneously achieve the twin objectives: maximizing the resource utilization and guaranteeing the promised for all accepted session. When a session has been established, the network has to ensure the traffic of session must be followed the traffic contract specification (i.e., source traffic descriptors and conformance definitions). Especially, we see the traffic flows in scene of heterogeneous traffic belonging to different classes of service. These classes of service include Constant Bit Rate (CBR), real-time Variable Bit Rate (rt-VBR), non-real-time Variable Bit Rate (nrt-VBR), Available Bit Rate (ABR), and Unspecified Bit Rate (UBR). Service classes may be characterized by different traffic characteristics and they offer for service randomly. Therefore, it is difficult to specify traffic behaviours in order to transfer these mix-services in network. In our model, we deal with the RPID messages as input for the PS. We consider that the RPIDs carry presence information only and that the size of a RPID message is below the maximum size of an IP packet. We also assume the messages are elastic due to the high arrival rate of the messages. This way, a flow of messages can be represented as a packet stream. Since the messages are serviced up to the assigned priority in a CBQ, channel needs to be assigned in order to have fair attention for all message-flows from the PS to be serviced. Hence, we allocate bandwidth to the groups and derive effective bandwidth (which is discussed in sub-section 4.4.4). If a certain class clears its queue, the available bandwidth is to be reallocated to other classes.

Our objective is to propose an efficient scheduler for the PS in heavy traffic scenario. The queuing model is shown in Figure 4-4 and the corresponding flow chart is provided in Figure 4-5. The scheme is created to support multiple traffic classes. The PS assigns the publishing presence information a hierarchy of priorities. Our tendency in creating this queuing algorithm was to take into consideration the high publishing rate and the number of watchers associated with a particular publisher (presentity).

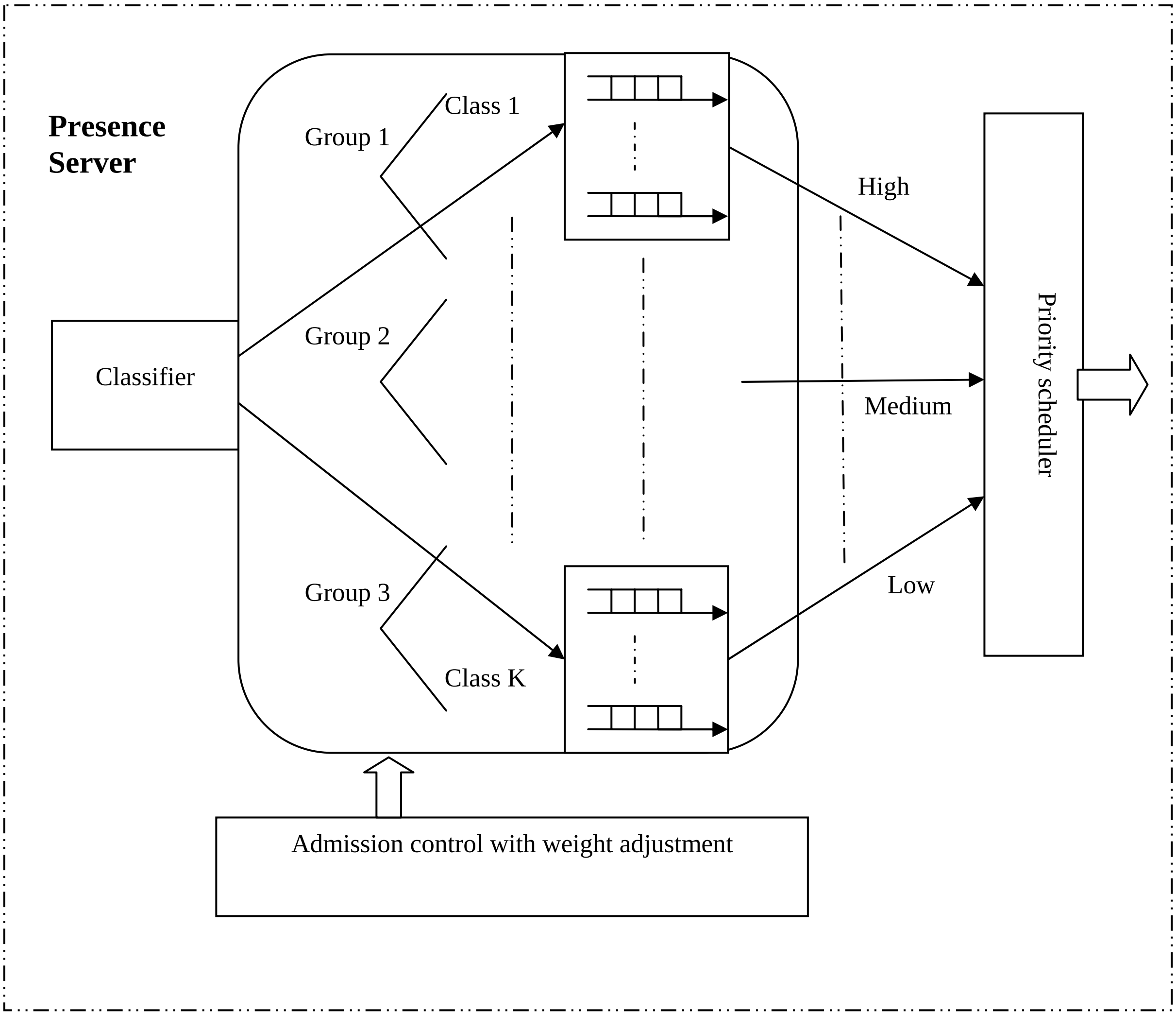


Figure 4-4: WCBQ model

The classifier differentiates traffic into classes based on arrival rate of messages from the presentities. A class selector separates arriving messages intro different queues for every class according to weight of each arriving message. We define the weight of a presentity message as the number of watchers watching that particular presentity. Lower arrival rate gets higher priority and the higher priority classes are placed at the top in sequence (see Figure 4-4) i.e., class 1 has lower arrival rate than that of class 2, class 2 has lower arrival rate than that of class 3 and so on.

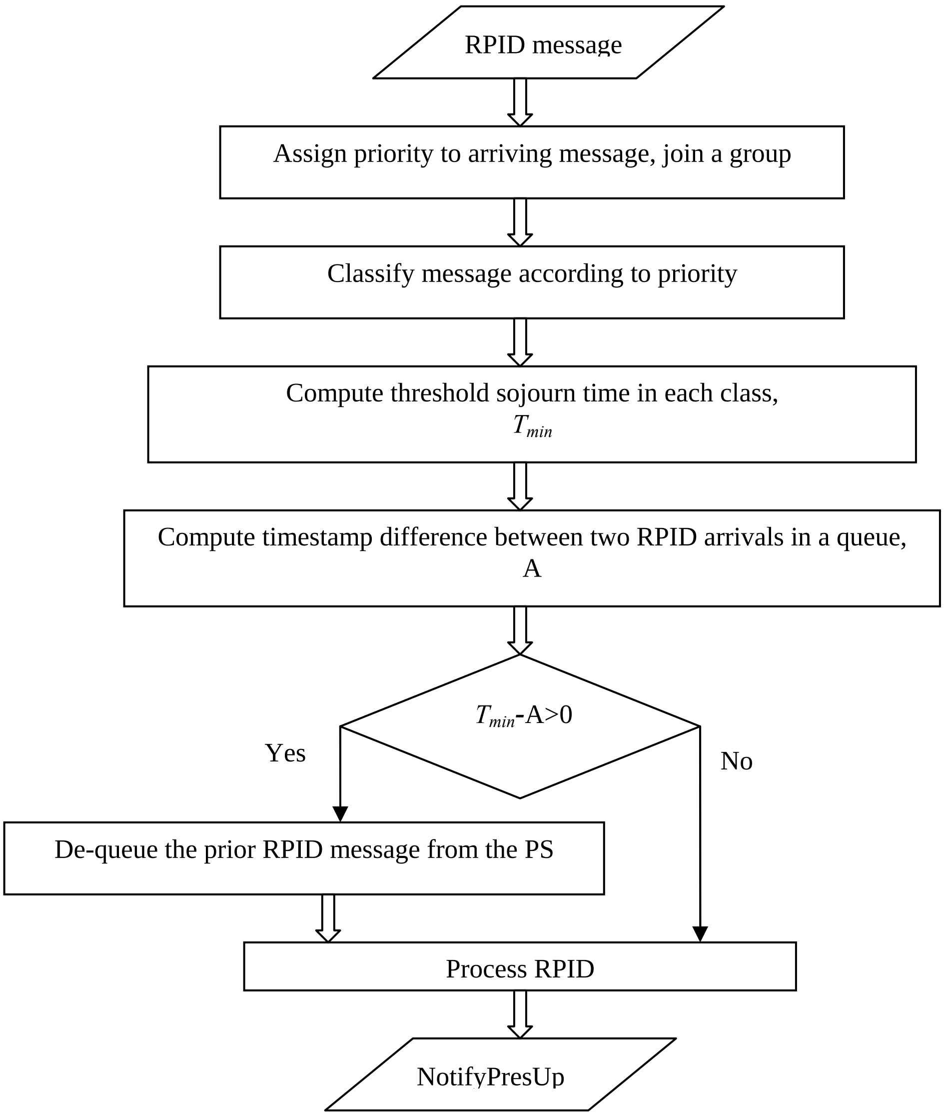


Figure 4-5: Flow chart for WCBQ

Having all flows of each class the same arrival rate, are further classified according to their weight. The lower the weight (number of watchers watching a presentity), the higher the priority for that message. Thus, the topmost flow in a class has the highest priority while the bottommost flow has the lowest priority. A flow may have messages from multiple presentities that have same arrival rate and same weight. The messages will be inserted in that case in the First Come First Served (FCFS) manner in the flow. A class will be able to use the empty space of other classes in the buffer in case its own buffer is full. With these assumptions, the flows belonging to class 1 will be first served until the buffer for this class is emptied and so forth.

However, to avoid monopolization of the bandwidth by the higher priority flows, we should limit the maximal capacity that can be allocated to them. This can be accomplished by an admission control mechanism. Since the higher priority classes have lower arrival rate, they may be grouped together (group 1) to be serviced in heavy traffic situation. Similarly, medium priority classes may be grouped into group 2 and lower priorities into group 3. The range of class-grouping will depend on the network scenarios and the frequency of state changes by the IMS presentities. The adaptation of group size may be followed from the work specified in [197]. In this work, we are particularly interested in the lower priority classes for which huge number of NotifyPresUp messages need to be generated by the PS destined to the end IMS terminals. The PS will keep a time stamp for every RPID message arriving. The PS will also generate an optimized/threshold stay time for each class which will be discussed later. The threshold time means minimum stay time of messages at the PS. This threshold time will be used to compare the time stamp difference between two arriving message of the same flow in the queue to drop a pre-existing message from the PS to reduce load and to save number of message generation. If the time stamp difference is greater than the threshold time, then the RPID is not dropped and a NotifyPresUp is generated for that RPID.

4.4 Admission Control Mechanisms

Let us denote B as bandwidth of the outgoing wireless link. The weights assigned to flows in a class j are $w_{ji,}$ $i=1,2,\ldots,N,$ where N is the number of flows in the class. The relative throughput of each flow normalized on the link bandwidth for the class j is:

$$RT_{ji} = \frac{w_{ji}}{\sum_{i=1}^{N} w_{ji}} \tag{4-3}$$

When the wireless path is error-free, a flow from class j should get bandwidth share b_{ji}:

$$b_{ji} = RT_{ji} * B(\hat{A}_j) = \frac{w_{ji}}{\sum_{i=1}^{N} w_{ji}} B(\hat{A}_j) \qquad (4\text{-}4)$$

where, $\hat{A}_j$ is the amount (in percent for instance) of bandwidth allocated to class j.

The admission controller may use the followings in heavy traffic situation to compute the maximum number of publishing information accessible in the PS queue. Let z_j be the number of arrivals of a class j, S is the total size of the PS buffer and l is the average size of the RPID document. Thus, the maximum number of messages is defined as:

$$X = \left\lfloor \frac{S - \sum_{j} z_j l p_{ts_j}}{l} \right\rfloor \qquad (4\text{-}5)$$

where, p_{ts_j} is the probability that the PS completes servicing a message of class j in time slot ts with mean service rate μ_j. Then,

$$p_{ts_j} = 1 - e^{-\mu_j (ts)} \qquad (4\text{-}6)$$

Bounds of tail distribution can be used to develop efficient admission control mechanisms. Let, there K classes of publishing information and K independent Poisson processes with arrival rates $\lambda_1, ..., \lambda_K$ in the system. The arrival rates are considered to be equivalent to the steady state probability of presentity movement which is defined later in section 4.4.5 in Eq. (4-45). By the law of superposition of independent Poisson processes: $\lambda = \sum_{j=1}^{K} \eta_j \lambda_j$ where, η_j is the number of different presentities publishing information to the PS with the same rate of λ_j in class j.

The traffic intensity for a class j is defied, $\rho_j = \dfrac{\eta_j \lambda_j}{\mu_j}$.

For the stability condition i.e., $\rho < 1$, we may define the mean waiting time at the PS by applying the Polloczek-Kinchin formula for an M/G/1 system (see [162]):

$$E[W] = \frac{\lambda \overline{\sigma}^2}{2(1-\rho)} \tag{4-7}$$

where,

$$\rho = \sum_{j=1}^{K} N_j \rho_j \tag{4-8}$$

and where, $\overline{\sigma}^2$ is the second-order moment of the service time of an arbitrary message. Let, $\overline{\sigma}_j^2$ be the second-order moment of messages of class $j=1,2,\ldots,K$. By the law of total probability and from Bayes' formula [133]:

$$\overline{\sigma}^2 = \sum_{j=1}^{K} \frac{\eta_j N_j \lambda_j}{\lambda} \overline{\sigma}_j^2 \tag{4-9}$$

Hence,

$$E[W] = \frac{\sum_{j=1}^{K} \eta_j N_j \lambda_j \overline{\sigma}_j^2}{2(1 - \sum_{j=1}^{K} N_j \rho_j)} \tag{4-10}$$

4.4.1 Blocking Probability

The presentities publish their state via the RPID document to the PS (RPID document travels through the presentities P-CSCF and the home networks S-CSCF) and PS acknowledges with a 200 (OK) response. Before the PS places the Publish message from the presentity to an appropriate class of queues, it has to decide whether to send a 200 (OK) message back to the presentity or to discard the message. Upon arrival of a new message, the PS checks for errors and determines whether there is room for this message in the buffer. Depending on outcome of these tests, a 200 (OK) is sent to the transmitting presentity or the packet is discarded at once. In case of the 200 (OK) not being sent from the PS (alternatively the PS might send an error message with code range 500-599), the presentity retransmits the RPID after a suitable interval of time which we call the timeout. We shall assume that the PS cannot contain more than E_K non-acknowledged messages. Under these conditions, the buffer overflow can be modelled for a PS as follows.

Let a PS is composed of

(i) K classes of queues denoted as class $1, 2, \ldots, K$;

(ii) K timeout boxes denoted as stations $1', 2', \ldots, K'$;

(iii) K 200 (OK) boxes denoted as stations $1'', 2'', \ldots, K''$.

With the above conditions, the PS will behave like the combination of several FCFS (First Come First Sever) and IS (Infinite Server) stations. In this situation, the results of BCMP network (Appendix A, [203]) can be applied to achieve the steady state probabilities. The transmission-retransmission process between PS and presentities can be captured by the timeout and 200 (OK) boxes. More precisely, it is assumed that with probability q_j the attempted transmission over class $j = 1, 2, \ldots, K$ fails, either through blocking or through message error. We model this event as having the message enter the timeout box where it resides for a random interval of time. The probability of successful transmission over class j i.e., $1 - q_j$ is modelled as having the packet enter the 200 (OK) box for a random period of time.

Thus, the probability of a message is retransmitted exactly n times over class j before success is:

$$(1 - q_j)(q_j)^n \tag{4-11}$$

Therefore, the mean number of transmission for a message over class j is:

$$\frac{1}{1-q_j} \qquad (4\text{-}12)$$

The total message arrival rate for all classes, λ has been defined before. Let, $p_j, j = 1,2,...,K$, be the probability of a message destined to class *j* in the queue, β be the mean service rate of the arrived messages and δ_j be the mean service rate for the messages in class *j* (processing rate only). The constant service rate β includes the checking whether the buffer is full and a message contains error. We also denote the arrival rate at box $j^{/}$ and $j^{//}$ as $\lambda_j^{/}$ and $\lambda_j^{//}$ respectively; and mean service rate in timeout box and in 200 (OK) box, as $\mu_j^{/}$ and $\mu_j^{//}$ respectively.

Assuming $\underline{n} = (n_0, n_1, n_2, ..., n_K, n_1^{/}, n_2^{/}, ..., n_K^{/}, n_1^{//}, ..., n_K^{//})$ with $|\underline{n}| \le E_K$ where n_0 is the arrived messages in the PS and $n_j, n_j^{/}, n_j^{//}$ are the number of messages in $j, j^{/}, j^{//}$ respectively in state $\underline{n}$, the traffic equations read as follows:

$$\begin{aligned} \lambda_j &= \lambda p_j + \lambda_j^{/} \\ \lambda_j^{/} &= \lambda_j q_j \\ \lambda_j^{//} &= \lambda_j (1-q_j) \end{aligned} \qquad (4\text{-}13)$$

We find

$$\begin{aligned} \lambda_j &= \frac{\lambda p_j}{1-q_j} \\ \lambda_j^{/} &= \lambda p_j \frac{q_j}{1-q_j} \\ \lambda_j^{//} &= \lambda p_j \end{aligned} \qquad (4\text{-}14)$$

Let $\pi_K^{WCBQ}(\underline{n})$ be the stationary probability of being in state $\underline{n} \in S_K$ where

$$S_K = \{(n_0, n_1, n_2, ..., n_K, n_1^{/}, n_2^{/}, ..., n_K^{/}, n_1^{//}, ..., n_K^{//}) : |\underline{n}| \le E_K\}. \qquad (4\text{-}15)$$

Then,

$$\pi_K^{WCBQ}(\underline{n}) = \frac{1}{G_K}\left(\frac{\lambda}{\beta}\right)^{n_0} \prod_{j=1}^{K}\left\{\left(\frac{\lambda_j}{\delta_j}\right)^{n_j} \frac{1}{n_j^{/}!}\left(\frac{\lambda_j^{/}}{\mu_j^{/}}\right)^{n_j^{/}} \frac{1}{n_j^{//}!}\left(\frac{\lambda_j^{//}}{\mu_j^{//}}\right)^{n_j^{//}}\right\} \qquad (4\text{-}16)$$

for all $\underline{n} \in S_K$ and $\pi_K^{WCBQ}(\underline{n}) = 0$ for $\underline{n} \notin S_K$ where G_K is a normalizing constant chosen such that $\sum_{\underline{n}\in S_K} \pi_n^{WCBQ}(\underline{n}) = 1$.

Therefore the total average blocking probability at a PS for the grouped weighted class based queuing is

$$B_{PS}^{WCBQ} = \sum_{\underline{n}\in S_K, |\underline{n}|=E_K} \pi_K^{WCBQ}(\underline{n}). \qquad (4\text{-}17)$$

Under the assumptions of the model, the steady state distribution for the First Come First Served (FCFS) admission control of a PS is given by:

$$\pi_K^{FCFS}(\underline{n}) = \pi(0)\prod_{j=1}^{K} \frac{\rho_j^{n_j}}{n_j!} \qquad (4\text{-}18)$$

where

$$\pi(\underline{0}) = \frac{1}{\sum_{\underline{n}\in F_K} \prod_{j=1}^{K} \frac{\rho_j^{n_j}}{n_j!}} \qquad (4\text{-}19)$$

where $F_K \equiv \{\underline{n} \mid \underline{n} \in S_K \ \ but\ \ n_i \notin S_K for\ some\ i\}$

The service rate here is $\mu_j = \beta\mu_j^{/} + \beta\delta_j\mu_j^{//}$ since the arriving messages are either put into the timeout boxes to discard or processed to generate NotifyPresUp. Therefore, the probability of blocking for FCFS admission control is obtained by:

$$B_{PS}^{FCFS} = \sum_{\underline{n}\in F_K} \pi_K^{FCFS}(\underline{n}). \qquad (4\text{-}20)$$

For the event throttle model, the throttle mechanism applies on the buffer of the PS to reduce the number of messages generated. A watcher may use a throttle mechanism during its subscription to the PS for the minimum time period between two notifications of its presentities. The PS is allowed to use a throttling policy in which the minimum time period of notifications is longer than the one given by the watcher. This throttle mechanism can be applied with both FCFS and WCBQ method. The difference will be in the point of servicing the messages. In such cases, the average blocking probability can be obtained from Eq. (4-17) and Eq. (4-20) with the modification of the mean processing rate, δ_j as the size of generated NotifyPresUp will vary. It is difficult to standardize such processing rate for event throttling mechanisms since the watcher list is not identical for every watcher. From the earlier discussion of event throttling mechanism, it is obvious that extra processing will be required due to state matching and batching processes which will increase blocking.

4.4.2 Efficient Dropping of Buffer Messages

As mentioned earlier that our concern is with the flows that have heavy weights associated in the lower priority classes. The bandwidth allocation should be higher for these types of flows. These types of flows will force the PS to generate huge amount of messages in quick succession. Since, our classifier and weight adjustment scheme will schedule the higher arrival-rate jobs that have heavy weights associated at the end of the buffer; we propose a mechanism that will drop these kinds of pre-existing jobs from the buffer with the arrival of new jobs from the same source/presentity or from the same message flow in order to ameliorate the message generation process for the PS. A minimum time frame is the key within which if a message at the same flow arrives, the pre-existing message from the buffer will be dropped. The mechanism may be applied periodically during heavy traffic in the network. Note that, we use the term job and message interchangeably.

We may use the square root dimensioning method to compute the minimum time frame. Let, T_{ji} be the mean sojourn time encountered by a job at flow i of class j

and C_{ji} the capacity allotted for flow i. If the stability condition $\eta_{ji}\lambda_{ji} < \mu_j C_{ji}$ for $i=1,2,..,N$ holds, then (ignoring the constant η for each class for the simplicity of derivation):

$$T_{ji} = \frac{1}{\mu_j C_{ji} - \lambda_{ji}} \quad \forall i = 1,2,..,N \tag{4-21}$$

Note that, here μ_j is the mean service rate that includes both timeout or 200 (OK) message servicing as well for class j, i.e., $\mu_j = \beta\mu_j^{/} + \beta\delta_j\mu_j^{//}$.

Thus, the average sojourn time of messages at class j is (assuming $\rho_{ji} = \frac{\eta_{ji}\lambda_{ji}}{\mu_j}$):

$$T_j = \sum_{i=1}^{N} \frac{\lambda_{ji}}{\lambda_j} \left(\frac{1}{\mu_j C_{ji} - \lambda_{ji}}\right) = \frac{1}{\lambda_j} \sum_{i=1}^{N} \frac{\rho_{ji}}{C_{ji} - \rho_{ji}} \tag{4-22}$$

The stability condition now reads $\rho_{ji} < C_{ji}$ for $i=1,..,N$. Therefore, our threshold time frame problem reduces to:

Minimize: T_j

With respect to: $\{C_{ji}\}_{i=1}^{N}$

Under the constraint: $D_j = \sum_{i=1}^{N} C_{ji}$

where, D_j is the total capacity of the PS allocated to class j.

Applying the Lagrange multiplier technique, we define:

$$f(C_{j1},...,C_{jN}) = \frac{1}{\lambda_j} \sum_{i=1}^{N} \frac{\rho_{ji}}{C_{ji} - \rho_{ji}} + \beta\left(\sum_{i=1}^{N} C_{ji} - D_j\right). \tag{4-23}$$

Solving the equation $\partial f(C_{j1},...,C_{jN}) / \partial C_{ji} = 0$ for every $i=1,2..,N;$ we obtain:

$$C_{ji} = \rho_{ji} + \sqrt{\frac{\rho_{ji}}{\lambda_j \beta}} \qquad i=1,2,..,N. \tag{4-24}$$

From the constraint $\sum_{i=1}^{N} C_{ji} = D_j$ and Eq. (4-24) we get:

$$\frac{1}{\sqrt{\lambda_j \beta}} = \frac{D_j - \sum_{i=1}^{N} \rho_{ji}}{\sum_{i=1}^{N} \sqrt{\rho_{ji}}} \tag{4-25}$$

Introducing the value of Eq. (4-25) into Eq. (4-24) yields:

$$C_{ji} = \rho_{ji} + \frac{\sqrt{\rho_{ji}}}{\sum_{i=1}^{N} \sqrt{\rho_{ji}}} (D_j - \sum_{i=1}^{N} \rho_{ji}) \tag{4-26}$$

Therefore the minimum timeframe or sojourn time in the class j becomes,

$$T_{\min} = \frac{\sum_{i=1}^{N} \rho_{ji}}{\lambda_j (D_j - \sum_{i=1}^{N} \rho_{ji})} \tag{4-27}$$

It can be observed that, in order to make the threshold time fair for all classes, η_j has to be the average number of presentities present for each class in the above derivation $\left(T_{\min} = \frac{\sum_{i=1}^{N} \rho_{ji}}{\eta_j \lambda_j (D_j - \sum_{i=1}^{N} \rho_{ji})}, \ \textit{including} \ \ \eta_j \right)$ i.e., total number of presentities in a presence system divided by the total number of classes in a PS. Eq. (4-27) may be used in the heavily weighted flows of the low priority classes in a round robin fashion during heavy traffic for a period of time. The idea here is to drop a pre-existing job from the buffer in order to reduce generating huge number of messages for the presentities with the same arrival rate in a very short span of time. The time stamp difference between the two job-arrivals of a same flow or presentity is compared with T_{min} and if that is less than T_{min}, then a prior job from the buffer will be dropped by the classifier. If a class load is greater than or equal to assigned capacity i.e., $(D_j - \sum_{i=1}^{N} \rho_{ji}) \leq 0$ then the threshold becomes negative or infinite. In that case, every two or three etc. alternate pre-existing message from a same presentity or a flow may be dropped. In practice, the accurate notifications for the very rapid state changes of presentities may not be significant for their watchers. The dropping will slow down the end devices which are low in battery capacity/memory receiving NotifyPresUp in very quick succession as well as message generation for the PS.

Alternatively, we may want to keep the dropping rate of the messages to a certain rate. Since the arrivals are Poisson process, the probability of *r* message arrives at a flow in a known period *T* in class *j* is

$$\Pr[R = r \mid T = t] = \frac{(\lambda_j t)^r}{r!} e^{-\lambda_j t} \tag{4-28}$$

Therefore, the probability of *r* arrivals to a flow or from a presentity for a length of time can be determined by the *Laplace* transformation as follows [162]:

$$\begin{aligned} p_r(r) &= \int_{t=0}^{\infty} \{\frac{(\lambda_j t)^r}{r!} e^{-\lambda_j t}\} e^{-st} dt \\ p_r(r) &= \frac{\lambda_j{}^r}{r!} [\ (-1)^r \frac{d^r}{ds^r} (\frac{1}{s + \lambda_j})\ ] \Big|_{s=\lambda_j} \end{aligned} \tag{4-29}$$

From above Eq. (4-28) and Eq. (4-29), a PS can compute the probability of specific number of message arrivals for each of the classes in the PS within time T_{min}. The steps to perform dropping of pre-existing messages by the PS requires *O(1)* time whereas the scheduling may take place in linear time which is practical. This scheduling delay is very low compared to the processing delay of heavily weighted messages; not to mention the bandwidth consumption at the outgoing link.

Many other packet dropping approaches have been presented in networking literature, commonly called Active Queue Management, and there have been a few papers on joint queue management and scheduling. However, our work proposes late dropping instead of early dropping (i.e., removing outdated messages from the queues when newer ones arrive), which has not been studied as extensively before. Our scheduling policy allows to group the low priority flows and then to apply the dropping only to these types of heavily weighted RPID messages.

4.4.3 Performance Analysis

We consider 3 groups of messages in the PS with multiple classes in each group. The priorities in the classes are assigned by the number of watchers associated. The blocking probability by each group of messages is taken as the performance measure. We simulated till the maximum number of unacknowledged messages in a group. We applied the dropping method over all classes of group 3 once. The numerical analysis was performed with Java programming language. The average RPID message length is considered to be fixed for simplicity. Thus the common distribution of service times can be considered exponential for the simplicity in computation (as in M/M/1 model). The parameters used are shown in Table 4-1. The arrival rates, associated weights, transmission failure probabilities and the maximum number of unacknowledged messages in the server of the three groups have been chosen in increasing fashion in the table to simulate the scenario that suits our problem statement for the PS. 22 channels is a realistic capacity in current wireless system. Bandwidth is allocated in quota of a channel which is a fixed transmission speed. Message flows in our problem request different transmission speeds or the unit of bandwidth (for instance, this may be a 9.6 Kbps channel), and the total number of channels available is the system capacity. As mentioned earlier that allocation of bandwidth should be defined for different groups in order to avoid monopolization of prioritization. A flow using x channels can transmit at x times the nominal rate of a single channel. The flow effective bandwidth has been furnished in a later section of this chapter. We used the fixed point technique to compute the blocking probabilities. The state change was captured as the number of messages getting serviced and departing from the PS or being dropped to cause a transition from state a_1 to a_2. We define this state transition probability as:

$$P[a_1, a_2] = \prod_{j=1}^{K} \frac{n_{a_1}!}{n_{a_2}!(n_{a_1} - n_{a_2})!} p_{ts_J}{}^{(n_{a_1} - n_{a_2})} (1 - p_{ts_J})^{n_{a2}} \qquad (4\text{-}30)$$

where, n_{a_1} and n_{a_2} are total number of messages in state a_1 and a_2 respectively and K is the number of classes in a group. p_{ts_j} is computed from Eq. (4-6). The slot duration was kept 1. Therefore, the average blocking probability (denoted as '*Block*') of Eq. (4-17) and Eq. (4-20) for a group is computed as follows:

$$Block = \sum_{\underline{n}} \pi_K(\underline{n}) \times \sum_{\underline{n}} P[a_1, a_2] \qquad (4\text{-}31)$$

where, $\underline{n} \in S_K$ for WCBQ and $\underline{n} \in F_K$ for FCFS.

For FCFS, the messages in each group were served in first come first served manner and were considered not to be classified.

Table 4-1: Parameters for blocking performance with varying load

Parameter	Group 1	Group 2	Group 3
Capacity (channels)	4	6	12
Total arrival rate, λ (arr./per min)	1-20	21-50	51-100
Weight (associated watcher/message, random)	1-24	25-49	50-100
Priority (1-3)	High (1)	Medium (2)	Low (3)
q_j (Probability of attempted transmission fails)	.005	.01	.015
E_K (Maximum no of unacknowledged messages)	10	35	75
β (msg/min)	50	50	50

μ' (msg/min)	5	5	5
μ'' (msg/min)	50	50	50

Since, an IMS terminal can have 100 watchers in its list; the maximum randomly generated associated watcher limit was kept 100 for an arriving message (in group 3). The processing load, (λ/δ) was varied from 4 to 10 for the three groups to compute the respective blocking probability. The processing load depends on processing a RPID document and generating NotifyPresUp messages for a presentity's state change. The blocking probability for the three groups as a function of the processing load is shown in Figure 4-6, Figure 4-7 and in Figure 4-8.

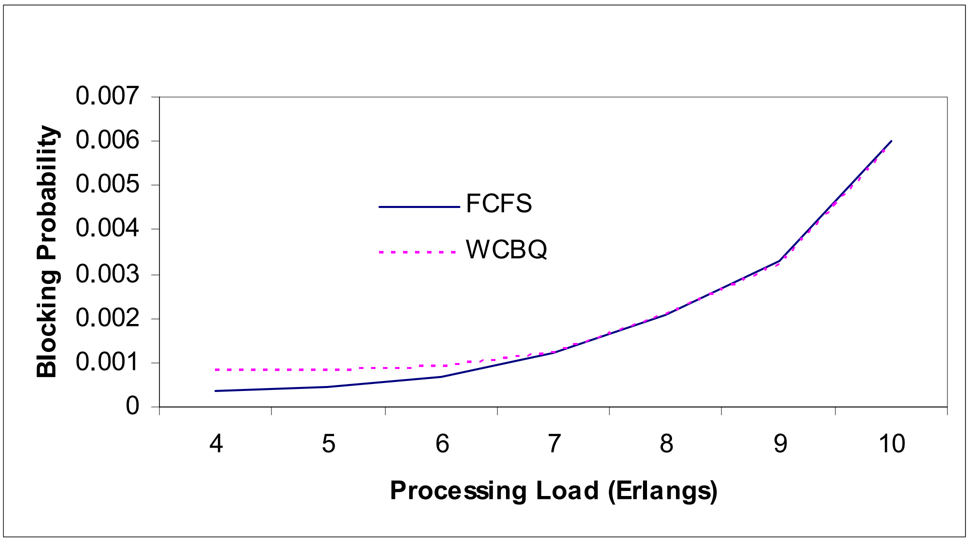


Figure 4-6: Comparison of Group 1 blocking performance for varying offered traffic

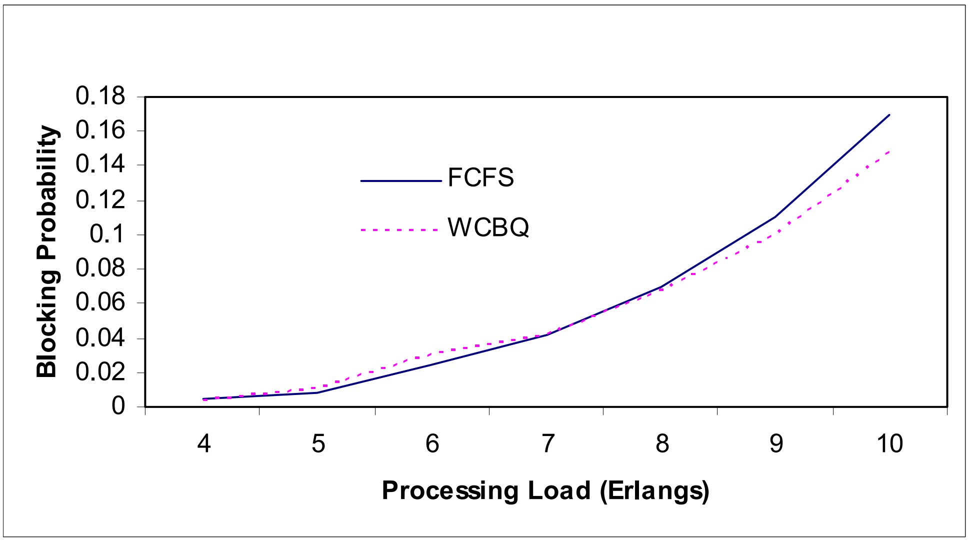


Figure 4-7: Comparison of Group 2 blocking performance for varying offered traffic

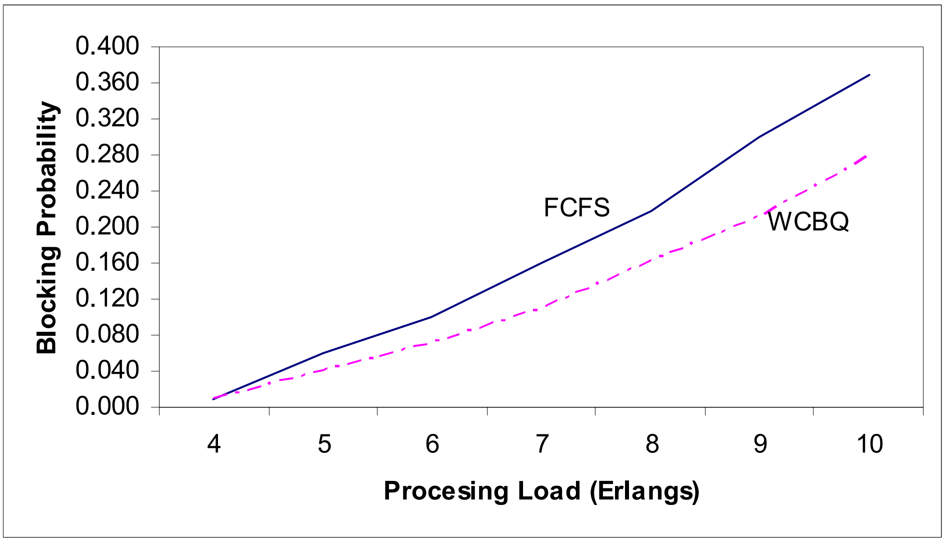


Figure 4-8: Comparison of Group 3 blocking performance for varying offered traffic

The FCFS scheme provides an improvement at the earlier stage in blocking performance for the low priority group where the arrival rates and the associated watchers are low (Figure 4-6). The performance of both the medium priority groups (FCFS and WCBQ) were recorded the same (Figure 4-7) at preliminary stage where as WCBQ supersedes FCFS gradually for higher load. However, performance degradation is experienced by high priority group of messages for FSFC scheme. Our proposed

WCBQ provides intelligent contention resolution for group 3 messages. The main reason for the performance gains can be summarised as follows:

WCBQ scheduling discriminates against the arrival rate and associated weight of the arrived messages in the sense of dropping pre-existing messages based on minimum sojourn time. Since, the system under consideration is non-pre-emptive, when the higher arrival rated messages arrives, they are buffered until capacity is free. The channels are utilized as such low arrival rated messages use them when the higher arrival rated messages are not availing them. The average channel utilization for three groups of WCBQ is depicted in Figure 4-9 for growing load with WCBQ scheduler. We see that the heavily weighted flows i.e., flows of group 3 utilize the PS capacity earliest which is obvious.

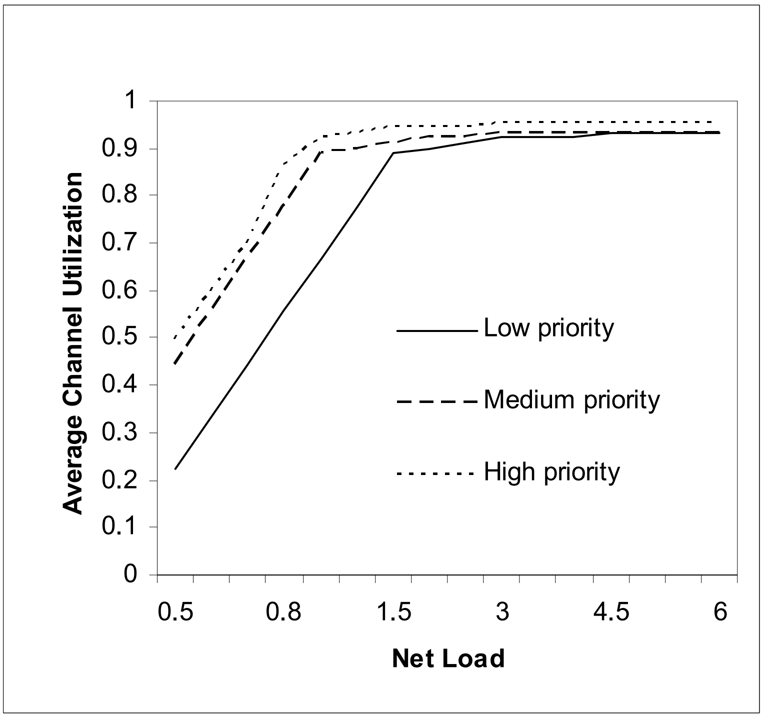


Figure 4-9: Probability of servicing messages for the three traffic groups

Here, it is obvious that the partial state event throttling notification will perform even poorly since, the PS will have to find the state difference from the last full state notification, and to batch the state changes to merge them though this will reduce the number of out going messages.

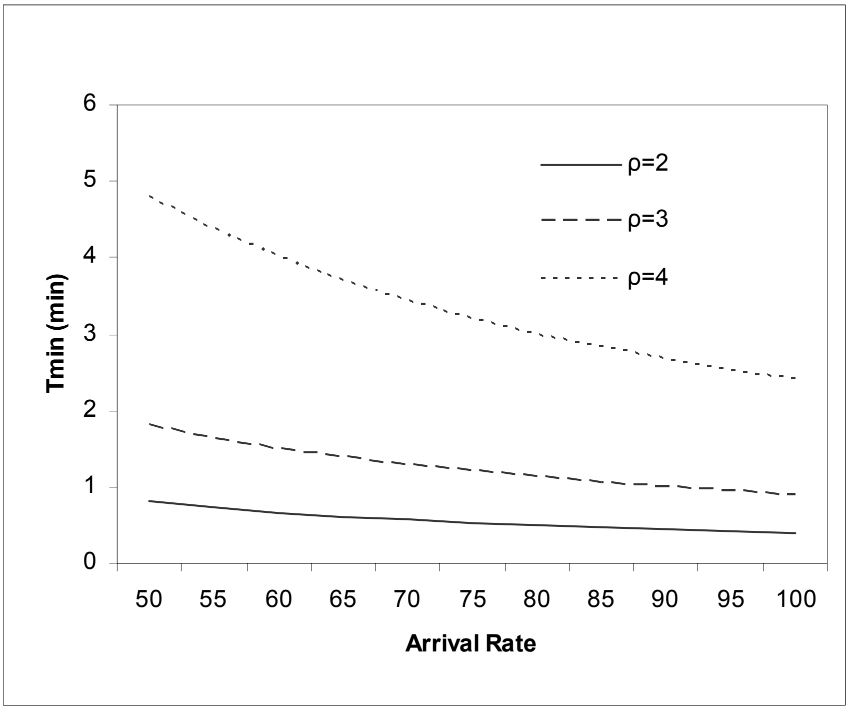


Figure 4-10: Minimum sojourn time for group 3

Next we performed the experiment for the minimum timeframe of messages of group 3 with different net load (see Figure 4-10). The timeframe goes down with growing arrival rates implying the wait time reduces for large number of arrivals. The obvious reason for decreasing timeframe is that the traffic intensity was kept fixed.

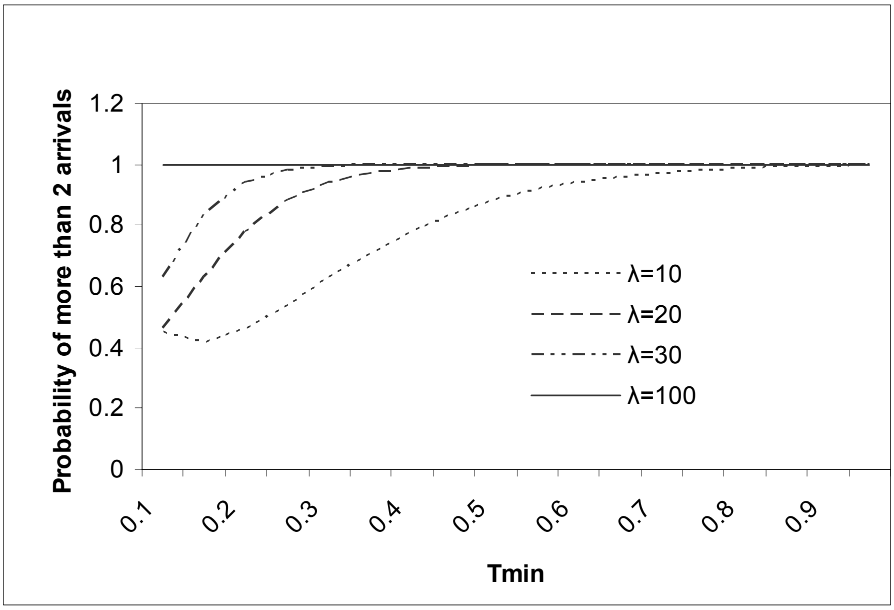


Figure 4-11: Probability of more than 2 arrivals for given Sojourn time

Since, WCBQ will drop a pre-existing message based of another arrival; we find the probability of more than two arrivals with the same time frame. It is obvious (Figure 4-11) that higher arrival rates have almost unit probability in such cases.

Based on our simulations in Figure 4-10 and Figure 4-11, we performed experiment shown in Figure 4-12 to find out the number of message generation saved for the PS by dropping pre-existing heavily weighted messages using our algorithm. Again, the simulation was performed over the group 3 messages for WCBQ, WCBQ-20 and WCBQ-30. We considered WCBQ with different throttle requirement for group 3 messages; the messages were held 20 seconds (WCBQ-20) and 30 seconds (WCBQ-30) for WCBQ before processing them. The pre-existing messages were de-queued once from the buffer-array according to the comparison of arrival time stamps and the threshold minimum time. The number of message generation saved, implies that the number of messages needed to be generated for the number of dropped messages from PS. If a NotifyPresUp message needs to be traversed via routers, then the PS may use the multicast mechanism with which generation of multiple messages are saved by the

PS. However, we consider here the worst case scenario where the PS has to generate a NotifyPresUp message for each of the associated watchers of a presentity in WCBQ. The weights were randomly generated. The more the arrival rate, the more the message generation saved by the PS. Moreover, we see that the more the throttling time, the more gain in saving generating messages. This is easy to perceive from the fact that throttling mechanisms will generate only one NotifyPresUp message for a watcher within the throttled timeframe.

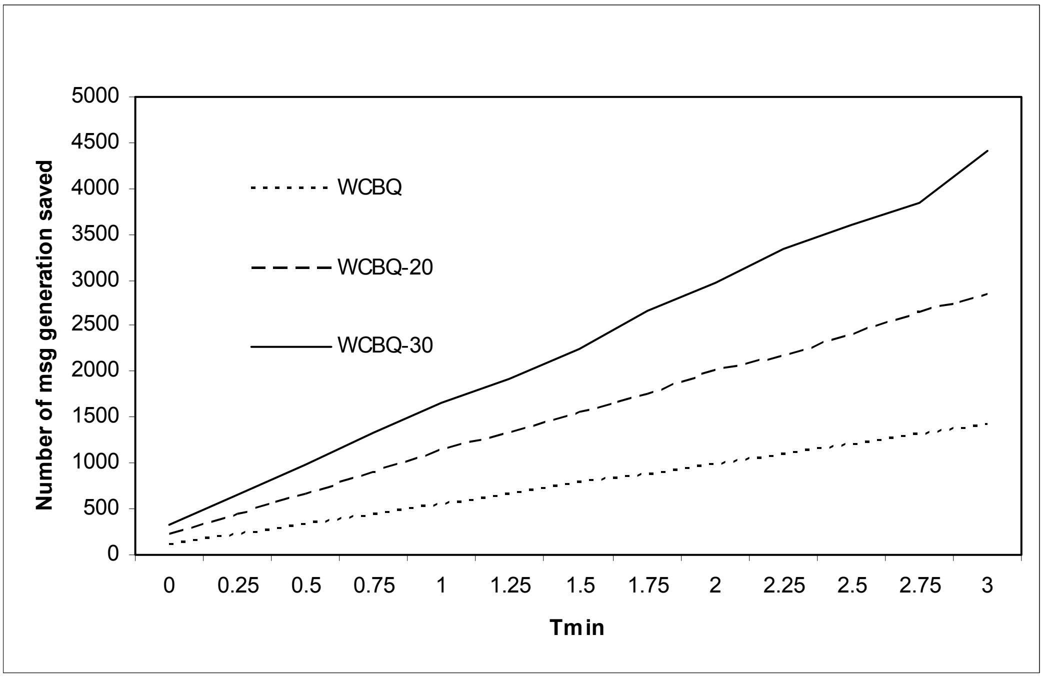


Figure 4-12: Number of message generation saved under WCBQ and throttled WCBQ

The results achieved in the above set of simulation will vary according to the input model of Table 4-1. However, we believe our WCBQ will exhibit parallel behaviour to the results presented above and will definitely reduce load for a PS.

4.4.4 Effective Bandwidth

One of the important parameters for any admission control system is the effective bandwidth. We provide theoretical expressions for our WCBQ system in this section as it is difficult to capture the exact behaviour. The effective bandwidth for FCFS admission control, B_{eff_FCFS} can be obtained from [196] if the messages at PS behave as elastic traffic. For a given average allocated rate of bandwidth b for a class, it is defined as in [196]:

$$B_{eff\_FCFS} = K.h \tag{4-32}$$

where,

$$h = \left(\frac{1}{b} + \frac{1}{\lambda_j l} \right)^{-1} \tag{4-33}$$

K is the number of classes/sources, l is the file size and λ_j is the arrival rate of a class j.

It is difficult to estimate the total bandwidth savings by using the throttle mechanism over a subscription, since such estimates will vary depending on the usage scenarios. However, it is easy to see that given a subscription where several full state notification would have normally been sent in any given throttle interval, a throttled subscription would only send a single notification during the same interval, yielding bandwidth savings of several times the notification size. With partial-state notifications, drawing estimates is further complicated by the fact that the states of consecutive updates may or may not overlap. However, even in the worst case scenario, where each partial update is to a different part of the full state, a throttled notification merging all of these n partial states together should at a maximum be the size of a full-state update. In this case, the bandwidth savings are approximately n times the size of the NotifyPresUp header. It can be observed that, the available compression schemes may be applied simultaneously with the WCBQ or throttle mechanisms for compound bandwidth savings.

When the message flows are elastic, the effective bandwidth required by an additive flow in a class j can be written as:

$$\sum_{j=1}^{K} N_j \alpha_j + \alpha_j \leq 1 \qquad (4\text{-}34)$$

with

$$\alpha_j = \frac{\lambda_j a(1-\phi_j(-(\log c)/a))}{\log c} \qquad (4\text{-}35)$$

where

$$a > 0, c \in (0,1), \phi_j(\theta) = \int_0^\infty \exp(\theta y) dG_j(y). \qquad (4\text{-}36)$$

The detail of the above expressions with parameters are provided in Appendix B which is the application of Kingman's (1970, [195]) result into the G/G/1 system [133] that is merged into the M/G/1[162] system to be applicable to a flow of our WCBQ.

4.4.5 Transition Probabilities

The transition probabilities of presentity's states can be computed from a simple Markov model. The arrival rates are considered to be equivalent to the steady state probability of presentities. Let the number of states for a presentity to change be arbitrary. The activity elements of a presentity can hop among any state from its initial state which can be modelled as a pure birth process. However, the probabilities of coming back to its initial state are equivalent. The scenario is depicted in Figure 4-13. We assume that state zero is the initial position of a presentity which may be thought of its actual anchoring position. The other states may represent the presentity's state change to busy, idle, not available etc. These state changes reflect the different values of the activity elements for instance; class, content-type, place-type, privacy, relationship, sphere etc. in the RPID (Rich Presence Information Data Format) extension. We also assume that the presentity's initial state is saturated so that upon completion of one state change, it will enter to another statically identical state instantaneously. Let, p the rate that denotes the presentity's state change and q denotes the state transition rate at which the presentity changes state from m to state 0.

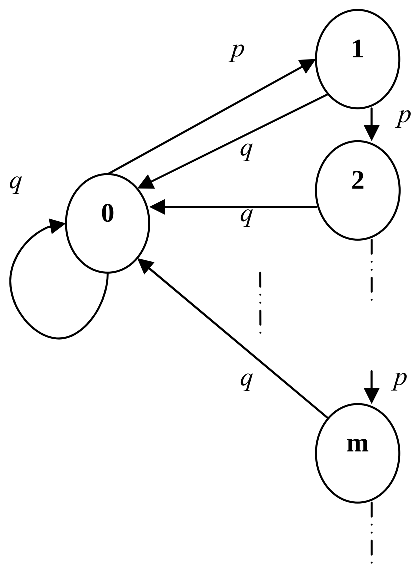


Figure 4-13: Markov chain for a presentity's states

Let, v_0 and v_m be the equilibrium probability of state 0 and m respectively.

The transition probability matrix P of the Markov chain is given by:

$$P = \begin{bmatrix} q & p & 0 & 0 & \ldots\ldots & 0 \\ q & 0 & p & 0 & \ldots\ldots & 0 \\ q & 0 & 0 & p & \ldots\ldots & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdots\cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdots\cdot & \cdot \\ q & 0 & 0 & 0 & \ldots\ldots & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ & & & & & \end{bmatrix} \tag{4-37}$$

We assume the Markov chain is finite with *m* states, then by solving linear equation:

$$v = vP \tag{4-38}$$

And the normalization condition:

$$\sum_{i=0}^{m} v_i = 1 \tag{4-39}$$

By the law of total probability for state *0,*

$$v_0 = \frac{q}{p+q} \tag{4-40}$$

For state 1,

$$pv_0 = pv_1 + qv_1 \tag{4-41}$$

i.e.,

$$v_1 = \left(\frac{p}{p+q}\right)v_0 \tag{4-42}$$

For state 2,

$$\begin{aligned} pv_1 &= pv_2 + qv_2 \\ v_2 &= \left(\frac{p}{p+q}\right)v_1 \\ v_2 &= \left(\frac{p}{p+q}\right)^2 v_0 \end{aligned} \tag{4-43}$$

Similarly,

$$v_m = \left(\frac{p}{p+q} \right)^m v_0 \qquad (4\text{-}44)$$

Substituting Eq. (4-40) into Eq. (4-44),

$$v_m = \left(\frac{p+q-q}{p+q} \right)^m v_0$$
$$v_m = (1 - v_0)^m v_0 \qquad (4\text{-}45)$$

4.5 Cost Consumption for PS

In this section, let us represent the cost of an IMS presence system in terms of number of messages generated by the Presence Server to provide the presence service that includes: (a) The PS has to generate 200OK message to acknowledge receipt of RPID message from each presentity (see Figure 4-1 in section 4.1); (b) The PS has to generate NotifyPresUp message to notify each associated watcher of a presentity about the presentity's state change (see Figure 4-1 in section 4.1). In summary, if a presentity changes state, the PS generates one 200OK message and number of NotifyPresUp messages upon receipt of a RPID message. Let H be the total number of IMS presentities observed by the IMS watchers via a PS in the system. Thus, assuming no RPID is corrupted, the total cost of presentities movement for FCFS is:

$$C_{FCFS} = \sum_{d=1}^{H} v_{md} (1 + M_d) \qquad (4\text{-}46)$$

where, M_d is the number of NotifyPresUp messages generated by the PS for a state change of the d_{th} presentity to notify its watchers. 1 is added due to the 200OK message generation upon receipt of a RPID. v_{md} is the steady state probability of the state change of the d_{th} presentity which is presented in Eq. (4-45) in section 4.4.5.

Eq. (4-46) can be used to compute the cost for a length of time. Thus, the total cost of a presence system at steady state in a real-time interval *T,* in the long run on the average for FCFS is:

$$C(t)_{FCFS} = \int_0^T C_{FCFS}\,dt \approx T\sum_{d=1}^{H} v_{md}(1+M_d) \tag{4-47}$$

Let, $U_d = f(T_{\min}) = f(\eta_j, v_{md}, \mu_j, D_j)$ is the rate in the PS, the RPID messages are dropped for the d_{th} presentity. Thus, the total cost for our dropping algorithm WCBQ in a real-time interval *T* is:

$$C(t)_{WCBQ} \approx T\sum_{d=1}^{H} v_{md}(1+M_d) - T\sum_{d=1}^{H} U_d M_d\,, \quad U \geq 0 \tag{4-48}$$

$$C(t)_{WCBQ} = T\sum_{d=1}^{H} \{v_{md}(1+M_d) - U_d M_d\}, \quad U \geq 0 \tag{4-49}$$

U is zero if the dropping is not applied to a class.

4.6 Simulation for Cost Consumption

We generated a simulation environment with Opnet Modeller. The environment considers that a PS was serving 1000 IMS terminals which were watching each other and generating RPID. Every terminal has a watch list that indicates the terminals it is watching. We kept all the watcher list size to be maximum i.e. 100 to accomplish the justification of our work. Since, the number of watcher associated with each terminal was fixed (100) we needed not to classify further in a class according to the weight. Every terminal also has a list of watcher watching it with the associated arrival rate. Here, arrival rate represents the class since classes are distinguished by message arrival rate i.e., the presentities message generation rate to the PS. We assumed that the group 3 messages were arriving from 51 message/minute to 100 messages/minute i.e., there are 50 classes in the PS from class number 51-100 in group 3. We applied our message dropping mechanism to these 50 classes. The total channel capacity, D of the PS for these 50 classes was kept to be 25 assuming a channel can service 2 message flows i.e., 2 classes in this instance (since the number of associated watchers is equal for each input message and thus not required to classify further as weight under a class). This means a class j will require 0.5 of channel. Unlike the simulation model in section 4.4.3, we vary the traffic intensity ρ_j in a class j randomly where $0 < \rho_j < 0.5$. The simulation was run for 50 minutes. We assumed no message was corrupted and all the messages were acknowledged properly upon arrival at the PS. Both WCBQ and throttled WCBQ were compared with FCFS.

The following figure (Figure 4-14) shows the number of terminals generating messages to the PS inside a class at the class rate. This is actually the computation of η_j for each class j which is required to compute the number of associated watchers and dropped messages. We see that the plots of η_j are random since the arrival rate/message generation rate was generated randomly for each of 1000 terminals.

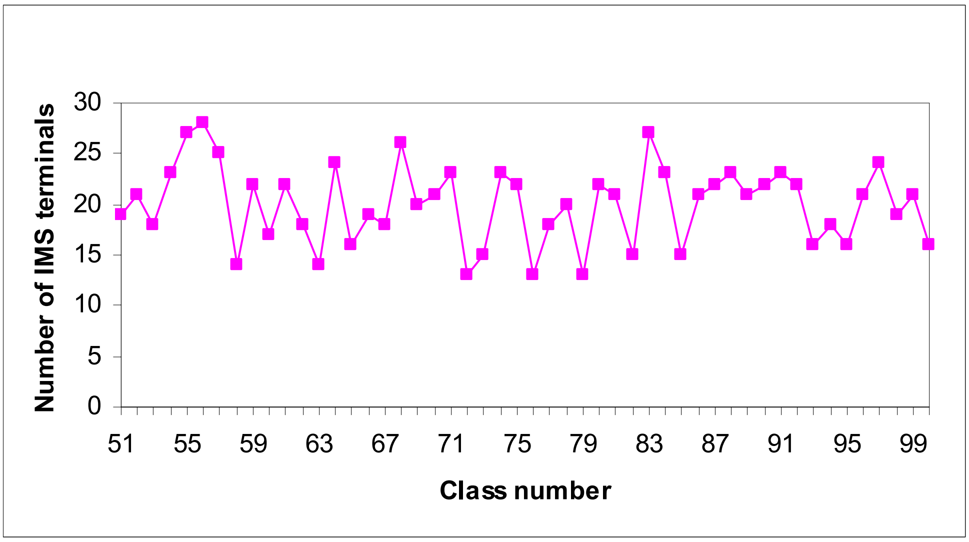


Figure 4-14: Number of terminals watching at the class rate

Figure 4-15 represents the average number of messages dropped in every minute in a class on the average. We see a sharp growth of dropping at the later classes as the later classes are served later and the threshold time is less for these classes. The spikes represent the random behaviour of the parameters of the simulation.

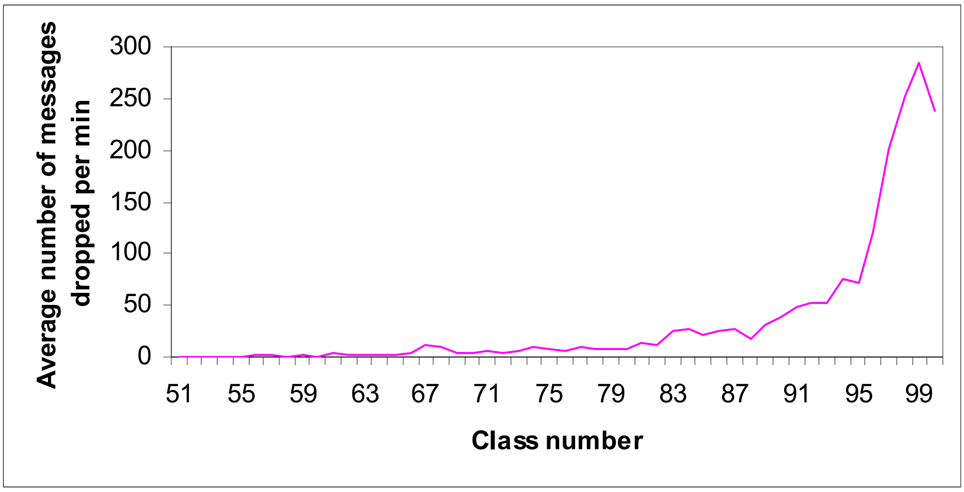


Figure 4-15: Messages dropped in a minute on average

The total number of messages dropped in the simulation lifetime is provided in the following figure (Figure 4-16). Again, we see that the later classes produce high number of message-drop. This graph is the consequence of Figure 4-15.

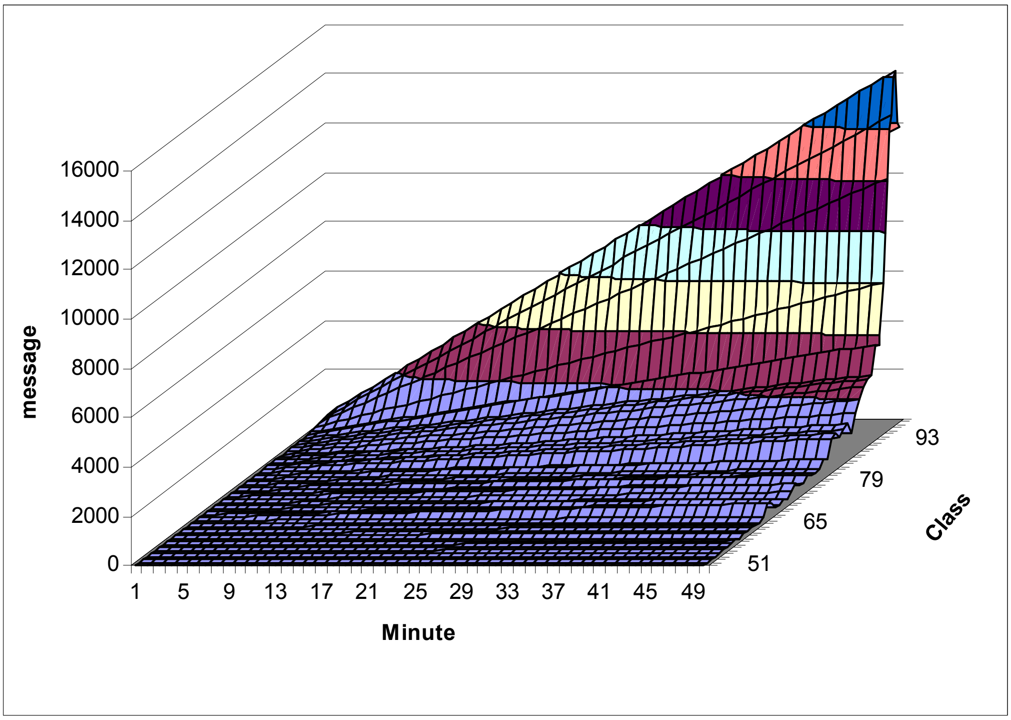


Figure 4-16: Cumulative message drop during simulation period

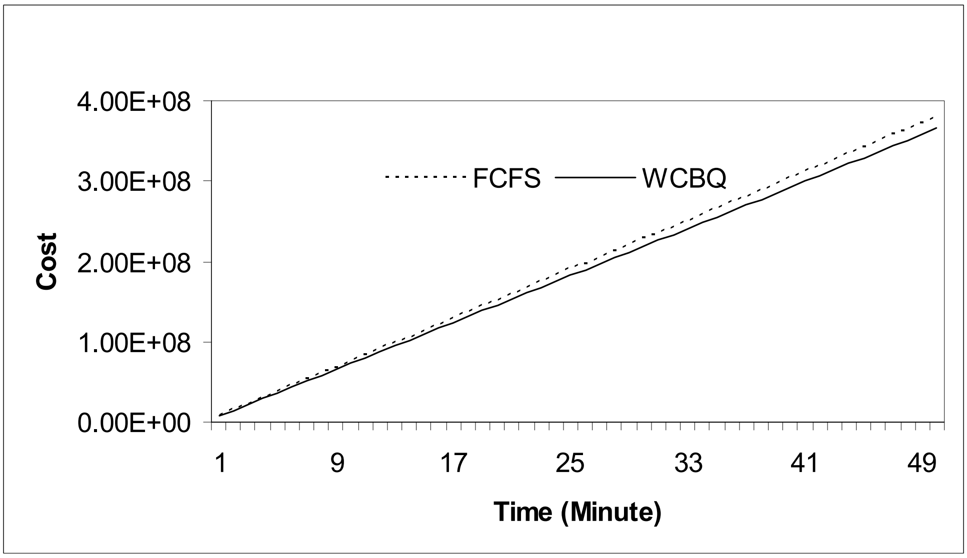


Figure 4-17: Comparison of message generation cost

Figure 4-17 shows the cost comparison of WCBQ and FCFS. According to the cost computation expressions, the cost represents number of messages generated by the PS due to the arriving RPID messages at the PS. Since the every terminal is watching to its maximum capacity i.e.100 watchers and since a 200OK message needs to be generated for every RPID arrival, the total number of messages generated by the PS for each arriving RPID is 101 (100 NotifyPresUp and 1 200OK message) for FCFS. For WCBQ, 100 messages were saved due to every RPID drop. The PS still needed to generate one 200OK message to acknowledge the generating terminal for every dropped RPID. Thus, WCBQ cost was computed by subtracting 'number of dropped messages * 100' from the corresponding FCFS cost. As mentioned earlier that the message size were less than IP packet. We used the Process module to initiate the message arrivals, Queue module to set the service behaviour and Sink module to count messages and dispose the message to free up memory (see Figure 4-18). The built in process module 'acb_fifo' of OPNET modeller was used to emulate FCFS system in an infinite buffer environment. We considered that the arriving message sizes are equal in a class and thus the service rate is same for an individual class. The cost difference was found 1.48E+07 in the final minute of simulation.

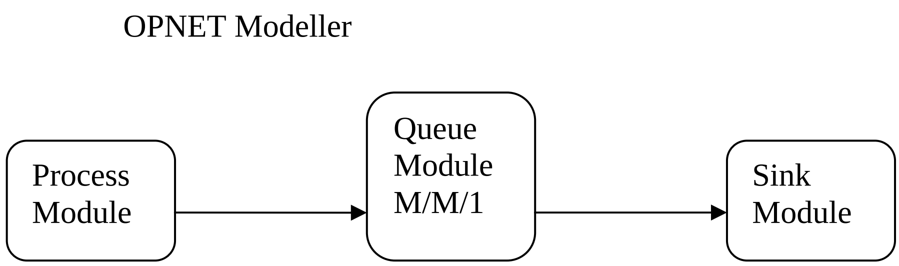


Figure 4-18: Network Topology of message streams for FCFS

Next, we compared the performance of throttled WCBQ-20 and WCBQ-30 with WCBQ and FCFS. Since with the throttled mechanism, the minimum elapsed time between two consecutive NotifyPresUp messages destined to one terminal is 20 seconds for WCBQ-20 and 30 seconds for WCBQ-30, it means that each of 1000 IMS terminals in the simulation receives three (60 seconds / 20 seconds) for WCBQ-20 or two (60 seconds / 30 seconds) for WCBQ-30 NotifyPresUp messages in every minute depending on the RPID generation rate of the terminals they are watching. For this, the message generation rate of the corresponding terminals of every node was calculated to determine whether message was needed to be generated after every minimum throttle period. Since our simulation was performed for the heavily arrival rated messages, we found that every terminal was destined to receive a NotifyPresUp after the minimum throttle period of 20 and 30 seconds. The watcher list was traversed for every node to find out how many number of terminals was watching a node with what rate. The following figure (Figure 4-19) shows the average number of watchers watching at each class rate for a node on the average. We find that the number is always at least greater than one i.e., there is always more than or equal to one node/terminal who is watching a node at each class rate.

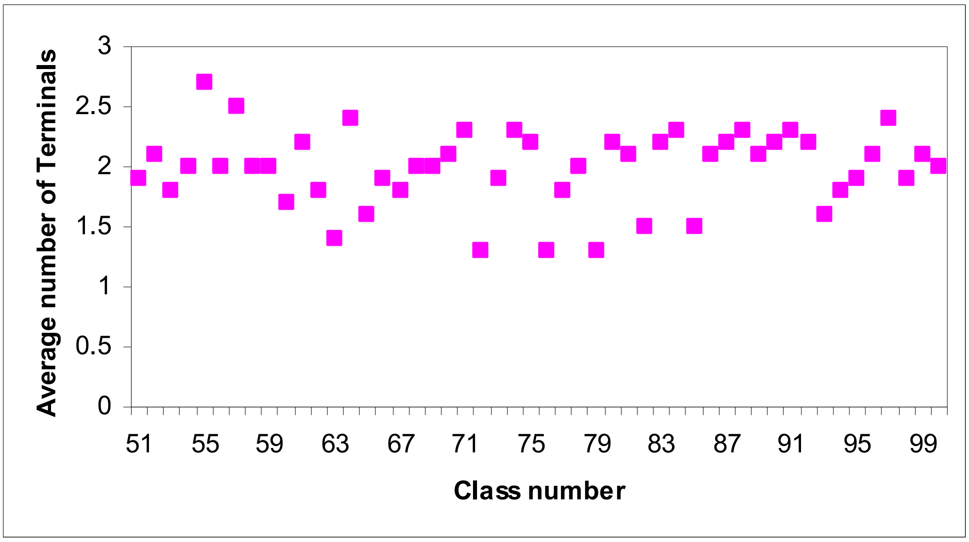


Figure 4-19: Average number of nodes watching a node at each class rate

We captured the cost for WCBQ-20 and WCBQ-30 (see Figure 4-20). It is to be mentioned that the cost of FCFS-20 and FCFS-30 are the same as the cost of WCBQ-20 and WCBQ-30 respectively since our cost function only computes the number of messages generated from the PS (both FCFS-20 and WCBQ-20 generate NotifyPresUp after every 20 secs and similarly, both FCFS-30 and WCBQ-30 generate NotifyPresUp after every 30 secs). In practical, the throttling methods differ in generating NotifyPresUp messages in terms of size as RPIDs destined to same presentity are batched and combined to produce one NotifyPresUp. But since it is difficult to compute such message size, we express out cost function in terms of number of message generation. For the costs in Figure 4-20, we computed the number of 200OK messages generated for every arrival of RPID plus the number of throttled NotifyPresUp messages generated (i.e., 1000*3 or 1000*2) for every terminal per minute.

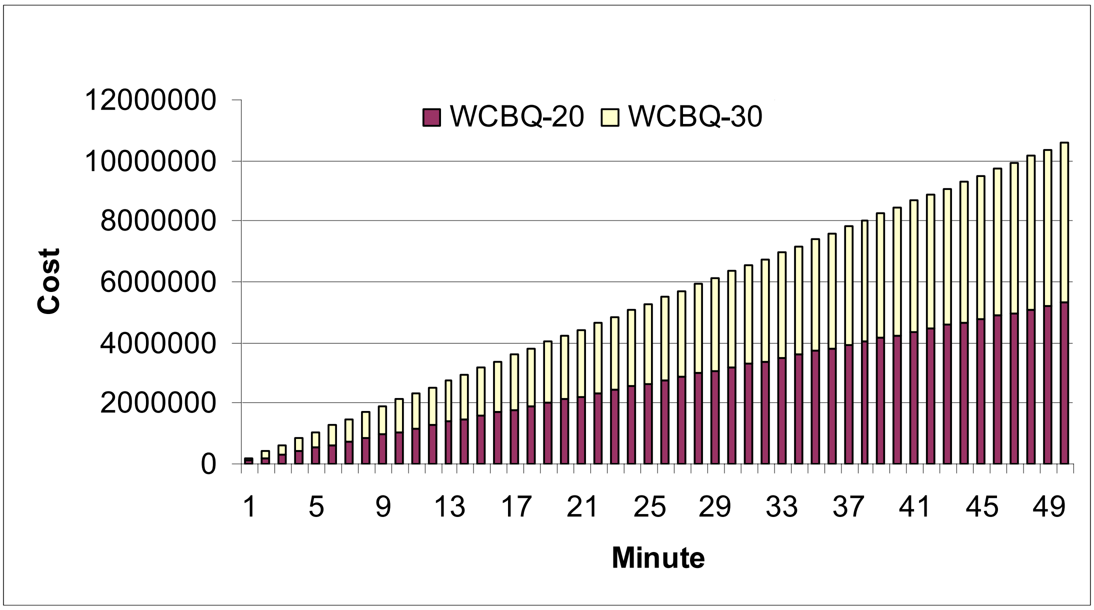


Figure 4-20: Cost comparison of WCBQ-20 vs WCBQ-30

Figure 4-21 and Figure 4-22 illustrate the number of message generation saved for WCBQ-20 and WCBQ-30 with respect to FCFS and WCBQ respectively. The cost of WCBQ-20 and WCBQ-30 were subtracted from the cost of FCFS and WCBQ to find the number of saved messages during the simulation lifetime. The computed difference during the final minute of the simulation runtime with respect to FCFS was found to be 515650000.00 and 515700000.00 for WCBQ-20 and WCBQ-30 respectively where as the computed difference with respect to WCBQ was found to be 498293800.00 and

498343800.00 for WCBQ-20 and WCBQ-30 respectively. The results of Figure 4-20, Figure 4-21 and Figure 4-22 suggest that the cost and number of message generation being saved (during simulation runtime) go up with increasing time.

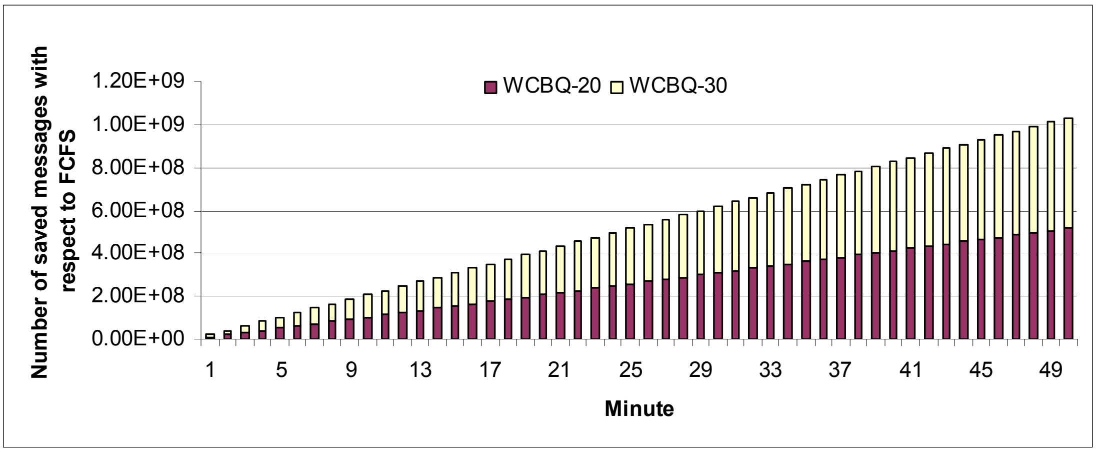


Figure 4-21: Number of message generation saved by throttled WCBQ compared to FCFS

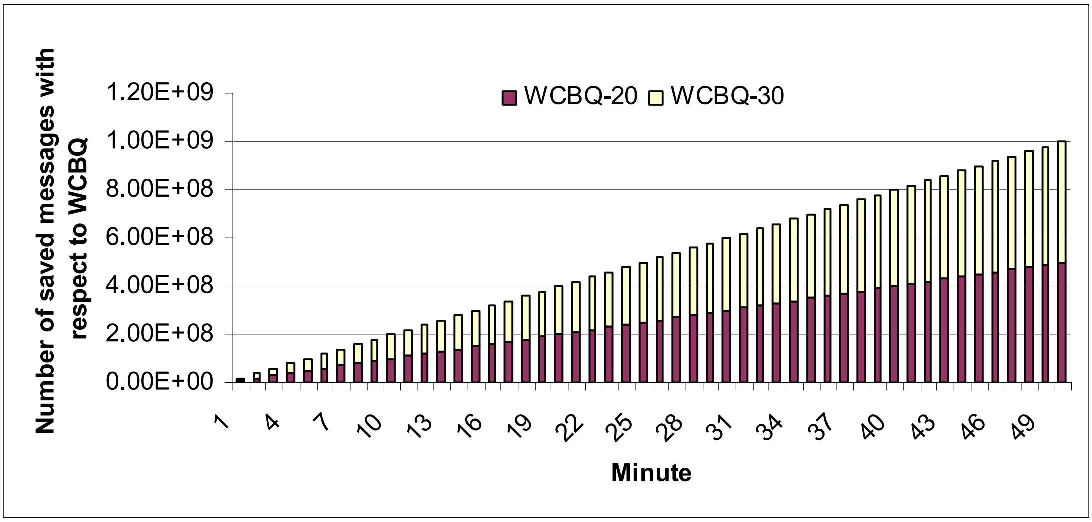


Figure 4-22: Number of message generation saved by throttled WCBQ compared to WCBQ

4.7 Method for Optimizing Subscription Time

We propose a theoretical method in this section that fits with the PS in IMS to generate the optimal subscription time for the IMS terminals [2]. According to the analytical model and the cost computation as discussed above, it is recognized that the total average cost for the presence server is a function of several parameters. In practice, the value of the watcher/IMS terminal subscription time, *T*, must be specified in the implementation of network resources. If a watcher is mobile and is visiting a network then, *T* can be defined only based on watchers sojourn time in the visited network. There are quite a number of works available today over mobility management and mobile node's cell residence time. We do not address in this thesis the mechanism by which a mobile monitors its location and velocity and such issues. A mobile terminal may determine its location through a variety of methods, including the global positioning system, signal triangulation, base-station self identifying beacons, or a combination of the above. Other methods and related references on mobile location and velocity determination can be found in [193], [194]. We find the sojourn time from [192] proposed by Guerin in 1987:

$$T = t_{sojourn} = \frac{9Z}{(3+2\sqrt{3})V} \qquad (4\text{-}50)$$

where *Z* is the "radius" of a cell and *V* is the average mobile node's velocity. The calculated rate of cell boundary crossings is 1/*t*sojourn. However, if the parameters for the cell residence time (*Z, V*) or above all, the mobility information of a watcher is not known, then cost computation Eq. (4-46) may be used with conjunction with *e* (considering the subscription rate is exponential which is practical in heavy traffic situation) to define *T*. The question is when and how a Presence Server will compute the necessary parameters for Eq. (4-46). An IMS watcher may subscribe new presentity with the PS while it joins. The number of presentities *H* is available from the watcher subscription list at any point of time. The other parameters for instance presentity mobility vectors and number of states may be computed using heuristic method. This will require the Presence Server to have extra cache and may introduce slight delay to lookup from its routing table. However, the message overhead of the generated messages is expected to be reduced significantly in return which is shown in the overhead collection later in this section.

In order to achieve the best performance, the following method may be applied for T_{op}:

$$C(t) = e^{tC_T} \qquad (4\text{-}51)$$

where, C_T may be defined from Eq. (4-46) assuming FCFS is used.

Thus the optimal subscription time algorithm can be evaluated as follows (see Figure 4-23):

1: If $t_{sojourn}$ == true

2: $T_{op} = t_{sojourn}$

3: Else

4: Compute T_{op} from (4-51)

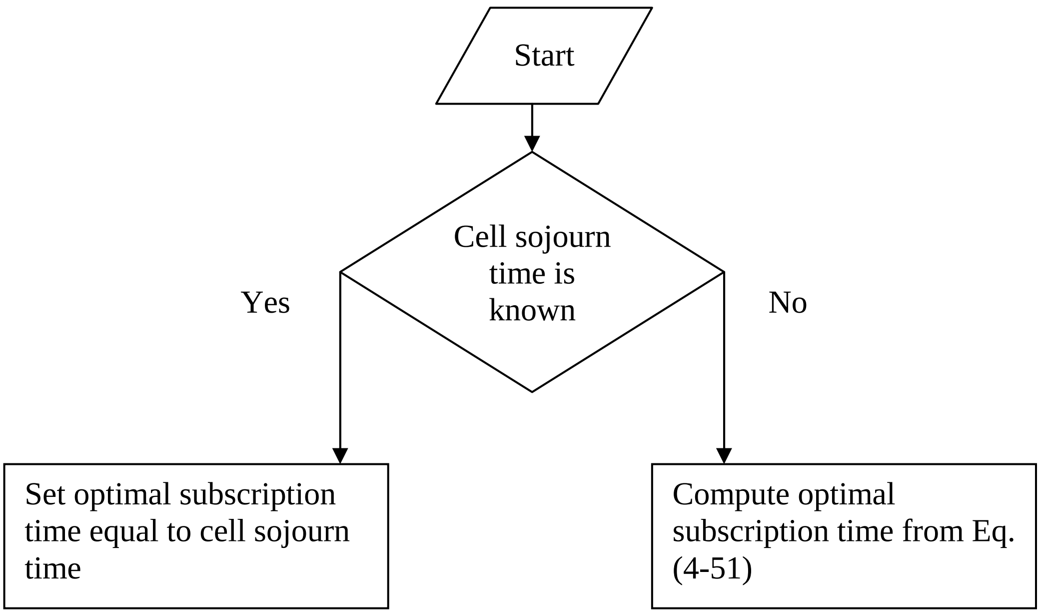


Figure 4-23: Method for optimizing subscription time

Line 1-3 of the above algorithm will take *O(1)* time to execute where as executing line 4 will take linear time, *O(H)* only.

Next we evaluate the overhead in terms of extra messages sent for various constant time values of watcher subscription time. Figure 4-24 shows the resource wasted area for a constant time set, C_{const} that is not equal to the T_{op} for the proposed curve. The figure has an intersection point, $t=T$. We argue that the optimal choice point is at the curve.

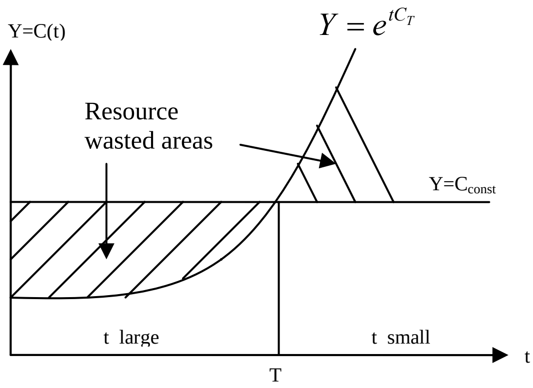


Figure 4-24: Optimal lifetime of a watcher

We denote smaller subscription time as *t_small* in the $Y=C_{const}$ line if $t>T$ and larger subscription time as *t_large* in the $Y=C_{const}$ line if $t<T$. Since, the model is based on the assumption that the IMS terminals are high in volume, we are particularly interested in the later part of the curve. Analytically, the total average overhead for Figure 4-24 is given by:

$$C_{overhead} = \int_0^T (C - e^{tC_T})dt + \int_T^t (e^{tC_T} - C)dt$$

$$\approx CT - \left.\frac{e^{tC_T}}{C_T}\right|_0^T + \left.\frac{e^{tC_T}}{C_T}\right|_T^t - Ct\Big|_T^t$$

$$\approx 2CT - Ct + \frac{1 - 2e^{TC_T} + e^{tC_T}}{C_T} \qquad (4\text{-}52)$$

Alternatively, the cost for overhead may be computed quantitatively for a single watcher at the PS. The PS has to acknowledge to a watcher/presentity with 200OK message in response to a watcher/presentity's subscription/registration with the PS (see Figure 4-25); in addition, it may notify the presence information to its joining watcher/subscriber (optional). Thus for each subscription, the PS has to generate 2 messages including the optional message (message number 3 in Figure 4-25): one 200OK message and one Notify message. The optional Notify message is generated by most of the system today in order to indicate the status of the subscription; this message can also contain XML document containing the list of watchers of the presence information.

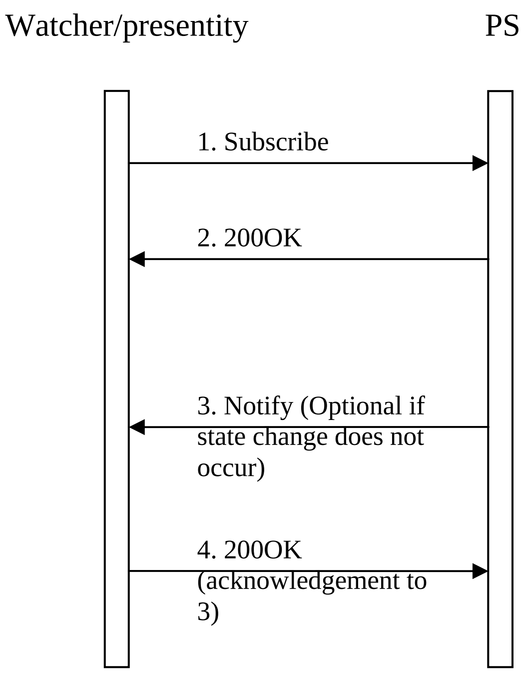


Figure 4-25: Subscription and Notification of Presence information

Thus, if the watcher subscription time is selected to be smaller than the $T_{op,}$ the watcher will have to subscribe again with the Presence Server and the information of the current presentities' (which are being watched by the watcher) status will be published to the watcher again as a routine work after it joins the Presence Server. Thus, the average cost of overhead for a watcher, C_{t_small} for smaller constant time can be defined as:

$$C_{t\_small} = 2\beta \tag{4-53}$$

where, β (>1) is the ratio between T_{op} and *t_small.* It can be easily observed that the inaccurate small constant time for large scale of watchers will be expensive.

If the watcher subscription time is selected to be larger than the $T_{op,}$ the PS will have to generate the messages for the presentities movement for all the watchers during the extra period of subscription time. This cost, C_{t_large} can be retrieved from Eq. (4-47) with time interval (*t_large*- T_{op}).

$$C_{t\_large}(t) = \int_{t\_large}^{T_{op}} C_T dt \tag{4-54}$$

4.8 Summary

The IETF engineers are still working on some optimal solution for facilitating the presence service to the IMS terminals. Sending less information in presence documents may lead to IMS users not getting a good experience with presence systems used from wireless terminals. Sending presence information less periodically will lead to an inaccurate presence view of the presentities. Obviously, there has to be a compromise between the amount of information sent, the frequency of the notification, and the bandwidth used. The job of a PS is to process and distribute information about the presence of entities in the system regarding reachability, availability, and willingness to communicate etc. When there are a large number of entities in the IMS that wish to know presence information about a large number of other entities, and if those entities have rapidly changing presence information, then a large number of messages must be processed and distributed by the PS. This can cause the PS to become overloaded. But not all messages are of equal importance, and that PS can use message importance to its advantage in reducing load. Instead of using event filtering or throttling that has been proposed elsewhere, we proposed a scheduling and message dropping mechanism based on priority in this chapter. It not only considers the types of messages but also the demand for those messages when dropping them. This allows messages to be dropped from lower priority presentities only when necessary. We showed how to limit the numbers of messages that are entered into the PS as a means of controlling the service and performance of these priority queues to avoid starving the lower priority queues. We provided a thorough analysis where the end objective of the derivations was to derive a threshold time. This time is used to decide whether existing messages in the queue should be dropped if new messages also arrive from the same presentity/flow. The threshold time for each class is derived based on the demands from each priority class and the capacity allocations of each flow. We found that the lower priority classes outperform FCFS because of the priority scheduling and dropping approach. The results of cost consumptions reveal the same outcome. In summary, the idea of using prioritized scheduling for managing the demand on the PS is helpful compared to other approaches.

Our WCBQ is a preliminary work of a novel queuing mechanism to provide class differentiation and to reduce the load in the IMS presence server during heavy traffic. The grouping was done to assign priority on the arrived messages. In our test-bed, the dropping application was limited to group 3 only in order to balance the real time view and the notification rate. The tasks of admission controller for the PS are demonstrated. The optimized dropping time frame has been developed based on Lagrange multiplier technique. The PS benefits significantly from the algorithm in terms of reducing the number of message generations. The channel allocation and mean service rate will affect the performance of the PS. Nonetheless, the dropping mechanism definitely reduces load from the PS when the message arrival rate is high and the number of watchers watching presentities are large. The determination of group size and the application rate of our WCBQ are to be configured by the IMS presence service providers. We demonstrated that WCBQ with throttle mechanism performs better in terms of generating messages; though their blocking probabilities are expected to be high.

We further developed a theoretical model in this chapter to optimize the watcher subscription time in the IMS presence service. The optimal life time of the watcher will reduce the signalling cost for the Presence Server. As an application of the mathematical model in the IMS, an algorithm for dynamically setting the watcher subscription time is proposed in the context of available IMS parameters. The overhead is also depicted when the watcher subscription time is not set carefully. A future research direction would be to generate a test-bed to test the optimized subscription time method within the IMS framework.

Chapter 5 Dimensioning Push-To-Talk Over Cellular Service

5.1 Introduction

"Push-to-Talk is a forerunner to peer-to-peer services over IP, for which IMS provides the capabilities and foundation. PoC is the first commercial application based on IMS" [138]. The driving forces behind the operators' Push-to-talk initiatives are the search for new revenue opportunities and finding ways to increase subscriber acquisition and reduce churn. In this chapter, we depict some of the key areas based on the OMA release [112] to dimension a PoC network service.

Comparing different PoC solutions from a radio resource utilization perspective is interesting from a technical perspective. A generic radio access network can be divided into three parts [138]. All sites are categorized into three categories as described in the following Table 5-1 [138]. For site categories number one and three, it is indifferent in technology chosen. For site type number two, the cost difference is directly linked to the resource utilization. Our work is to identify optimal points while installing additional resources in category 2 i.e., to define and analyse resource usage optima.

Table 5-1: Cost model for introduction of Push-to-Talk service [138]

1	2	3
Full base stations and no spare capacity.	No spare capacity but capacity expansion possible by installing additional capacity cards.	Pure "coverage" sites with spare capacity.
Introduction of PTT means new sites and new radio plan.	Introduction of PTT means investment in new capacity cards and some alternations in radio plan. Cost proportional to resource consumption of new service.	Introduction of PTT means no new investments.

As mentioned in the literature review (section 3.1.4 in chapter 3) that, the related work available today focuses on the performance analysis over PoC. An architecture for enabling PoC services in 3GPP networks has been furnished by Raktale S. (2005) in [139]. Similar work is reported by Parthasarathy A. (2005, [140]). The design of a PoC service operated over a General Packet Radio service / Universal Mobile Telecommunications System (GPRS/UMTS) network is depicted by Kim *et al* (2005, [141]). The PoC performance is analysed over GPRS by Balazs (2004) in [137]. However, these works are ignorant about dimensioning PoC service to optimize revenue for service providers. The basic challenges that affect the end-to-end service performance for PoC are: (a) Network configuration and dimensioning, (b) Timer settings in terminals and networks, (c) Traffic handling priorities used, (d) Service option choices such as early media session establishment; and (e) Client implementations on the terminals native operating systems. In this chapter, we add controls to a PoC server to be able to perform efficiently during busy hour. We dimension the PoC service based on the assumption that the network Grade of Service (GoS) is provided. In this way, a PoC server is able to control PoC functionalities to the optimal level. GoS is a measure of the blocking probability of an incoming call.

Usually, a PoC Radio Access Network (RAN) infrastructure is dimensioned for 1%-2% GoS for PoC sessions. This means that the network should block less than 1%-2% of all incoming PoC sessions during busy hour. The contributions of our work in this chapter are [1]:

i. Optimize traffic flows for the available Transmit/Receive Units (TRU) of a PoC Base Station (BS);
ii. Controlling access of special sessions based on available TRUs;
iii. Optimize the session timer for a PoC controller;
iv. Optimize number of session initiation for a PoC client during busy hour.

5.2 The Four Problem Overview

Recapping the PoC server description from section 2.6 in chapter 2, it implements the application level network functionality for the PoC service. The PoC server performs a Controlling PoC Function and/or Participating PoC Function. The Controlling PoC Function and Participating PoC Function are different roles of the PoC server [112]. The determination of the PoC server role (Controlling PoC Function and Participating PoC Function) takes place during the PoC session setup and lasts for the duration of the whole PoC session. Each session is controlled by one controlling function. PoC server performs the following when it fulfils the controlling PoC function: (a) Provides centralized PoC Session handling, (b) Provides the centralized media distribution, (c) Provides the centralized Talk Burst Control functionality including Talker Identification, (d) Provides Session Initiation Protocol (SIP) Session handling, such as SIP Session origination, release, etc. (e) Provides policy enforcement for participation in Group Sessions, (f) Provides the Participants' information, (g) Provides for privacy of the PoC Addresses of Participants, (h) Collects and provides centralized media quality information, (i) Provides centralized charging reports, (j)

Supports User Plane adaptation procedures and (k) Support Talk Burst Control Protocol negotiation [112]. The work presented here is to dimension a PoC service based on resources available at the cell base station or Radio Access Network (RAN) infrastructure. The PoC controlling/participating function of the PoC servers will be able to perform according to the blocking requirement of the RAN.

Long sessions Vs short sessions: As mentioned in the OMA release [112], PoC usage has two main scenarios: 1. Short interactive sessions (Type 1) and 2. Long session with sporadic, interactive talk periods (Type 2). Figure 5-1 illustrates these two types of PoC sessions. The distinction between the two talk is that one contains chat sessions after long intervals within a single session where as the other refers to the separate sessions for each talk. The key challenge is to reduce the delay involved as well as message flows in PoC session set up schemes. It has been mentioned in the literature review (section 3.1.4) that pre-established session takes less time than that of on-demand session as registration is performed beforehand for pre-established PoC session set up. However, the extra steps that need to be performed are the state transition from STANDBY state to READY state in the pre-established sessions. When the READY timer expires, a PoC terminal shall return to STANDBY state. The READY timer that controls the time a PoC terminal remains in READY state is set by the operator. The steps to be performed for state change are

a) Paging with which the PoC server defines the location of the PoC terminal on cell level,
b) Cell update with which the terminal tells the PoC server in which cell it is located
c) Radio resource assignment procedures which are the part of session set up procedure and finally
d) PoC signalling.

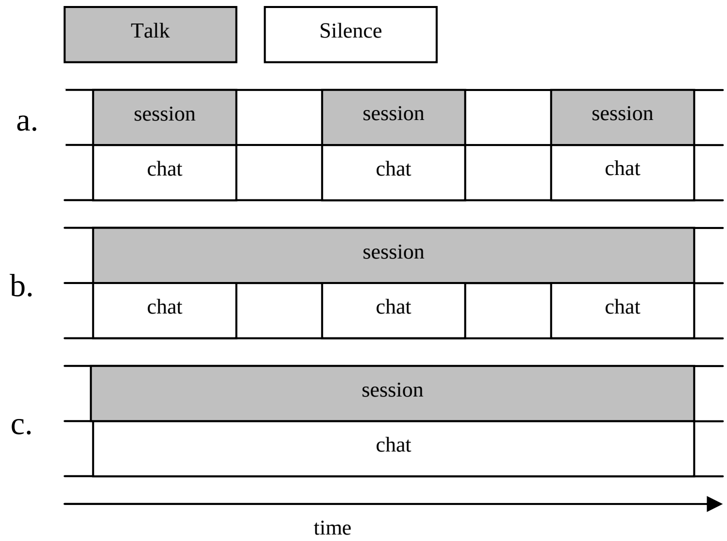


Figure 5-1: (a) PoC short session (b) PoC long session and (c) Normal phone call

Obviously the long sessions will prefer a pre-established session than on demand session set up. Moreover, an IMS PoC client achieves the independence of initiating as many PoC sessions as it wants with pre-established session set up. We define the access control of these two kinds of session set up in this chapter. Priority is provided to on demand session set up based on number of available and busy TRUs. The capacity of PoC framework is measured by TRUs of the base station which has the direct impact on cost analysis of the PoC service. A TRU can transmit on eight time-slots and receive on eight time-slots i.e., eight time-slot pairs. One time slot can share 5.48 sessions in GPRS and a TRU can support 43 simultaneous PoC sessions [138]. Usually, a cell will have 5 installed TRUs. The on-demand sessions can use any free TRU while pre-established sessions can use a TRU only when total number of busy TRUs is less than some fixed number (threshold/protection level). This way pre-established sessions will be forced to be initiated as on-demand sessions after the protection level of TRUs is exceeded. Thus a number of message flows will be reduced for each session as there are few extra steps involved in the state change for pre-established sessions.

Session timer: One of the basic challenges for a PoC service provider is the timer setting for a PoC session. The length of a PoC session timer should be carefully chosen in dynamic manner. A constant cut off time of a session will affect system performance and consequently reputation of service providers. A long timer setting will incur traffic overhead at the PoC server queue whereas a short timer setting will generate frequent requests from the PoC clients, consequently will increase message flows in the long run. We derive a simple relation to control the session lifetime based on network GoS, time slot duration and number of TRUs installed. The PoC controller is able to terminate any session if it exceeds the timer setting during busy hour from the derivation provided in this paper.

Path optimization: The detail of all the PoC traffic flow scenarios can be found in the OMA release [112]. A base station (BS) can be thought of as a combination of TRUs and each TRU having a number of time slots. Each time slot can serve multiple PoC sessions. A PoC session flows can be shown as in Figure 5-2. The initial INVITE messages of PoC clients go through the SIP/IP Cores (Session Initiation Protocol/Internet Protocol Cores). The SIP/IP Core is the reference point that supports/provides session signalling between PoC client and server, address resolution services, charging information, publication / subscription / notification of presence information, indication capabilities and relaying service settings including answering mode indication, incoming PoC session barring and incoming instant personal alert, etc. These types of huge traffic flows arise the niche of path optimization while passing through a TRU (Transmit/Receive Unit) of a base station. The lost traffic from a source PoC client to a destination PoC client must be minimized to avoid message re-generations. The solution to minimize lost traffic depends on the parameters of link offered traffic and the blocking probabilities at the TRUs. In this research work, we compute the optimized path for the available TRU of the base station to share the load and minimize the traffic overflow. The traffic flows are controlled by the controlling PoC function of the PoC server assigned to the originating PoC client.

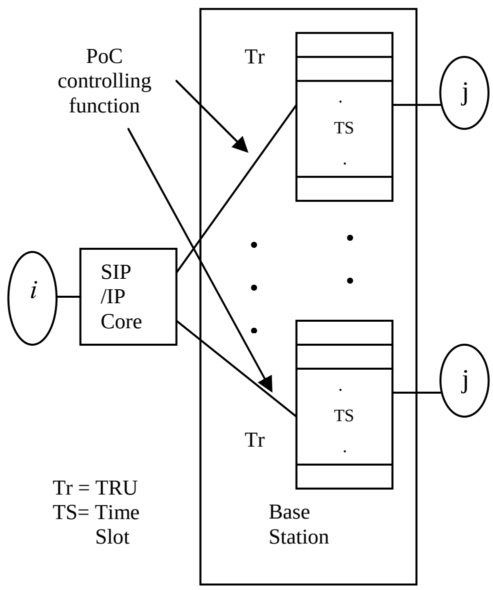


Figure 5-2: PoC route optimization between two PoC clients: *i* and *j*

Simultaneous sessions: A PoC client should not be allowed to initiate or take part in as many long sessions as it wants to initiate/join in busy hour as that may introduce congestion and performance degradation at the PoC server. The controlling PoC function must be able to limit the number of simultaneous sessions initiated by a PoC client. Simultaneous PoC session means a PoC client being the participant in more then one PoC session simultaneously. We introduce a simple two state Markov mechanism to optimize the number of simultaneous session for a PoC client during rush hour. Our derivation leads to an optimal number based on system resources.

The time complexity of all the derived computation is negligible which strengthens the justification of our work. We believe a PoC service provider will be benefited from the adoption of the models presented in this paper.

5.3 Model Assumptions

The typical nature of a PoC session is depicted in Figure 5-1. The PoC session consists of chats and pauses. The number of chats of long sessions is greater than that of small sessions. In fact, the statistical analysis shows that the voice activity factor has found to be 67%. That means that 33% of a conversation is actually pauses and silence [138]. The throttled arrivals (of these kinds of chats) with Poisson model has been extensively studied in many text books. The inter-arrival time of session follows the negative exponential distribution (NED) and the probability density function (PDF) with arrival rate λ takes the form $p(u) = \frac{\lambda}{u^2} e^{-\lambda/u}$ and the corresponding cumulative distribution function is $C(u) = e^{-\lambda/u}$. Appendix E provides the corresponding proof of these two equations which may be found in many textbooks [162]. The graphical representation of these functions has been studied by Liu (2002) in [184] which are provided in Figure 5-3 and in Figure 5-4. The interval rate takes a skew for growing value of the function. The chat arrivals of a session are considered to be Poisson process. We control the access of these types of inter-arrival rates during rush hour which is discussed in subsequent sections.

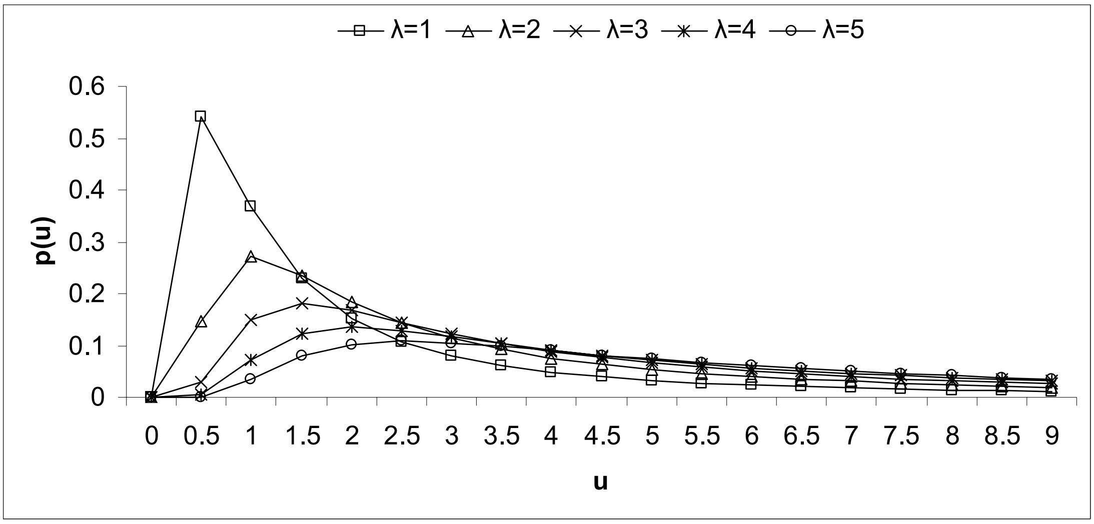


Figure 5-3: Behaviour of session inter-arrival rate in terms of probability density function

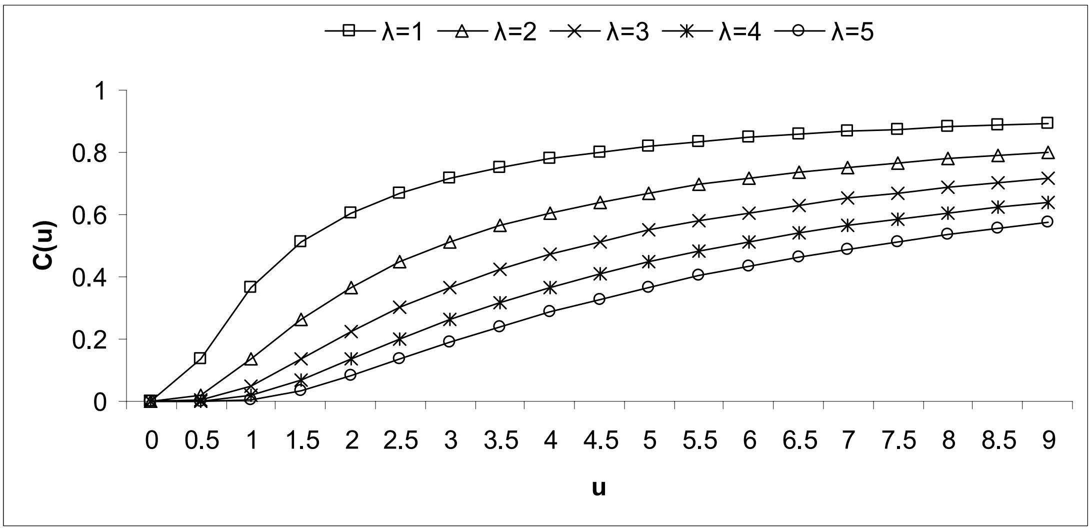


Figure 5-4: Behaviour of session inter-arrival rate in terms of Cumulative distribution function

For a mobile network operator launching a Push-to-talk service, some investments in new RAN infrastructure are required. Our models are derived based on the resources available at the BS/RAN. The assumptions made for the presented models are

1) Capacity dimensioning of the RAN is directly linked to the expenditure of installed TRUs,

2) Each cell has installed TRUs,

3) One TRU contains 8 time slots as in [138],

4) The minimum unit of service rate is based on the number of PoC session-chats getting serviced by a time slot,

5) A time-slot duration is 20ms (which is practical in Code Division Multiple Access system) unless specified.

Since, we dimension the PoC service based on the given GoS and since each PoC session is handled by a controlling function of a PoC server, we assume the controlling policy will be set at the PoC server to function according to the models derived in this chapter.

5.4 Controlling Session Access

As mentioned earlier, Type 2 (pre-established) sessions should not be allowed during the busy hour where as type 1 (on-demand) sessions should be able to use any free TRU. Let, a Type 2 session can use a time slot only when the total number of busy TRU is less than some protection level of number *b.*

We denote, λ, λ_1, and λ_2 as total arrival rate of PoC session chats, arrival rate of type 1 session-chats and arrival rate of type 2 session-chats respectively. Since we let both session chat arrivals as Poisson processes, the session chat arrivals in the base station are exponentially distributed with mean $1/\lambda_1$, $1/\lambda_2$. As mentioned in the model assumption section before, if we represent the service rate as the service rate of the chats (not the whole session) then we also consider service time to be exponentially distributed. Let, μ represents the mean service rate of chats for both Type 1 and Type 2 sessions where number of Type 2 chats (Figure 5-1.b) is greater than that of Type 1 chats (Figure 5-1.a). Treating the service times of chats as independent identically distributed random variables, the common distribution becomes exponential with mean $1/\mu$. Thus, if the chats are served in their order of arrival then the system can be viewed as the birth and death a process of M/M/m queuing machines where *m* is equal to number of TRUs, *b*. The system model of M/M/m queuing system is provided in Appendix C. We apply the model with variation of the restricted arrival change to our context. The corresponding Markov state change model with probabilities is presented in Figure 5-5. A state κ represents the number of chats present in the PoC BS. Under the above assumptions, we find:

$$\lambda_\kappa = \begin{cases} \lambda_1 + \lambda_2 & if \;\; \kappa < b \\ \lambda_1 & otherwise \end{cases} \tag{5-1}$$

$$\mu_\kappa = \kappa\mu \tag{5-2}$$

where, μ_κ is the mean service rate in state κ

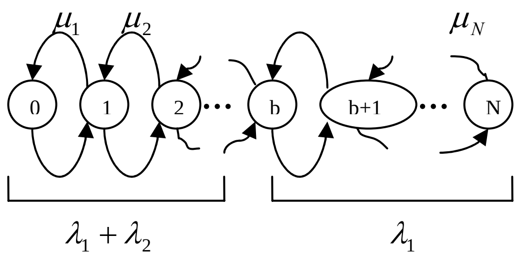


Figure 5-5: Markov model for accessing session

The steady state probabilities are:

$$p_\kappa = \begin{cases} p_0 \dfrac{1}{\kappa!}\left(\dfrac{\lambda_1+\lambda_2}{\mu}\right)^\kappa & if\ \kappa < b \\ p_0 \dfrac{1}{\kappa!}\left(\dfrac{\lambda_1+\lambda_2}{\mu}\right)^b\left(\dfrac{\lambda_1}{\mu}\right)^{\kappa-b} & otherwise \end{cases} \quad (5\text{-}3)$$

In this system, p_0 is given by the ordinary normalization condition. Given the state probabilities, it is possible to compute all the moments of the traffic, in particular of the variance. From Figure 5-5 we have the mean of total traffic offered, $a = \frac{\lambda}{\mu} = \frac{(\lambda_1 + \lambda_2)}{\mu}$

and the ratio of Type 2 to total traffic, $r = \frac{\lambda_2}{a\mu}$. Let,

N = Total number of TRUs in the base station;

$B =$ The probability that all of the N TRUs are occupied;

$B_{b-1} =$ The probability that more than $b-1$ TRUs are busy.

B and B_{b-1} determine how much of the two types of PoC session streams will be blocked. We have,

$$B = p_0 \frac{a^N}{N!}(1-r)^{N-b} \quad (5\text{-}4)$$

$$1 - B_{b-1} = p_0 \sum_{j=0}^{b-1} \frac{a^j}{j!} \quad (5\text{-}5)$$

$$p_0 = \frac{1}{\sum_{\kappa=0}^{b} \frac{a^\kappa}{\kappa!} + \sum_{\kappa=b+1}^{N} \frac{a^\kappa}{\kappa!}(1-r)^{\kappa-b}} \quad (5\text{-}6)$$

This model can be used whenever a PoC controller wants to offer better service to one type of PoC sessions by restricting the availability of TRUs to the other type of sessions. The protection level can be adjusted based on given B and B_{b-1} in the above equations. The time complexity is dominated by p_0 which is $O(a^b b)$ for $b > \frac{N}{2}$. Since, a cell will have only a few number of TRUs installed, the computation time becomes negligible to that context.

The behaviour of B and B_{b-1} are depicted in Figure 5-6 and in Figure 5-7 respectively. The experiment was performed for $N = 5$, $\lambda_1 = 5000$ chats per sec, $\mu = 5*40$ chats per 20ms or 10000 chats per sec. μ was taken in consideration that a TRU can have 8 time slot pairs and that a time slot can serve 5 simultaneous session-chats.

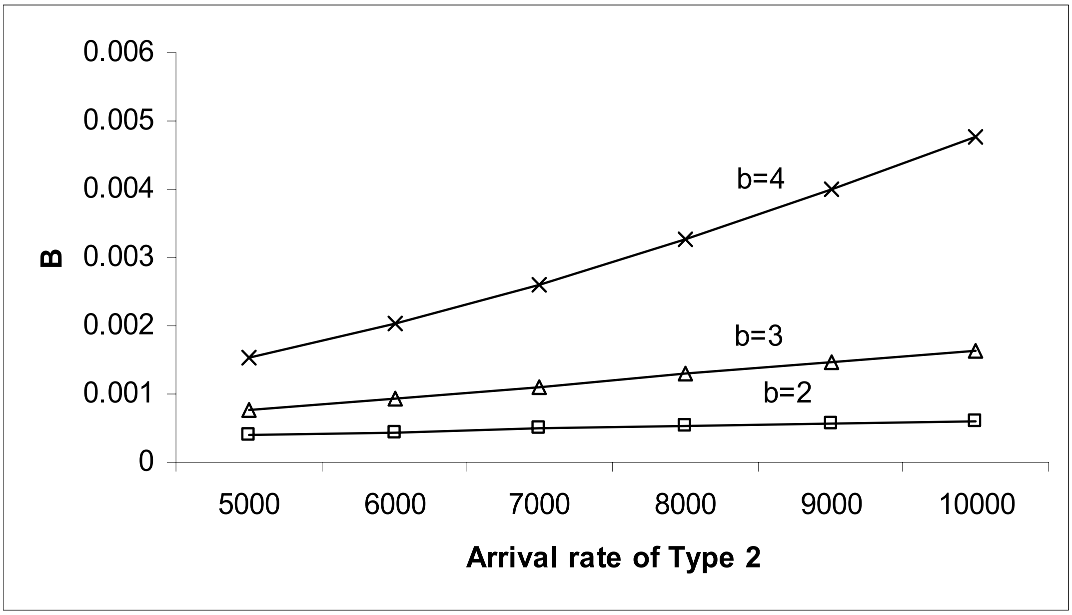


Figure 5-6: Total blocking probability for different protection level with 5 TRUs

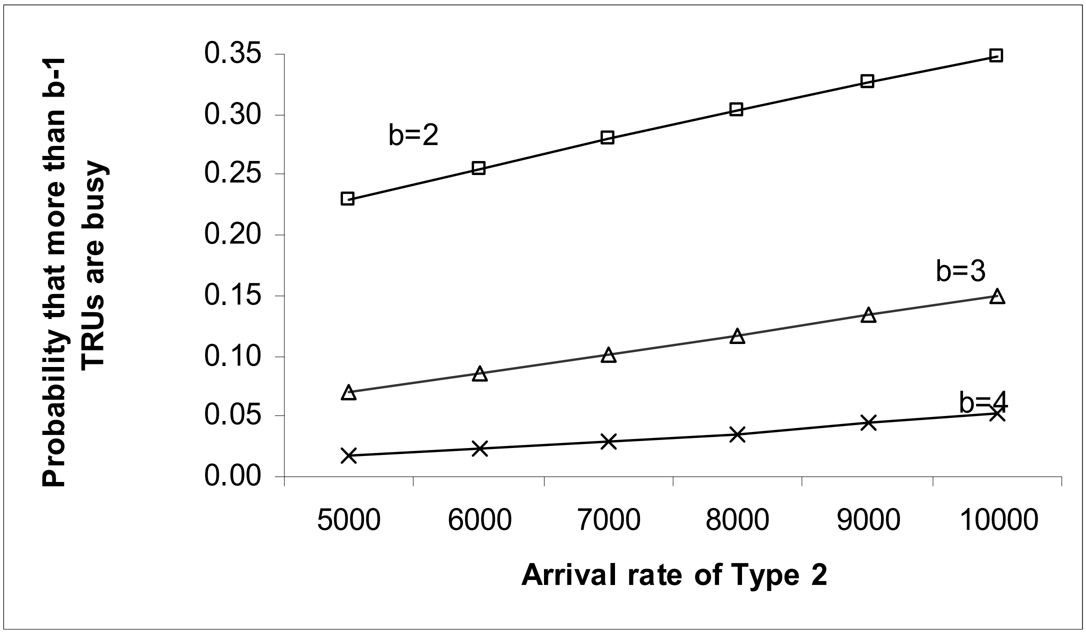


Figure 5-7: Blocking probability for protection level with 5 TRUs

The arrival rate of Type 2 session-chats was varied from 5000 chats/s to 10000 chats/s and the blocking probabilities were computed for different protection level of TRUs. The total blocking probability goes up as the protection level goes high where as B_{b-1} goes down with raising protection level.

The results in Table 5-2, for $N = 10$, $\lambda_1 = \lambda_2 = 10000$ chats per sec, $\mu = 10 * 40$ chats per 20ms or 20000 chats per sec exhibit similar behaviour. This is obvious from the fact that higher values of *b* will allow more Type 2 (pre-established) sessions. If the value of *B* i.e., network GoS is provided then, p_0 can be defined and used in Eq. (5-5) to determine B_{b-1}. The analysis presented here can be used to fix a threshold level *b,* based on GoS and session-chat arrivals. Once there is a fixed protection level, any pre-established session will be blocked to initiate as on-demand session by the corresponding PoC server after the total PoC session arrivals exceed the threshold number of TRUs.

Table 5-2: Blocking probabilities for N=10

b	***B***	***B**$_{b-1}$*
2	0.0000000004	0.2292529587
3	0.0000000008	0.0705525580
4	0.0000000016	0.0170324915
5	0.0000000032	0.0033416557
6	0.0000000063	0.0005498781
7	0.0000000127	0.0000778205
8	0.0000000253	0.0000096562
9	0.0000000507	0.0000010645

5.5 Load Sharing at PoC BS

We assume that the traffic offered is given to compute the amount of overflow traffic offered to the TRUs for each PoC client. An optimization problem can be formulated based on the link offered traffic of the source PoC clients to the BS and of the BS to the destination PoC clients. This can lead to route optimization for the TRUs in a PoC BS. Our objective is to minimize the total traffic lost in the network i.e., from Figure 5-2 we have:

$$\min_{A_k^{i,j}} Z = \sum_{i,k} \hat{a}_{i,k} + \sum_{k,j} \hat{a}_{k,j} \tag{5-7}$$

with the constraints

$$\begin{aligned} &\sum_k A_k^{i,j} = A^{i,j} \\ &A_k^{i,j} \geq 0 \end{aligned} \tag{5-8}$$

where $\hat{a}_{i,k}$ and $\hat{a}_{k,j}$ denote total blocked or overflow traffic at link *(i,k)* and *(k,j)* respectively;

And,

$A_k^{i,j}$ = The amount of traffic offered to TRU *k* from PoC client *i* to PoC client *j*

$a_{i,k}$ = The total traffic offered to link *(i, k)*.

$a_{k,j}$ = The total traffic offered to link *(k, j)*.

The total offered traffic to a link is computed as traffic arrival rate divided by traffic service rate. The Lagrange function to this load sharing optimal problem is

$$L(A,u,v)=\sum_{i,k}\hat{a}_{i,k}+\sum_{k,j}\hat{a}_{k,j}-\sum_{i,j}v^{i,j}\left(\sum_{k}A_k^{i,j}-A^{i,j}\right)-\sum_{i,j,k}u_k^{i,j}A_k^{i,j}. \tag{5-9}$$

where,

u, v = Vectors of Lagrange multipliers

The first order conditions are given by $\partial L / \partial A_k^{i,j} = 0$, which can be expressed as,

$$u_k^{i,j}=-v^{i,j}+\sum_{l,n}\gamma_{l,n}\frac{\partial a_{l,n}}{\partial A_k^{i,j}}+\sum_{n,m}\gamma_{n,m}\frac{\partial a_{n,m}}{\partial A_k^{i,j}}, \tag{5-10}$$

where,

$$\gamma_{i,k}=\frac{\partial \hat{a}_{i,k}}{\partial a_{i,k}} \tag{5-11}$$

Eq. (5-11) is called the marginal overflow of link *(i,k)*. In other words γ is the increment of overflow traffic corresponding to a small increase in the offered traffic. Indices, l, *m* and *n* represent the origin (PoC client l), destination (PoC client *m* [note that in this section, *m* represents a PoC user unlike in section 5.4 where *m* represented number of TRUs as *m* is the number of servers in M/M/m queuing system]) and TRU respectively. The assumption here is that traffic is conserved in the network i.e., the blocking probabilities on the links is so small that it can be neglected. For a session from origin *i* to destination client *j* through TRU *k*, we have:

$$a_{i,k}=\sum_{j}A_k^{i,j} \tag{5-12}$$

$$a_{k,j} = \sum_i A_k^{i,j} \qquad (5\text{-}13)$$

Eq. (5-10) can be reduced to [117, 183],

$$u_k^{i,j} = -v^{i,j} + \gamma_{i,k} + \gamma_{k,j} \qquad (5\text{-}14)$$

Eq. (5-14) has the following interpretation. Consider a particular PoC session flow *(i,j)*. The sum $\gamma_{i,k} + \gamma_{k,j}$ is the total marginal overflow on the path through TRU k for this traffic stream. Because $v^{i,j}$ is independent of k, the optimal load sharing is as follows. If the right hand side of Eq. (5-14) is positive, we should not use the path through TRU k. Conversely, for all paths where there is some flow $\varepsilon_l^k > 0$, share the load to equalize the marginal overflow on all paths. This is obvious from the form of the optimality equation, which becomes $v^{i,j} = \gamma_{i,k} + \gamma_{k,j}$ i.e., the marginal blocking probabilities must be the same on all paths and must be equal to $v^{i,j}$.

However, Eq. (5-14) is not satisfied for overload condition since it does not take into account the path blocking and the lost traffic. In order to address this issue, we use the following case:

$$a_{i,k} = \sum_j A_k^{i,j} \qquad (5\text{-}15)$$

$$a_{k,j} = \sum_i A_k^{i,j} [1 - B_{i,k}] \qquad (5\text{-}16)$$

where, $B_{i,k}$ is the blocking probability of the path *(i,k)*. This kind of path optimization in circuit-switched networks is studied extensively by Kelly (1986, 1988) in [181, 182, 183]. Here, we assume that the traffic offered to the first link in a path is independent of the blocking on the second link, but that the converse is not true; i.e., the traffic offered to the second link has been thinned by an amount proportional to the blocking probability of the first link. This is in fact practical as the incoming traffic to a destination PoC client is dependent on the traffic from PoC BS. In this case the optimality equation becomes (see the Appendix D for details):

$$u_k^{i,j} = -v^{i,j} + \gamma_{i,k} + (1 - B_{i,k})\gamma_{k,j} - \frac{\partial B_{i,k}}{\partial a_{i,k}} \sum_m A_k^{i,m} \gamma_{k,m} \tag{5-17}$$

For the paths of positive flows $\varepsilon_l^k > 0$, we have, the complementary condition:

$$v^{i,j} = \gamma_{i,k} + (1 - B_{i,k})\gamma_{k,j} - \frac{\partial B_{i,k}}{\partial a_{i,k}} \sum_m A_k^{i,m} \gamma_{k,m} \tag{5-18}$$

where, the sum is taken over all destination clients *m.* In this load sharing model at PoC BS, we considered that the blocking at SIP/IP core is negligible and the coupling term is small which is practical. Thus the equal marginal overflow may lead to an optimal path via the PoC BS TRUs. A PoC controlling function will be able to use Eq. (5-17) and Eq. (5-18) to minimize the traffic overflow in busy time. The time complexity here is only *O(m)* provided that offered traffic and blocking values are given.

5.6 Timer Control

Our objective in this section is to control lifetime of the long PoC sessions for instance, session of Figure 5-1(b) for a PoC controller. The following derivation can be used with the assumption that the service GoS, time slot duration and the service rate of time slots are provided [204]. Note that the derivation below addresses one long PoC session only. We define the following notations that will be used throughout this section;

$q(x)$ = The probability that x number of times a PoC session goes through a time slot of a TRU during time interval *T*.

t = Duration of a time slot.

p = The probability of all time slots being occupied at a point of time interval *T*, i.e., the probability that all time slots of a PoC BS are found to be occupied during a chat of a session passing through a time slot; since we want to dimension a PoC service according to the provided GoS, we define p to be equal to the network GoS.

If a session is composed of only single chat, then the session can be serviced by a timeslot and does not need to be constrained. Thus, a session needs to be upper bounded if it contains more than one chat i.e., $x \geq 2$.

The relation between a session chats and the GoS is:

$$p = \sum_{x=2}^{\infty} q(x) \tag{5-19}$$

The assumption here is that the chat arrivals of a PoC session is Poisson process. As traffic is unequally distributed in reality at the TRUs of a BS, it is more correct to calculate the timer with regards to time slot duration [138]. Assuming a session will be active during the whole interval *T* we let, $q(x)$ to have the mean *tT.* Thus, the Poisson distribution *q(x)* is:

$$q(x) = \frac{(tT)^x}{x!} e^{-\left(t+\frac{1}{\mu}\right)T} \tag{5-20}$$

Here, μ represents the mean service rate of each TRU (*40* chats per time slot duration) considering that a TRU has 8 time slot pairs and that each slot can serve 5 chat sessions. Since chat arrivals of a session are continuous, $1/\mu$ is added in order to have the impact of mean service time of a chat in the session chat distribution. A session may go through any of the *N* TRUs in a PoC BS. Therefore,

$$q(x) \leq \frac{(tT)^x}{x!} e^{-\left(t+\frac{1}{\mu}\right)T} \left(\frac{1}{N}\right) \tag{5-21}$$

From Eq. (5-21) and Eq. (5-19) we get,

$$p \leq \sum_{x=2}^{\infty} \frac{(tT)^x}{x!} e^{-\left(t+\frac{1}{\mu}\right)T} \left(\frac{1}{N}\right) \tag{5-22}$$

Using the Taylor series

$$e^{(tT)} = 1 + tT + \frac{(tT)^2}{2!} + \ldots.. \tag{5-23}$$

We find,

$$p \le \frac{e^{tT} - tT - 1}{Ne^{\left(t+\frac{1}{\mu}\right)T}} \tag{5-24}$$

Solving Eq. (5-24) provides a bound for T. Here the computation complexity with 2nd degree approximation is dominated by $1+\left(t+\frac{1}{\mu}\right)T+O\left\{\left(t+\frac{1}{\mu}\right)T\right\}^2$. We use up to the 2nd degree approximation of Taylor expansion to solve Eq. (5-24):

$$p \approx \frac{(tT)^2}{2N+2N\left(t+\frac{1}{\mu}\right)T+N\left(t+\frac{1}{\mu}\right)^2 T^2}, \tag{5-25}$$

i.e.,

$$2Np + 2Np\left(t+\frac{1}{\mu}\right)T + Np\left(t+\frac{1}{\mu}\right)^2 T^2 - (tT)^2 = 0 \tag{5-26}$$

Taking the 1st derivative with respect to T, we get a simple relation:

$$2Np\left(t+\frac{1}{\mu}\right) + 2Np\left(t+\frac{1}{\mu}\right)^2 T - 2t^2 T = 0 \tag{5-27}$$

i.e.,

$$T = \frac{Np\left(t+\frac{1}{\mu}\right)}{t^2 - Np\left(t+\frac{1}{\mu}\right)^2} \tag{5-28}$$

The relationship between a session lifetime and a PoC GoS is provided in Figure 5-8 for $t = 0.02s$, $\frac{1}{\mu} = 0.0005s$. The result shows that a PoC client can have longer session with higher network GoS and higher number of installed TRUs in the network. This is just the reflection of the fact that more expansion of resources will exert better performance.

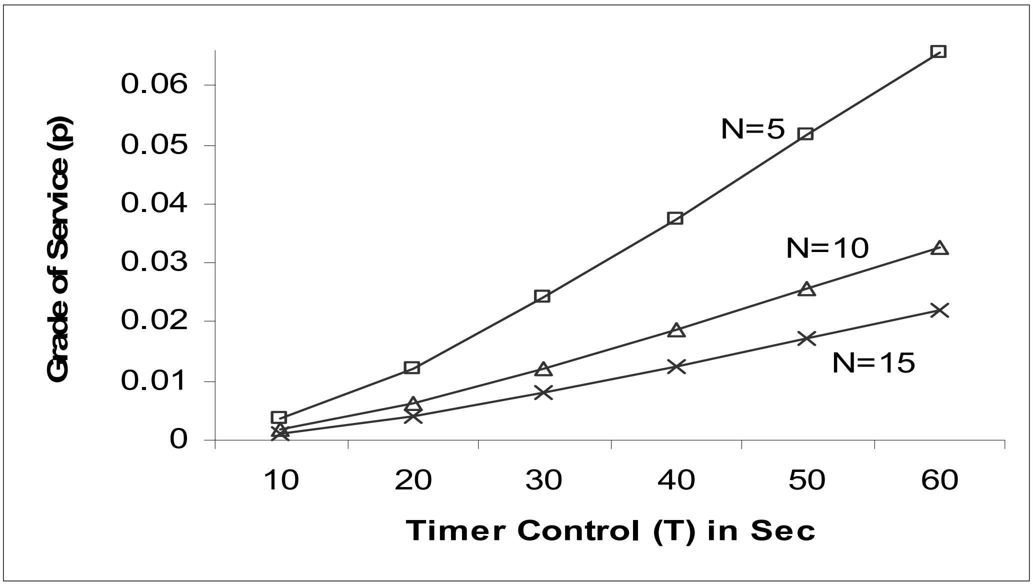


Figure 5-8: Effect of T for multiple installed TRUs

Further the relationship between a session lifetime and a PoC server blocking probability is provided in the following figure (Figure 5-9) for (a) $t = 0.004s$, $1/\mu = 0.0001s$ (b) $t = 0.001s$, $1/\mu = 0.000025s$ and (c) $t = 0.0005s$, $1/\mu = 0.0000125s$. The total number of installed TRUs(N) were kept 5. The mean service rate was defined for 1 TRU with consideration that a TRU can serve 40 simultaneous chats on average per time slot (duration) with slot duration 0.004s, 0.001s and 0.0005s. We find that time slots with lower slot duration performs better in terms of blocking PoC sessions. This is because with smaller slot duration the service time will decrease i.e., chat service rate will increase and as a result will reduce the blocking probability.

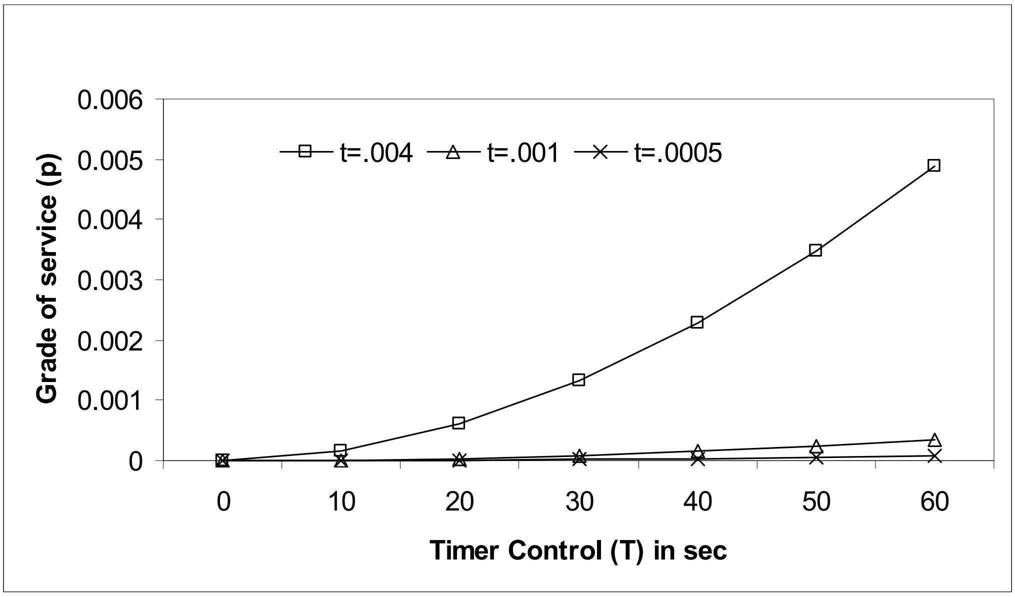


Figure 5-9: Effect of timer for various length slots

5.7 Optimization of Simultaneous Sessions

Our objective in this section is to control the number of simultaneous sessions for a PoC client during busy time. Since, the Northstream report [138] suggests that cost analysis based on time slots of PoC servers produce equal outcomes as that of TRUs, we consider our next analysis based on number of time slots. The two-state Markov chain model has been extensively used for the voice traffic. Gilbert's model (1960, [185]) and recent works in [186-190] have shown that a simple two-state Markov chain can measure packet loss over the Internet efficiently. We use similar approach to compute the number of the optimal sessions for a PoC client. The analysis presented here is to limit number of simultaneous long sporadic/pre-established sessions (Type 2) for a PoC client during busy hour. Thus the notations λ, μ, a denote arrival rate, mean service rate and traffic intensity respectively of Type 2 sessions in this section. The other notations that will be used throughout this section are as follows:

N_T = The total number of time slots of a PoC network,

N_c = The number of PoC clients being served by a PoC BS/the whole network.

The two state natures of Figure 5-10 and Figure 5-11 can capture the bursty nature of the number of simultaneous sessions in busy hour. The former represents the states of the BS where as the later represents the states of a PoC client. The model in Figure 5-10 has two states: Blocking or busy and Not busy. H_1 and H_2 are the state transition probabilities. The PoC BS goes to Blocking state *0***,** when all channels/time slots are busy at a random point of time that can be computed from Erlang's loss formula. In this state, number of session arrival in the BS is greater than $5N_T$, assuming that a time slot serves 5 PoC sessions at the same time on the average.

$$H_2 = \frac{\frac{a^{N_T}}{N_T!}}{\sum_{d=0}^{N_T} \frac{a^d}{d!}} \qquad (5\text{-}29)$$

H_2 is the transition probability that causes the BS enter into Blocking state i.e., by definition H_2 is the given GoS. It goes to Not busy state *1***,** when there is at least one time slot available that can be computed from the Binomial distribution. Any new session will be blocked when the server is in state *0***.** A successful session set up only depends on the current state. Because of the throttled nature of the PoC sessions, a session changes between idle (inactive) and busy (active), the offered traffic per session is

$$\alpha = \frac{T_{busy}}{T_{idle} + T_{busy}} = \frac{1/\mu}{1/\lambda + 1/\mu} = \frac{a}{1+a} \qquad (5\text{-}30)$$

Then, for non busy state,

$$H_1 = \sum_{d=0}^{N_T - 1} \binom{N_T}{d} \alpha^d (1-\alpha)^{N_T - d} \qquad (5\text{-}31)$$

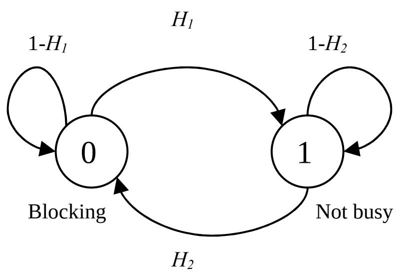


Figure 5-10: Markov model for the PoC BS states

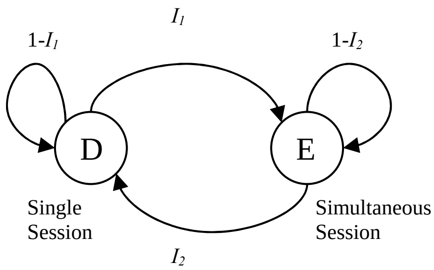


Figure 5-11: Session states of a PoC Client

The session Blocking is equal to the state probability $P(0)$. Similarly the probability of successful session set up is equal to the state probability $P(1)$. The transition between two states occurs at each session set up/termination. Thus in steady state:

$$P(0) + P(1) = 1 \tag{5-32}$$

The state transition matrix is given by

$$P_H = \begin{bmatrix} 1-H_1 & H_1 \\ H_2 & 1-H_2 \end{bmatrix} \tag{5-33}$$

Figure 5-11 illustrates the nature of a session initiation situation of a PoC client. State D represents a client initiating one session and state E represents multiple session initiation. I_1 and I_2 are the transition probabilities. The probability that a PoC client initiates a session is the mean arrival rate of all PoC clients i.e.,

$$I_2 = \frac{\lambda}{N_c} = \frac{\sum_{i=1}^{N_c} \lambda_i}{N_c} \qquad (5\text{-}34)$$

The probability of simultaneous session initiation of a PoC client during a known period *T* can be determined by one less the probability of one session initiation of a PoC client. Since, we assume that the session initiations are Poisson streams we have,

$$\begin{aligned} I_1 &= 1 - \Pr[one\,session \mid T = t_s] \\ &= 1 - I_2 t_s e^{-I_2 t_s} \\ &= 1 - \frac{\lambda}{N_c} t_s e^{-\left(\frac{\lambda}{N_c} t_s\right)} \end{aligned} \qquad (5\text{-}35)$$

where, t_s is the session lifetime of a PoC client. The probability of a successful session set up of a particular client is equal to the state probability *P(D)*. Similarly, the probability that a client is successful in establishing more than or equal to two simultaneous sessions is equal to the state probability *P(E).* In steady state,

$$P(D) + P(E) = 1 \qquad (5\text{-}36)$$

The state transition matrix is:

$$P_I = \begin{bmatrix} 1 - I_1 & I_1 \\ I_2 & 1 - I_2 \end{bmatrix} \qquad (5\text{-}37)$$

Since the PoC system going to busy state depends on total number of sessions, we concatenate two models as shown in Figure 5-12.

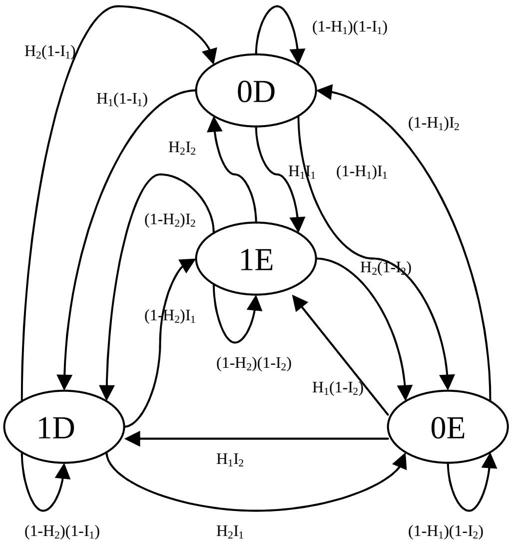


Figure 5-12: Four state Markov chain for session set up

In this model, state *(0D)* and *(0E)* represent session blocking whereas, state *(1D)* and *(1E)* represent successful session set ups. Again success or failure of session set up depends on the current state. At steady state:

$$P(0D)+P(0E)+P(1D)+P(1E)=1 \tag{5-38}$$

The state transition probability matrix is given as:

$$P_{HI}=\begin{bmatrix} (1-H_1)(1-I_1) & H_1(1-I_1) & I_1(1-H_1) & H_1I_1 \\ H_2(1-I_1) & (1-H_2)(1-I_1) & H_2I_1 & I_1(1-H_2) \\ I_2(1-H_1) & I_2H_1 & (1-I_2)(1-H_1) & H_1(1-I_2) \\ H_2I_2 & I_2(1-H_2) & H_2(1-I_2) & (1-H_2)(1-I_2) \end{bmatrix} \tag{5-39}$$

5.7.1 Estimating steady state probabilities

Let:

$$
\begin{aligned}
e_1 &= (1-H_1)(1-I_1) \\
e_2 &= H_1(1-I_1) \\
e_3 &= I_1(1-H_1) \\
e_4 &= H_1 I_1 \\
f_1 &= H_2(1-I_1) \\
f_2 &= (1-H_2)(1-I_1) \\
f_3 &= H_2 I_1 \\
f_4 &= I_1(1-H_2) \\
g_1 &= I_2(1-H_1) \\
g_2 &= I_2 H_1 \\
g_3 &= (1-I_2)(1-H_1) \\
g_4 &= H_1(1-I_2) \\
h_1 &= H_2 I_2 \\
h_2 &= I_2(1-H_2) \\
h_3 &= H_2(1-I_2) \\
h_4 &= (1-H_2)(1-I_2)
\end{aligned} \tag{5-40}
$$

Therefore, the transition probability matrix becomes:

$$
P_{HI} = \begin{bmatrix} e_1 & e_2 & e_3 & e_4 \\ f_1 & f_2 & f_3 & f_4 \\ g_1 & g_2 & g_3 & g_4 \\ h_1 & h_2 & h_3 & h_4 \end{bmatrix} \tag{5-41}
$$

In steady state we have the following vectors:

$$
P(0D) = e_1 P(0D) + f_1 P(1D) + g_1 P(0E) + h_1 P(1E) \tag{5-42}
$$

$$
P(1D) = e_2 P(0D) + f_2 P(1D) + g_2 P(0E) + h_2 P(1E) \tag{5-43}
$$

$$
P(0E) = e_3 P(0D) + f_3 P(1D) + g_3 P(0E) + h_3 P(1E) \tag{5-44}
$$

$$
P(1E) = e_4 P(0D) + f_4 P(1D) + g_4 P(0E) + h_4 P(1E) \tag{5-45}
$$

$$
P(0D) + P(1D) + P(0E) + P(1E) = 1 \tag{5-46}
$$

Using the Gaussian elimination we have:

$$
P(0D) = \frac{f_1 P(1D) + g_1 P(0E) + h_1 P(1E)}{1-e_1} \tag{5-47}
$$

$$
P(1D) = \frac{\{g_2(1-e_1) + e_2 g_1\} P(0E) + \{h_2(1-e_1) + e_2 h_1\} P(1E)}{(1-f_2)(1-e_1) - e_2 f_1} \tag{5-48}
$$

$$P(0E)=\frac{\begin{bmatrix} h_3(1-e_1)\{(1-e_1)(1-f_2)-e_2f_1\}+e_3h_1(1-f_2)(1-e_1)-e_2f_1\} \\ +(1-e_1)\{h_2(1-e_1)f_3+e_2h_1f_3\}+e_3f_1\{h_2(1-e_1)+e_2h_1\} \end{bmatrix}}{\begin{bmatrix} (1-e_1)\{(1-e_1)(1-f_2)-e_2f_1\}(1-g_3)-(1-e_1)\{f_3g_2(1-e_1)+e_2g_1f_3\} \\ -e_3g_1\{(1-f_2)(1-e_1)-e_2f_1\}-e_3f_1\{g_2(1-e_1)+e_2g_1\} \end{bmatrix}}P(1E) \tag{5-49}$$

$$P(1E)=\frac{(1-e_1)\{(1-f_2)(1-e_1)-e_2f_1\}\begin{bmatrix} (1-e_1)\{(1-e_1)(1-f_2)-e_2f_1\}(1-g_3) \\ -(1-e_1)\{f_3g_2(1-e_1)+e_2g_1f_3\}-e_3g_1 \\ *\{(1-f_2)(1-e_1)-e_2f_1\}-e_3f_1\{g_2(1-e_1)+e_2g_1\} \end{bmatrix}}{\begin{bmatrix} \begin{bmatrix} (1-e_1)\{(1-e_1)(1-f_2)-e_2f_1\}(1-g_3)-(1-e_1)\{f_3g_2(1-e_1)+e_2g_1f_3\} \\ -e_3g_1\{(1-f_2)(1-e_1)-e_2f_1\}-e_3f_1\{g_2(1-e_1)+e_2g_1\} \end{bmatrix} \\ *\left[\{f_1+(1-e_1)\}\{h_2(1-e_1)+e_2h_1\}+\{(1-f_2)(1-e_1)-e_2f_1\}\{h_1+(1-e_1)\}\right] \\ +\begin{bmatrix} h_3(1-e_1)\{(1-e_1)(1-f_2)-e_2f_1\}+e_3h_1\{(1-f_2)(1-e_1)-e_2f_1\} \\ +(1-e_1)\{h_2(1-e_1)f_3+e_2h_1f_3\}+e_3f_1\{h_2(1-e_1)+e_2h_1\} \end{bmatrix} \\ *\left[\{f_1+(1-e_1)\}\{g_2(1-e_1)+e_2g_1\}+\{(1-f_2)(1-e_1)-e_2f_1\}\{g_1+(1-e_1)\}\right] \end{bmatrix}} \tag{5-50}$$

5.7.2 Optimal values

The matrix multiplication takes *O(16)* time only. Let, the total probability of simultaneous session being successful be π.

$$\begin{aligned} \pi &= P(1E) \\ &= 1-[P(0D)+P(1D)+P(0E)] \end{aligned} \tag{5-51}$$

where *P(0D)* is the probability that a single session set up is blocked; *P(1D)* is the probability that a single session can be established; and *P(0E)* is the probability that simultaneous sessions is blocked.

Therefore, the mean random variable, $\bar{n}$, i.e., the number of simultaneous session for a PoC client can be obtained by:

$$\lfloor \bar{n} \rfloor = \left\lfloor \pi\left(\frac{\lambda}{N_c}t_s\right)\right\rfloor \tag{5-52}$$

A list of values for $\overline{n}$ has been furnished in Table 5-3, Table 5-4, Table 5-5, Table 5-6, and in Table 5-7 for variable parameters:

Table 5-3: Number of allowable simultaneous sessions for a PoC client

$\alpha = \frac{0.99}{1+0.99}$, $N_T = \{40 \to 120\}$ $N_c = 500$, $H_2 = 0.02$, $T = 40s$								
λ	H_1	I_1	I_2	P(1E)	P(0E)	P(1D)	P(0D)	$\overline{n}$
50	0.99999	0.92673	0.1	0.88351	0.01908	0.09548	0.00190	3.53406
75	0.99999	0.98512	0.15	0.84866	0.01918	0.12955	0.00258	5.09201
100	0.99999	0.99731	0.2	0.81374	0.01921	0.16377	0.00326	6.50994
125	0.99999	0.99954	0.25	0.78070	0.01922	0.19616	0.00390	7.80703
150	0.99999	0.99992	0.3	0.74999	0.01922	0.22628	0.00450	8.99991
175	0.99999	0.99998	0.35	0.72151	0.01922	0.25421	0.00505	10.10117
200	0.99999	0.99999	0.4	0.69505	0.01922	0.28015	0.00556	11.12094
225	0.99999	0.99999	0.45	0.67042	0.01922	0.30431	0.00603	12.06771
250	0.99999	0.99999	0.5	0.64743	0.01922	0.32685	0.00647	12.94879
275	0.99999	0.99999	0.55	0.62593	0.01922	0.34795	0.00688	13.77054
300	0.99999	0.99999	0.6	0.60577	0.01922	0.36773	0.00726	14.53853
325	0.99999	0.99999	0.65	0.58683	0.01922	0.38631	0.00762	15.25764
350	0.99999	0.99999	0.7	0.56900	0.01922	0.40379	0.00796	15.93219
375	0.99999	0.99999	0.75	0.55219	0.01922	0.42028	0.00828	16.56599
400	0.99999	0.99999	0.8	0.53632	0.01922	0.43586	0.00858	17.16244

The corresponding graphical representation is provided in Figure 5-13.

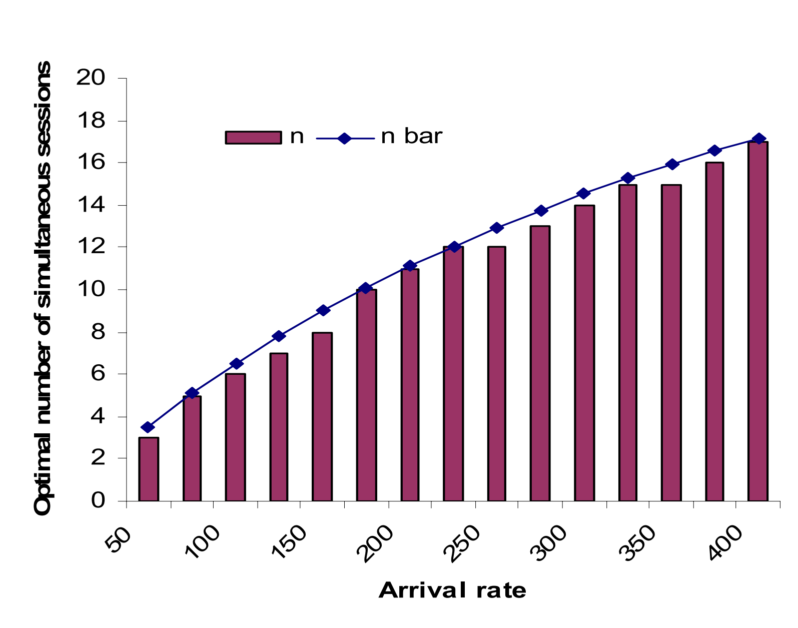


Figure 5-13: Number of allowable simultaneous sessions

Table 5-4: Number of allowable simultaneous sessions for a PoC client

$\alpha = \frac{0.99}{1+0.99}$, $N_T = \{40 \rightarrow 120\}$ $N_c = 500$, $H_2 = 0.02$, $\lambda = 50/s$								
T(sec)	H_1	I_1	I_2	*P(1E)*	*P(0E)*	*P(1D)*	*P(0D)*	$\overline{n}$
20	0.99999	0.72932	0.1	0.86082	0.01859	0.11821	0.00236	1.72164
30	0.99999	0.85063	0.1	0.87588	0.01892	0.10313	0.00205	2.62765
40	0.99999	0.92673	0.1	0.88351	0.01908	0.09548	0.00190	3.53406
50	0.99999	0.96631	0.1	0.88705	0.01916	0.09194	0.00183	4.43527
60	0.99999	0.98512	0.1	0.88864	0.01919	0.09035	0.00180	5.33188
70	0.99999	0.99361	0.1	0.88934	0.01921	0.08964	0.00179	6.22543
80	0.99999	0.99731	0.1	0.88964	0.01921	0.08934	0.00178	7.11719
90	0.99999	0.99888	0.1	0.88977	0.01922	0.08921	0.00178	8.00799

The corresponding graphical representation is provided in Figure 5-14.

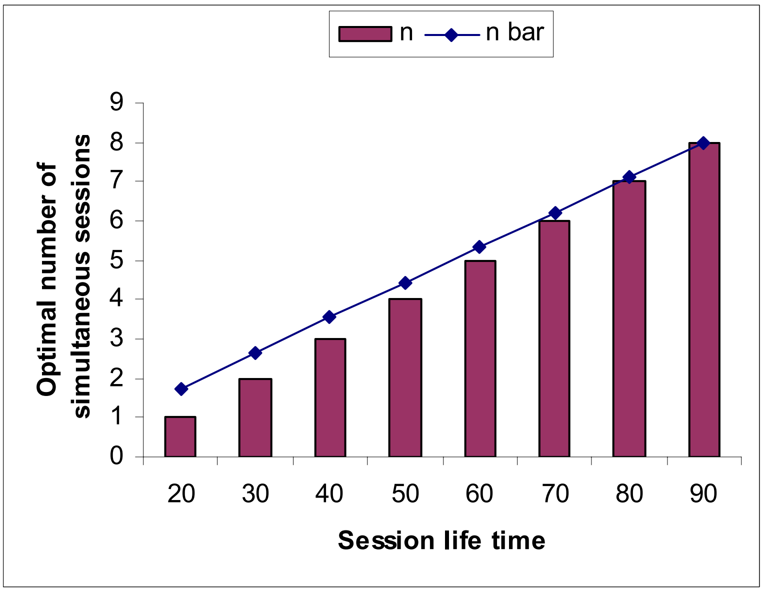


Figure 5-14: Number of allowable simultaneous sessions

Table 5-5: Number of allowable simultaneous sessions for a PoC client

$\alpha = \frac{0.99}{1+0.99}$, $N_T = \{40 \rightarrow 48\}$, $N_c = 500$, $T = 40s$, $\lambda = 50/s$								
H_2	H_1	I_1	I_2	$P(1E)$	$P(0E)$	$P(1D)$	$P(0D)$	$\overline{n}$
0.01	0.99999	0.92673	0.1	0.89287	0.00973	0.09643	.00096	3.57148
0.02	0.99999	0.92673	0.1	0.88351	0.01908	0.09548	0.00190	3.53406
0.03	0.99999	0.92673	0.1	0.87452	0.02807	0.09456	0.00283	3.49810
0.04	0.99999	0.92673	0.1	0.86588	0.03672	0.09365	0.00373	3.46352
0.05	0.99999	0.92673	0.1	0.85757	0.04503	0.09276	0.00462	3.43028
0.06	0.99999	0.92673	0.1	0.84957	0.05302	0.09189	0.00550	3.39831
0.07	0.99999	0.92673	0.1	0.84189	0.06071	0.09103	0.00636	3.36756
0.08	0.99999	0.92673	0.1	0.83449	0.06810	0.09019	0.00720	3.33799
0.09	0.99999	0.92673	0.1	0.82738	0.07521	0.08936	0.00803	3.30955

The optimal number of simultaneous sessions is 3 for all values of H_2 in the Table 5-5.

Table 5-6: Number of allowable simultaneous sessions for a PoC client

$\alpha = \frac{0.99}{1+0.99}$, $N_T = 40$, $H_2 = 0.02$, $T = 40s$, $\lambda = 50/s$								
N_c	H_1	I_1	I_2	*P(1E)*	*P(0E)*	*P(1D)*	*P(0D)*	$\overline{n}$
100	0.99999	0.99999	0.5	0.64743	0.01922	0.32685	0.00647	12.94879
200	0.99999	0.99954	0.25	0.78070	0.01922	0.19616	0.00390	7.80703
300	0.99999	0.99151	0.17	0.83689	0.01920	0.14109	0.00281	5.57930
400	0.99999	0.96631	0.13	0.86630	0.01914	0.11229	0.00224	4.33154
500	0.99999	0.92673	0.1	0.88351	0.01908	0.09548	0.00190	3.53406
600	0.99999	0.88108	0.08	0.89456	0.01902	0.08471	0.00169	2.98188
700	0.99999	0.83590	0.07	0.90230	0.01897	0.07718	0.00154	2.57800
800	0.99999	0.79478	0.06	0.90815	0.01893	0.07147	0.00142	2.27039

The corresponding graphical representation is provided in Figure 5-15.

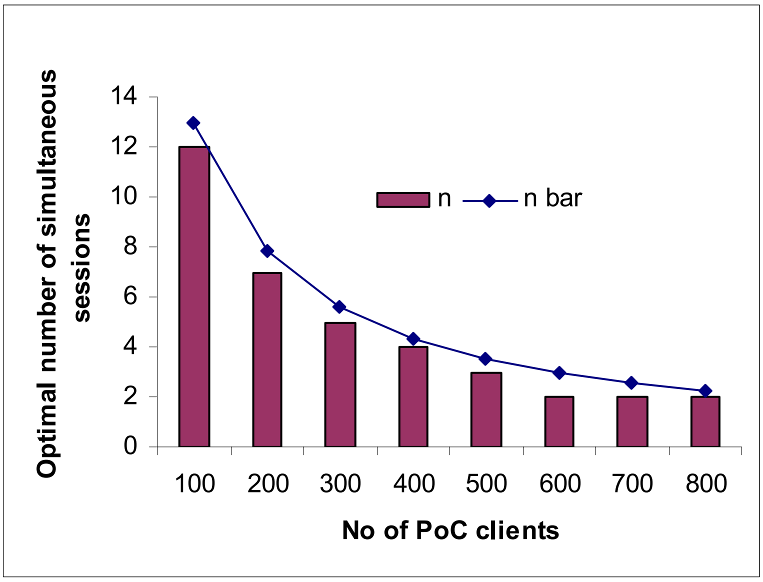


Figure 5-15: Number of allowable simultaneous sessions

Table 5-7: Number of allowable simultaneous sessions for a PoC client

$N_c = 500,\quad N_T = 40,\quad H_2 = 0.02,\quad T = 40s,\quad \lambda = 50/s$								
a	H_1	I_1	I_2	$P(1E)$	$P(0E)$	$P(1D)$	$P(0D)$	$\overline{n}$
2	0.99999	0.92673	0.1	0.88351	0.01908	0.09548	0.00190	3.53406
10	0.97790	0.92673	0.1	0.88309	0.01950	0.09544	0.00195	3.53239
20	0.85795	0.92673	0.1	0.88045	0.02214	0.09515	0.00223	3.52183

The above results are dependent on allocated resources in a cell. As mentioned before that we assume the TRU/Time slots can be expanded in a cell. We have shown the effects on simultaneous sessions for a cell serving 100-800 IMS PoC users, for varying GoS from 0.01-0.09, for increasing arrival rate from 50-400 session-chats/sec and for increasing session timer from 20-90s etc. The literature review [138] suggests that on average each user may start 3 Push-to-Talk sessions in busy hour having 40 seconds session timer with 2% radio network GoS for GSM calls, 400,000 users for

3000 base station sites and with 60% of category 2 sites. Our results can be applied according to the number of PoC users being served and available resources. These results may also be used for computation of additional measurement of resource expansion.

5.8 Summary

In this chapter of the thesis, we derived and analysed several optimal characteristics to dimension a PoC service. The performance for PoC is highly dependent on tuning the service from an end-to-end perspective. We have shown the effects of providing controlled access to two different types of sessions, optimized load sharing expressions for a PoC controller as a decision criterion, a simple relation to control the session timer and finally an expression to compute the maximum number of allowable simultaneous sessions for each PoC client during busy hour. The analyses suggest that

- Careful modelling can reduce load at BS and secure low GoS.
- Access should be restricted to the sessions that require more message flows in the network.
- Path Optimization will reduce traffic overflow at the BS.
- Optimizing session life time can achieve the desired GoS.
- Optimizing number of simultaneous sessions for each PoC user can achieve desired GoS.
- Optimal resource allocation depends on optimal usage of resources.

A service provider can benefit from the analyses performed in this chapter.

Chapter 6 Efficient IMS Session Set Up in Mobile Environment

6.1 Introduction

As mentioned earlier in the thesis that though the literature reviews on mobility management is handful, the session set up in mobile environment received very low attention. The recent QoS support work over SIP mobility management can be found in [163-169]. Wang and Abu-Rgheft (2006) presented a cost efficient mobility management technique by integrating MIP with SIP [168]. Similar work can be identified in [169]. Molina *et al* (2006) implemented a prototype which solves most of the scalability problems without requiring major SIP extensions [166]. Their approach was built on QoS support to SIP calls carrying them in DiffServ IP trunks and QA extensions to SIP servers. Some applications of SIP message prioritization have been shown in [164]. The essential processing requirements of SIP elements are studied in [163]. A dual stack scheme to reduce the SIP tunnelling overhead and transmission delay in 3G IMS has been demonstrated by Huang *et al* (2006) in [167]. However, all of these mobility management techniques are directed towards post session set up data transfer. Our work is on the mobility management while a session is being set up. The different SIP hand-off-delay analysis has been performed in [71] by Banerjee *et al* (2003). We use the basic delay of M/M/1 machines and some fixed delays (to be discussed later in this chapter) to compute end to end delay to set up sessions.

Our objective in this chapter is to identify the best method for session set up in mobile environment for the IMS terminals via performance analysis. A source node or a destination node or both may be mobile while they participate in a session establishment process. Usually if a terminal is mobile (MN) in IMS session set up, the SIP redirect server which is an application server, informs the originating terminal (CN) to initiate a new INVITE message destined to the new location of the mobile terminal (MN). Also, there are some other issues for instance channel handoffs while a mobile node changes location. In this chapter of the thesis, we study and compare the possible session set up scenarios and analyse the pros and cons of them in IMS framework.

6.2 Scenario Description

Every mobile node must register with the home network in IMS. Re-registration takes place once the time out occurs. If we recap a session set up in IMS, the SIP INVITE request is sent from the UE (user equipment) to S-CSCF#1 (serving call session control function) by the procedures of the originating flow to initiate a session between two nodes via its Proxy P-CSCF#1. This message may contain the initial media description in the SDP (session description protocol). S-CSCF#1 performs an analysis and passes the request to I-CSCF#1 (Interrogating CSCF) and so on. Thus the intermediate nodes analyse and forward the request to the next node till it reaches the destination node. The detail of IMS SIP session set up procedures with MIPv6 can be found in [24].

The whole procedure of session set up may be divided into four stages as shown in Figure 6-1 (This figure provides a general overview whereas the actual set up flows were presented in section 2.3 in chapter 2 of this thesis. Protocol specific session set up in IMS is provided later in the simulation section of this chapter). Stage one includes sending of INVITE message from source to destination and getting a response from the destination back to the source. Stage two, three and four can be described as same manner for Response, Reservation and Acknowledge messages respectively.

6.2.1 Reasons of a Session failure

A session set up may fail anytime due to the different processing complexity. The channel handoff in a network may fail where no bandwidth/channel is available. The destination node may send a BUSY message in response to the INVITE request. The message may also be corrupted or lost in the intermediate nodes which raise the possibility for a session to fail at any stage during the period of establishment. Thus, we identify two main reasons for a session failure: (a) handoff failure and (b) node failure.

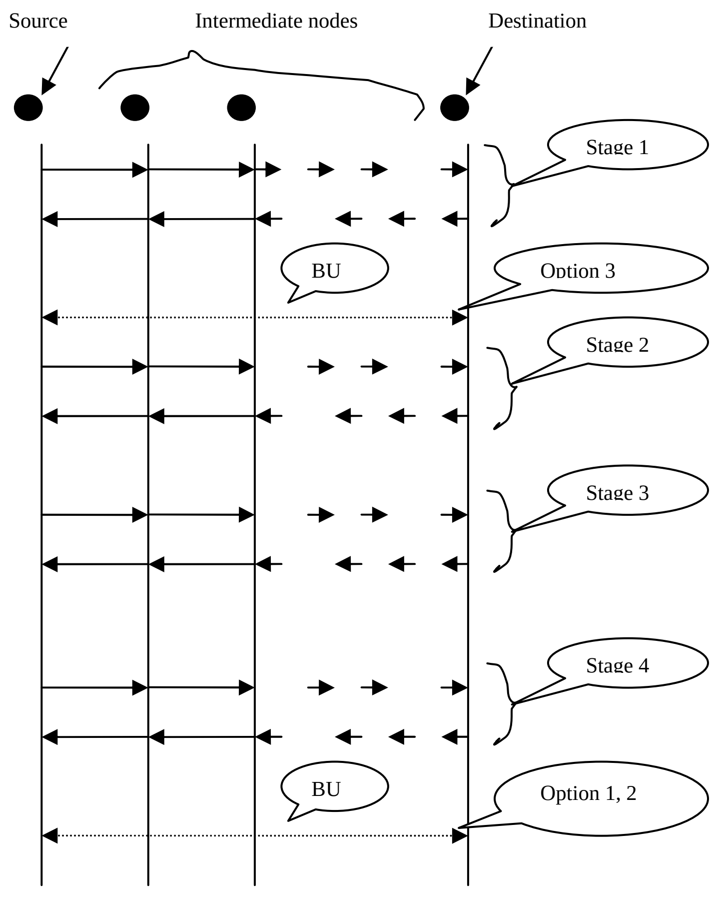


Figure 6-1: Three options for IMS session set up

6.2.2 The Three Session Set up Scenarios

1. The basic scenario (Option 1) is that the MN (mobile node or destination) receives packets from the CN (corresponding node or source) tunnelled though the HA (Home Agent), and initiates the route optimization procedure (see Figure 6-2) after BU (Binding Update) is sent. This implies that traffic will be routed through the HA before being routed directly to the MN, even if for a limited amount of time. This can have implications on quality of service (QoS), since QoS is initially established only for the route from the MN to the HA and to the CN, whereas QoS for the optimized route is not established.

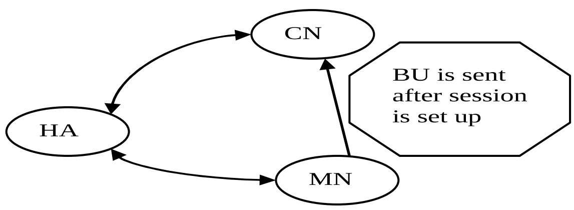


Figure 6-2: Mobility in IMS by SIP

2. A second scenario (Option 2) introduces an optimization where the MN sends a BU to the CN immediately after setting up the SIP call, before any traffic is received from the CN [81]. This requires slight modifications to the implementation of the MN, but benefits from route optimization since the beginning of the communication. This option in [81] is proposed by Faccin *et al* (2004) without much detail.

The main difference between Option 1 and Option 2 lies in the data transfer. MN may start sending the data via HA immediately after the session is set up, before the BU has been sent in Option 1. MN waits till the BU has been sent before it starts to send any data in Option 2.

3. We propose an additional optimization (Option 3) by sending the BU message in parallel while the SIP session is still being set up in IMS [180]. There is handful number of location prediction work available today that can be utilized if necessary to send the BU in advance. For example, after the first round trip (which includes INVITE message reaching the destination and coming back to the source as 183 session progress) of messages of a session set up, the source may initiate a BU for the destination and try to ensure QoS. Alternatively, the BU message could be initiated after the second round trip of a session set up progress (which includes PRACK message reaching the destination and coming back to the source as 200OK session progress). In this approach, the CN can immediately start to send data once the session has been set up using the optimized route. However, the overhead would be high if a session fails to set up for various reasons as mentioned above.

Note that Option 1 and Option 2 are applicable only for a successful session set up while Option 3 is applicable to both successful and unsuccessful session set ups. In Option 1 and 2, the BU is sent only after the session is set up while the BU is sent in parallel in Option 3. In following sections, we derive a model to compare and evaluate the performance of the three abovementioned options.

6.3 Modelling

We define the costs for the three possible options in terms of delays in this section. The following nomenclature is used throughout this chapter:

$\lambda_{i,}$, $i=1,2,\ldots n$: SIP message arrival rate at node i,

$\mu_{i,}$ $i=1,2,\ldots n$: serving rate for each SIP message at node i,

$\rho_{i,}$ $i=1,2,\ldots n$: load at node i for SIP messages,

where $\rho_i = \lambda_i / \mu_i$, for $\lambda_i < \mu_{i,}$

D_i: the queuing delay at node i,

p_n: the probability of a message reaching next node,

Q: the probability that a session will fail to establish,

N: number of messages sent from source node (CN) to destination (MN) to send data,

n: number of intermediate nodes involved for sending BU between the source and destination node,

r: number of total nodes involved in a session set up from stage two to four, (n is the nodes involved in the optimized route whereas r is the number of nodes involved in the initial route multiplied by the number of message flows required to set up a session after a specific stage in the session set up),

x: number of extra nodes (including HA) a message has to go through in

Option 1 before the BU has been sent, (extra nodes represent number of additional nodes compared to the nodes involved in the optimized route),

C_1: total cost for Option 1 in SIP set up,

C_2: total cost for Option 2 in SIP set up,

C_3: total cost for Option 3 in SIP set up,

C_{3_S}: total cost for a successful SIP set up in first trial for Option 3,

C_{3_F}: total cost for a successful SIP set up in second trial for Option 3,

y: cost of sending data,

U: cost for sending Binding Update message,

p_d : network-wide session dropping probability,

p_f : handoff failure probability during session set up,

p_h : handoff probability of a session set up,

H: number of possible handoffs during the life of a session set up,

t_μ : duration of a particular session set up

t_h : cell residency time of a particular mobile node which is involved in a session set up

$1/\mu$: mean session set up duration

$1/h$: mean cell residency time during session set up

We define D_i as the queuing delay for a node *i* that behaves as an M/M1 machine.

We define, $y = \frac{V_w}{\eta}$ where η is the average throughput of the channel and V_w represents the data sent in bytes.

The cost of sending BU is measured as the sum of delay at each node that is involved to send the BU between source node and destination node,

$$U = \sum_{i=1}^{n} D_i \tag{6-1}$$

Cost of Option 1 includes (a) the extra cost to send packets from source (CN) to destination (MN) through HA before the binding update (BU) has been sent, and (b) sending the BU message from MN to CN (c) less the cost of sending data that has already been sent before the BU has been received by the destination IMS terminal.

$$C_1 = N\sum_{i=1}^{x} D_i + U - y \tag{6-2}$$

In practice, there will be other delays while calculating Eq. (6-1) and Eq. (6-2) for instance, Internet transmission delay D_I, end to end frame propagation delay, D_p (distance between nodes over speed of light) in the radio link, and queuing delay $D_{NON-SIP_i}$ at hop *i* in order to process messages other than SIP messages. Since it is difficult to standardize the heterogeneous transmission paths, Internet transmission delay can be kept constant. The queuing delay for messages other than SIP is computed in [71, 174] as: $\frac{\frac{1}{\mu_i}\left(1-\rho_{NON-SIP_i}-\rho_i\right)+R}{\left(1-\rho_{NON-SIP_i}\right)+\left(1-\rho_{NON-SIP_i}-\rho_i\right)}$, where $R = \frac{\lambda_{NON-SIP_i} \overline{X}^2_{NON-SIP_i} + \lambda_i \overline{X}^2_i}{2}$; $\overline{X}^2_{NON-SIP_i}$ $\overline{X}^2_i$ are the second moments of $\mu_{NON-SIP_i}$ and μ_i respectively; $\lambda_{NON-SIP_i}$, $\mu_{NON-SIP_i}$, $\rho_{NON-SIP_i}$ are the arrival rate, service rate and load for messages other than SIP at hop i. R is computed from the expressions $\overline{X}^2_{NON\_SIP_i} = E\left[X^2_{NON\_SIP_i}\right] = \sigma^2_{NON\_SIP_i} + \left(E\left[X_{NON\_SIP_i}\right]\right)^2$ and $\overline{X}^2_i = E\left[X^2_i\right] = \sigma^2_i + \left(E\left[X_i\right]\right)^2$ where $\sigma^2_{NON-SIP\,i}$, σ^2_i are the respective variance. The expression of $D_{NON-SIP_i}$ is obtained by using the result of a non-pre-emptive priority based M/G/1 queue. However, for simplicity our simulation is centred on SIP messages only.

Cost of Option 2 is (a) the cost of waiting till the binding update message has been sent (It might have been possible to send some data during the elapsed time of BU to be completed.) and (b) the cost of sending BU message from MN to CN.

$$C_2 = y + U \tag{6-3}$$

Cost of Option 3 in a successful session set up (C_{3_S}) is the cost of sending BU message only from MN to CN. However, if the session fails after stage one the source node will initiate the INVITE message again to try to set up the session. The BU message will be sent again in that case. Thus the total cost for Option 3 is the sum of both successful and unsuccessful case. We assume that the session is always successful in the second trial if it fails in the first trial after stage one. If the session fails in second trial, the IMS source does not initiate a session with the same destination immediately which is practical. Since, the source node will re-initiate INVITE message after first failure and since the BU will be sent again in Option 3 for the first failure, Cost of Option 3 in an unsuccessful session set up is:

$$C_{3\_F} = 2C_{3\_S} \tag{6-4}$$

As mentioned before, the probability of a session failure depends on (a) node failure and (b) session dropping probability, i.e. a session may fail to set up because of a node failure or handoff failure in the network. Although session dropping probability is more meaningful for mobile users and service providers, calculating the handoff failure probability is more convenient. The session dropping probability in our context is (see [156] for more details):

$$p_d = \sum_{H=0}^{\infty} (p_h)^H (1-p_f)^{H-1} p_f = \frac{p_h p_f}{1-p_h(1-p_f)} \tag{6-5}$$

where,

$$\begin{aligned} p_h &= \Pr(t_\mu > t_h) \\ &= \int_{t=0}^{\infty} \Pr(t_\mu > t_h)\Pr(t_h = t)dt \\ &= \int_{t=0}^{\infty} he^{-\mu t} e^{-ht} dt \\ &= \frac{h}{\mu + h} \end{aligned} \tag{6-6}$$

Therefore,

$$p_f = \frac{p_d}{1-p_d}\left(\frac{\mu}{h}\right) \tag{6-7}$$

It means that, for a given p_d, the equivalent p_f can be easily computed based on the above equation. The detail of the handoff probability under cell residency distributions can be located in [156-158].

We may define the cell residency time, t_h based on mobile users tracking within a cell as derived in [159-161]. For modelling purposes it is assumed that cells have hexagonal form with side *a*, and IMS subscribers are uniformly distributed within a cell. In our model, we approximate hexagonal cell with a circle, with radius *R* (see Figure 6-3):

$$R^2\pi = \frac{3\sqrt{3}}{2}a^2 \tag{6-8}$$

where, *a* is the size of the hexagonal side.

The position of the user at the initiation of a cell is defined with radius *d,* where *d* is the distance from the centre of the cell (it is the position of the base station in case of omni cell).

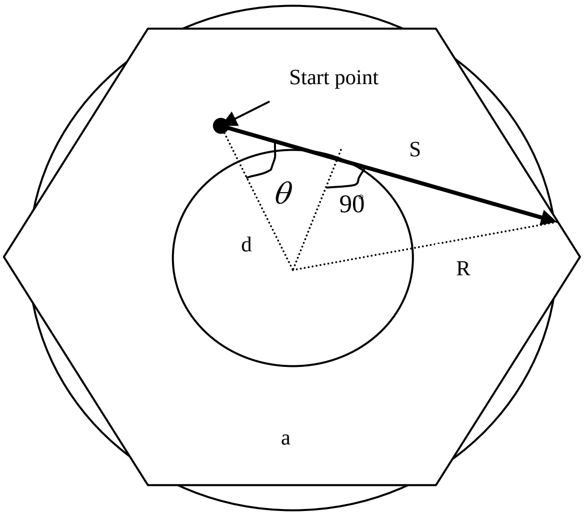


Figure 6-3: Movement of an IMS terminal

The probability density function for the IMS terminal density in a cell is:

$$f_d(d) = \begin{cases} 2d/R^2, & 0 \le d \le R \\ 0, & d > R \end{cases} \qquad (6\text{-}9)$$

The direction of a mobile node within a cell is defined with angle $\theta,$ uniformly distributed. So, probability distribution function for the direction of user movement after session initiation is:

$$f_\theta(\theta) = 1/2\pi, \quad 0 \le \theta < 2\pi. \qquad (6\text{-}10)$$

We assume that the direction and the speed of the mobile nodes remain constant within a cell and these are allowed to change at handover to another cell. The initial velocity of the mobile stations is assumed to be a random variable with Gaussian probability density function truncated at *v = 0 km/hr*. For this case a factor, *k* was introduced in [160, 161] where the probability density function (pdf) of the velocity is:

$$f_v(v) = \begin{cases} k \dfrac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(v-m)^2}{2\sigma^2}}, & v \geq 0 \\ 0, & , v < 0 \end{cases} \tag{6-11}$$

Here, *m* is the average speed of the mobile nodes in a cell. To define probability density function for the mobile nodes speed, *k* needs to be evaluated. We have (as defined in [160-161],

$$\begin{aligned} & k \int_0^{+\infty} \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(v-m)^2}{2\sigma}} dv = 1 \\ & \Rightarrow k = \frac{1}{\dfrac{1}{2} + \dfrac{1}{\sqrt{\pi}} \displaystyle\int_0^{\frac{m}{\sqrt{2\sigma}}} e^{-v^2} dv} \end{aligned} \tag{6-12}$$

If $(d,\ \theta)$ is the initial position of the IMS terminal, then the maximum length, *S* of the terminal trajectory in the cell can be defined as:

$$S = \sqrt{R^2 - d^2 \sin^2 \theta} + d \cos\theta, \quad \theta \in [0, 2\pi) \tag{6-13}$$

With the given initial velocity *Y*, the maximum time an IMS terminal *i* can spend in current cell is:

$$t_{h_i} = {S_i}/{Y_i} \tag{6-14}$$

The summary of the assumptions made in [159-161] to compute *S* with the application to the context of this chapter is provided below:

- IMS terminals are uniformly distributed within a cell
- The initial location of the IMS terminal is defined with radius *d* from the centre of the cell
- Angles for the direction of the movement are uniformly distributed

- Mobiles are allowed to move in any direction from the starting point
- Velocity of the mobiles is constant within a cell
- Initial velocity of the mobiles is assumed to be Gaussian pdf, truncated at 0 km/hr.
- Sessions from different terminals are independent.
- Equilibrium of handovers is assumed.

The relationship between handoff probability and the velocity of a mobile node has been extensively studied in [171-174]. S. Mohanty and I.F. Akyildiz (2006, [174]) showed that the probability of handoff failure increases with increasing speed of a mobile terminal both in inter and intra cell environment.

We now define the probability of an INVITE message reaching next node. The probability of a message failing to reach next node is, $q_n = 1 - p_n$. Since the session set up may fail at any node, we sum up all the probabilities that may fail at each node after stage one. The probability that a session will fail at the last but one node of the last stage, i.e. the success rate up to the last but one node is $p_n^{r-1} q_n$; the probability that a session will fail at the last but two node of the last stage .i.e. the success rate up to second last node is $p_n^{r-2} q_n$ and so on. The probability that a session will fail at the first node is $p_n q_n$.

Thus, the probability that a session will fail because of a node failure is,

$$Q_{Node} = p_n q_n + p_n^2 q_n + p_n^3 q_n + \ldots + p_n^{r-2} q_n + p_n^{r-1} q_n \qquad (6\text{-}15)$$

i.e.,

$$Q_{Node} = q_n (p_n + p_n^2 + p_n^3 + \ldots\ldots\ldots + p_n^{r-2} + p_n^{r-1}) \qquad (6\text{-}16)$$

Let

$$S = p_n + p_n^2 + p_n^3 + \ldots\ldots\ldots\ldots + p_n^{r-1} \qquad (6\text{-}17)$$

Multiplying by p_n to both sides of the above equation we get,

$$p_n S = p_n^2 + p_n^3 + ...+ p_n^r \qquad (6\text{-}18)$$

Subtracting Eq. (6-18) from Eq. (6-17),

$$S - p_n S = p_n - p_n^r$$

$$S(1 - p_n) = p_n - p_n^r$$

$$S = \frac{p_n(1 - p_n^{r-1})}{1 - p_n} \qquad (6\text{-}19)$$

Substituting the value of S into equation (6-16) we get,

$$Q_{Node} = \frac{q_n p_n (1 - p_n^{r-1})}{1 - p_n} = \frac{q_n p_n (1 - p_n^{r-1})}{q_n} = p_n (1 - p_n{}^{r-1}) \qquad (6\text{-}20)$$

The total probability that a session will fail due to node failure and/or handoff failure is:

$$Q = Q_{Node} + p_f + p_f . Q_{Node} \qquad (6\text{-}21)$$

Note that an intermediate node may be overlapped several times to count this probability and the source and destination node would be counted multiple times as well in the round trip of session establishment. In Eq. (6-20), we can observe that if r is high, then $Q_{node} \approx p_n$ i.e. the probability of a session failure (due to node failure) is almost equivalent to the probability of a message succeeding to reach next hop, if the number of total nodes involved in a session set up from stage two to four is very large. In that case, Eq. (6-21) becomes $Q \approx p_n + p_f + p_n p_f$.

It is mentioned earlier that the session is assumed to be successfully established at the second try if it fails at the first trial after stage one, and the total cost for Option 3 is the sum of both successful and unsuccessful case. Therefore,

$$C_3 = (1 - Q) C_{3\_S} + Q C_{3\_F} \qquad (6\text{-}22)$$

where

$$C_{3\_S} = U \, . \qquad (6\text{-}23)$$

The overhead for Option 3 is C_{3_S} i.e., sending BU if the session fails to set up in the first trail. It has been mentioned earlier a BU is sent in Option 3 after the first stage has been completed successfully in the session set up. Thus a session may fail after the BU is sent and therefore, sending BU will be the overhead for Option 3. The BU is not sent while a session fails to set up in Option 1 and Option 2. Thus there is no overhead for Option 1 and Option 2.

6.4 Simulation Model

We developed a system level simulator in OPNET modeller 11.5. Since the main aim of the study is to investigate the delay or cost of the three discussed options in mobile environment, we use the MIPv6 utility of OPNET with SIP messages. The message sizes for session sequences are provided in Table 6-1. These values are consistent with [69, 170, 175] (these papers evaluated SIP-based session performance under UDP in different types of networks for instance UMTS etc.). The experiment was performed over UDP as Transport layer protocol. We assume that each UDP datagram is carried over one IP packet. When SIP is carried by UDP, the reliability is ensured by SIP. UDP is the widespread SIP transport protocol [11]. The establishment of a session using UDP is illustrated in Figure 6-4.

Table 6-1: Message size for SIP over UDP/IPv6

Messages	Payload size (bytes)	Message size (bytes)
SIP INVITE	620	668
SIP183	500	548
SIP PRACK	250	298
SIP 200OK	300	348
SIP 180	230	278
SIP ACK	230	278
BU	572	620

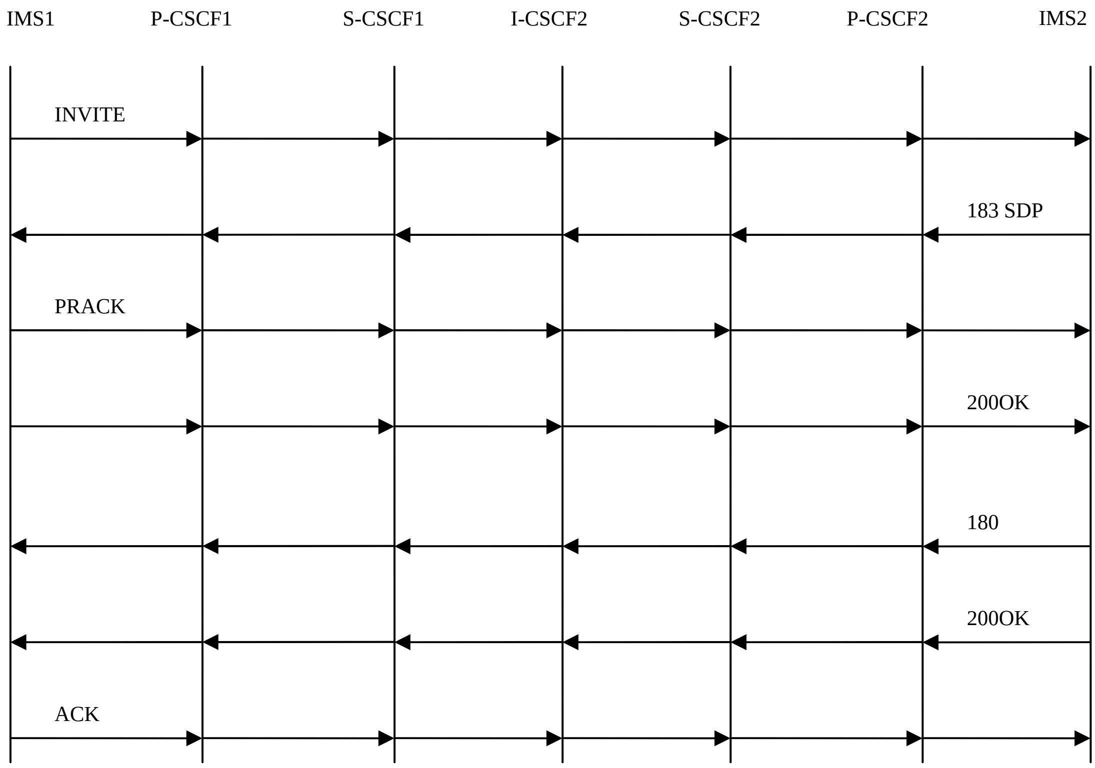


Figure 6-4: SIP session set up over UDP

SIP does not support TCP connections properly because the end points of a TCP connection are not kept constant with SIP mobility support [71]. Although the work in [28] by E. Wedlund, H. Schulzrine (1999) supports the complete range of applications by using SIP for real time communication and Mobile IP for TCP connections, most of the time TCP connections are short enough to make the cost of reconnection relatively small on the average. Another underlying protocol is Radio Link Protocol (RLP) that can be adopted by SIP. However, our test-bed is generated on availability basis. Since IPv6 is adopted, all messages sent have an UDP/IPv6 header of 48 bytes. The higher layer messages are passed to the dedicated 4.8 Kbps and 9.6 Kbps channel with SIP message service rate μ, where $\frac{1}{\mu} = 4*10^{-3}s$. The channel bandwidth represents the rate of the voice signalling traffic that is transmitted before session. A high bandwidth can reduce the overall delay drastically, but as bandwidth is a scarce resource, it is more relevant to investigate in the environment of smaller bit rates. If air frame link duration is known, then the number of frames in the air link for each SIP message can be computed. We set the air link duration per frame to 20ms as in [170, 176]. Therefore the radio channel contains $9.6*10^3*20*10^{-3}*\left(\frac{1}{8}bytes\right)=24$ bytes in each frame for 9.6 Kbps channel and $4.8*10^3*20*10^{-3}*\left(\frac{1}{8}bytes\right)=12$ bytes in each frame for 4.8 Kbps channel. This leads the number of air link frames in SIP messages to $\frac{message\ size}{24}$ and $\frac{message\ size}{12}$ for 9.6 Kbps and 4.8 Kbps channel respectively. The number of air link frame can be used to compute the packet loss rate in the channel. We consider the Frame Error Rate (FER) be the probability of a frame being erroneous in the air link. Therefore, (1-*FER*) is the probability of a frame not being in error in the air link. If we know the number of frames contained in one UDP packet then, $(1-FER)^{NoF}$ is the probability that the UDP packet is not erroneous (Here, *NoF* = Number of Frames). Hence, the packet loss rate is $(1-(1-FER)^{NoF})$. The message retransmission depends on type of message and the number of lost packets. For instance, the probability of a retransmission of SIP INVITE will depend on the first INVITE packet (consider INVITE containing $frame_{INVITE}$ frames) is lost or that the first packet is received but the response SIP 183(consider 183 containing $frame_{183}$ frames) is lost. Therefore, the probability of having a retransmission of INVITE over SIP UDP is equal to $(1-(1-FER)^{frame_{INVITE}+frame_{183}})$. We set the end to end (node to node) propagation delay, D_P, to 100ms as in [71, 170, 176]. The delay introduced by the Internet depends on the number of routers and the type of links in the path of datagram transmission. We assume the one way Internet transmission delay, D_I, over the wired network to be constant and equal to 50ms.

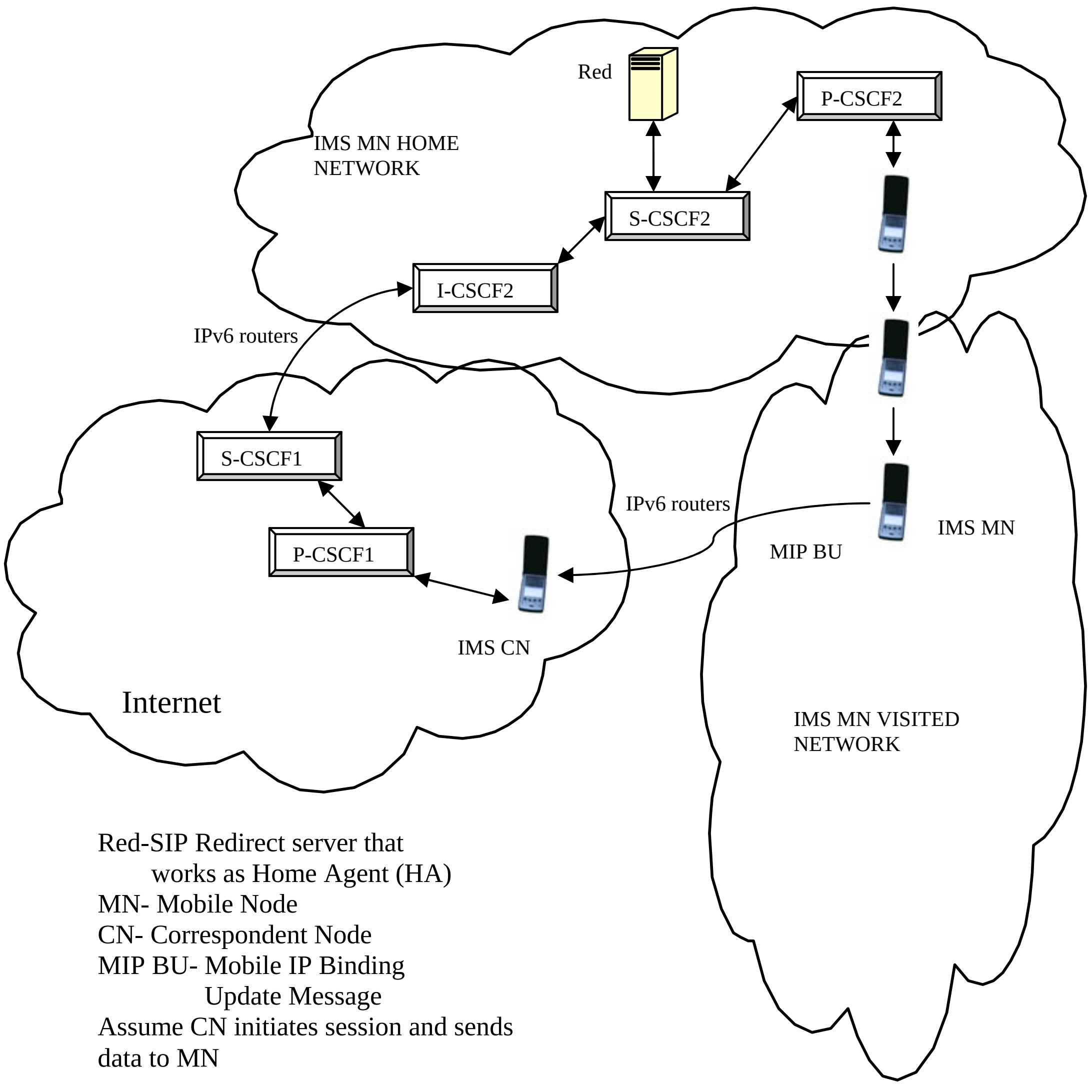


Figure 6-5: Experimental test-bed prototype

The experimental test-bed configuration in Opnet modeller 11.5 with adoption of IPv6 is provided in Figure 6-5. The configuration is the simulation of both MN and CN getting serviced by same operator [24] (Figure 2-2 in chapter 2). There are six intermediate nodes in between MN and CN. We assume that CN initiates session and sends data to MN. The SIP Redirect server which is an application server in IMS works as the HA here. Usually, SIP redirect servers are used for call forwarding services where they generate a SIP 302 (Moved Temporarily) response with contact details of the mobile terminals. Also, we assume the handoff occurs successfully only once throughout a session set up after stage one i.e., after the SIP 183 has been sent as MN changes network. This nullifies the handoff failure issue in our simulation model. Alternatively, as mentioned earlier prediction mechanisms can be employed to predict the location update instance based on speed of a MN. Nonetheless, we consider that the MN changes network after stage one of session set up in our simulation model. The measurement of handoff flow analysis and SIP session set up delay are mature topics today [71, 88, 69, 12, 81, 163, 167, 168, 169, 170, 173, 179] and is of not much interest in this instance. For this reason, we do not address the issues of router selection, duplicate address detection (DAD), router table update etc. Our objective here is to simulate and capture the delay cost of the three Options mentioned. As mentioned earlier, the main components introducing delay are queuing delays, wireless propagation delay, Internet transmission delay which depend on the number of routers, message arrival and service rate. For the optimized route i.e., for the route MN sends BU to CN, we consider the Internet transmission delay is double (100ms) than the normal route of SIP Red. We double this delay simply because of the logistic that the Internet path length will increase as MN changes network and thus there will be more routers in this route. However, the queuing delays will decrease as there is less number of nodes (only 2 servers 1. P-CSCF#1 for CN 2. P-CSCF#2 for MN i.e., n in Eq. (6-1) is 2) in the optimized route. Packets/data are sent from CN to MN only. Since SIP is an application/session layer protocol, the SIP based messages may not be served with highest priority in the associated components and this may introduce additional delay. However, we assume that the IMS servers/nodes are the recipients of SIP messages only. We have assumed the M/M/1 queuing model for the IMS servers. It is expected that the cost incurred in our model will reflect the cost behaviour including delays for non-SIP related messages. The default 'dra_ber' model (which is used to set the bit error rate) of Opnet wireless module has been adjusted to evaluate the bit error rate and accordingly frame error rate (FER). The FER is varied between 0-10 percent to compute the cost of delays.

6.5 Simulation Results

First we simulate the cost over 4.8 Kbps channel for SIP message arrival rate 50msg/sec, 100msg/sec and 200msg/sec. Figure 6-6, Figure 6-7 and Figure 6-8 depict the cost behaviour of the three options. The cost of Option 3 was calculated for successful set up in the first trial only i.e., for $C_3 = C_{3\_S} = U$ only. All three figures suggest that the cost of Option 3 with successful session set up in the first trial performs the best while cost of Option 1 performs the worst. This is obvious since the cost of Option 3 with successful session set up includes sending a BU only. We also find from the results that the cost difference increases for higher FER, specially after 2% of the FER. The reason for this is the message retransmission rate increases as the FER rate goes up. With 10% FER, the cost of Option 1, Option 2 and Option 3 were recorded 34s, 19s and 12.7s for arrival rate 50msg/s; 35.7s, 19.95s and 13.335s for arrival rate 100msg/s and 37.842s, 21.147s and 14.1351s for arrival rate 200msg/s respectively. These values suggest that with mean service time, $\frac{1}{\mu} = 4 * 10^{-3} s$ the message arrival rate does not affect the costs much. It can be concluded that the results will exhibit same behaviour for all arrival rate where $\rho < 1$.

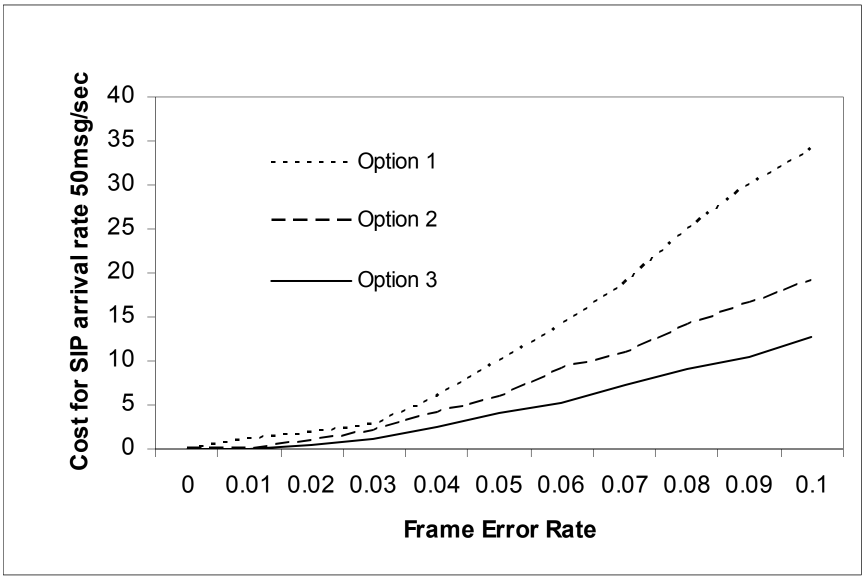


Figure 6-6: Cost comparison for 4.8 Kbps, arrival rate 50msg/s

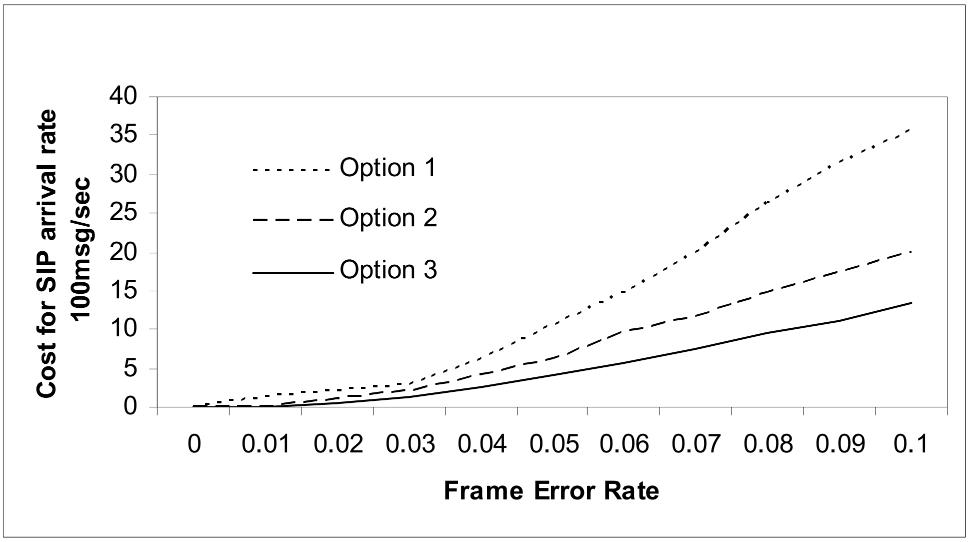


Figure 6-7: Cost comparison for 4.8 Kbps, arrival rate 100msg/s

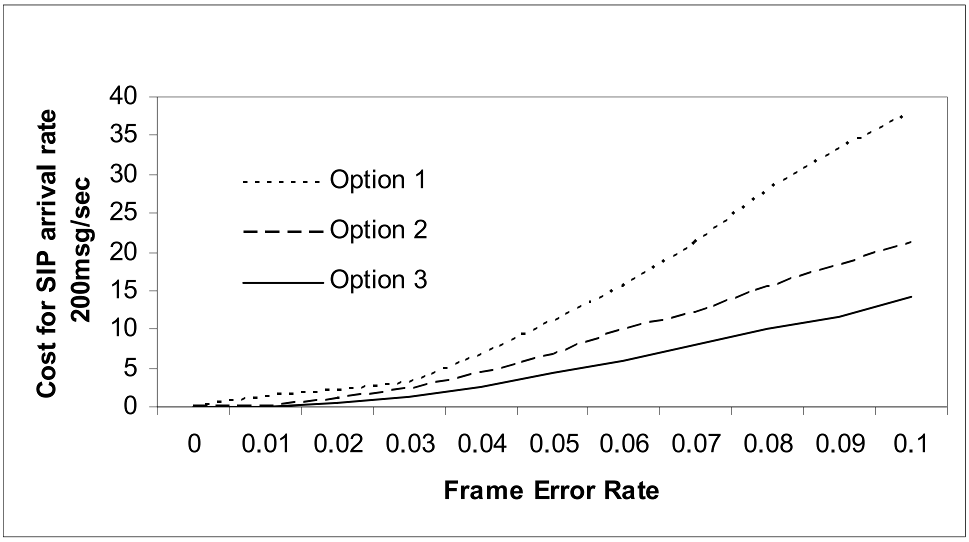


Figure 6-8: Cost comparison for 4.8 Kbps, arrival rate 200msg/s

We use the same results to draw Figure 6-9, Figure 6-10 and Figure 6-11 for Option 3 with successful session set up in the second trial i.e., for $C_3 = C_{3\_F} = 2C_{3\_S} = 2U$ using 4.8Kbps channel. We find that Option 2 performs the best among all for all different arrival rates. The reason of Option 3 taking more time is that the BU needs to be sent twice. Thus the overhead here is *U*.

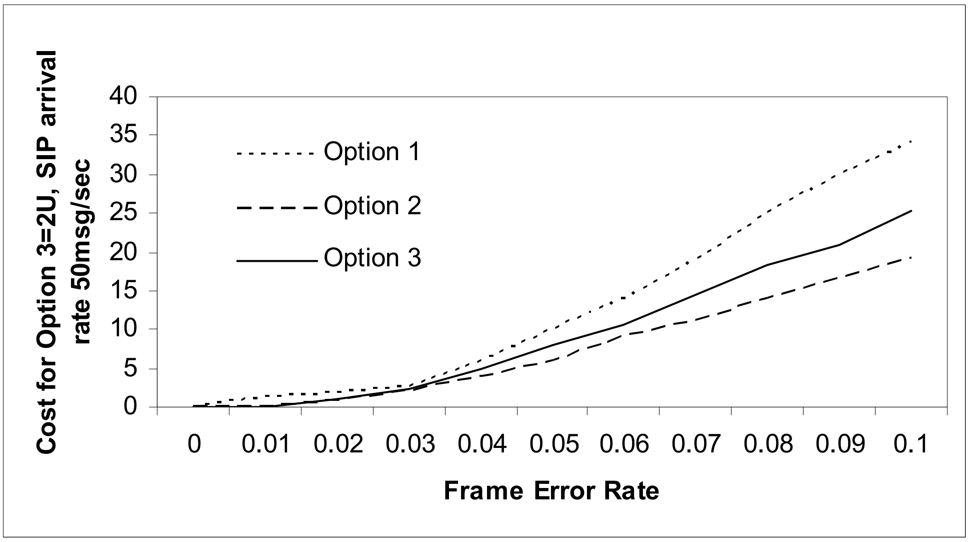


Figure 6-9: Cost comparison for Option 3 being successful in 2nd trial with 4.8Kbps channel and arrival rate 50msg/s

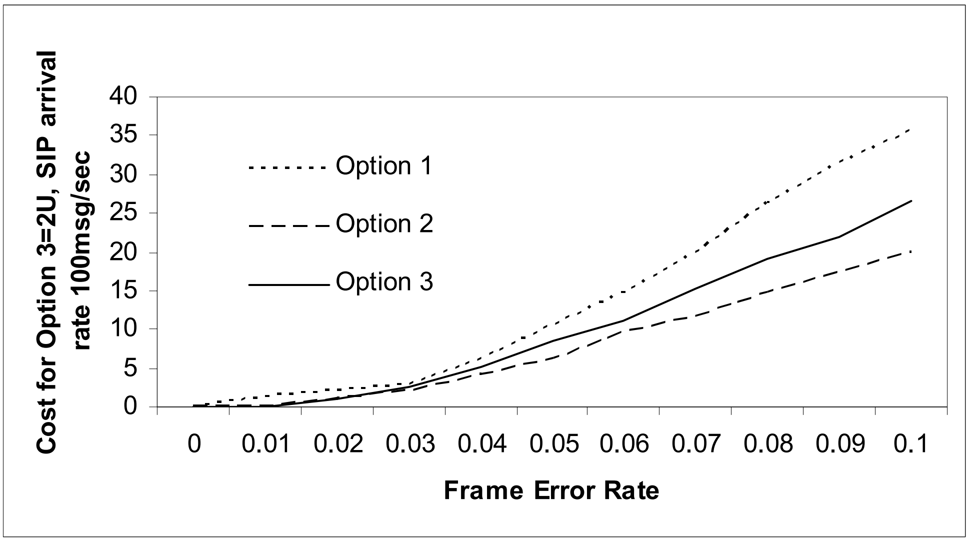


Figure 6-10: Cost comparison for Option 3 being successful in 2nd trial with 4.8 Kbps channel and arrival rate 100msg/s

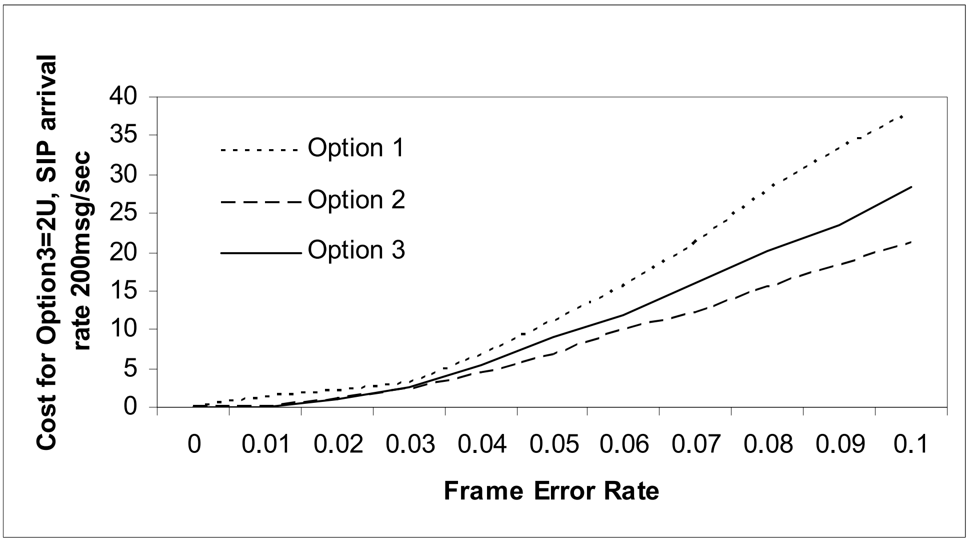


Figure 6-11: Cost comparison for Option 3 being successful in 2nd trial with 4.8 Kbps channel and arrival rate 200msg/s

Next we simulate the cost over 9.6 Kbps channel for SIP message arrival rate 50msg/sec, 100msg/sec and 200msg/sec. Figure 6-12, Figure 6-13, Figure 6-14 depict the cost behaviour of the three options. Again, the cost of Option 3 was calculated for successful set up in the first trial only i.e., for $C_3 = C_{3\_S} = U$ only. All three figures suggest that the cost of Option 3 with successful session set up in the first trial performs the best while cost of Option 1 performs the worst. This is the obvious from the same reason discussed above for 4.8Kbps channel. We find sharp increase in the cost for higher FER, specially after 3% of the FER. Also, the curves tend to smoothen up after 9% of FER. With 10% FER, the cost of Option 1, Option 2 and Option 3 were recorded 13s, 7.3s and 4.6s for arrival rate 50msg/s; 13.117s, 7.3511s and 4.6276s for arrival rate 100msg/s and 13.6417s, 7.6451s and 4.8127s for arrival rate 200msg/s respectively. Again, these values suggest that with mean service time, $\frac{1}{\mu} = 4*10^{-3}s$ the message arrival rate does not affect the costs much for 9.6Kbps channel.

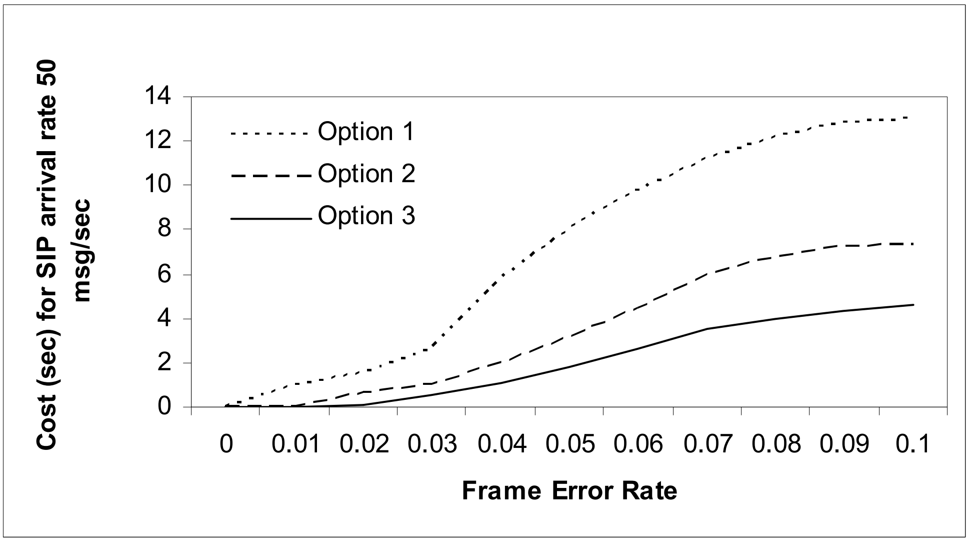


Figure 6-12: Cost comparison for 9.6 Kbps, arrival rate 50msg/s

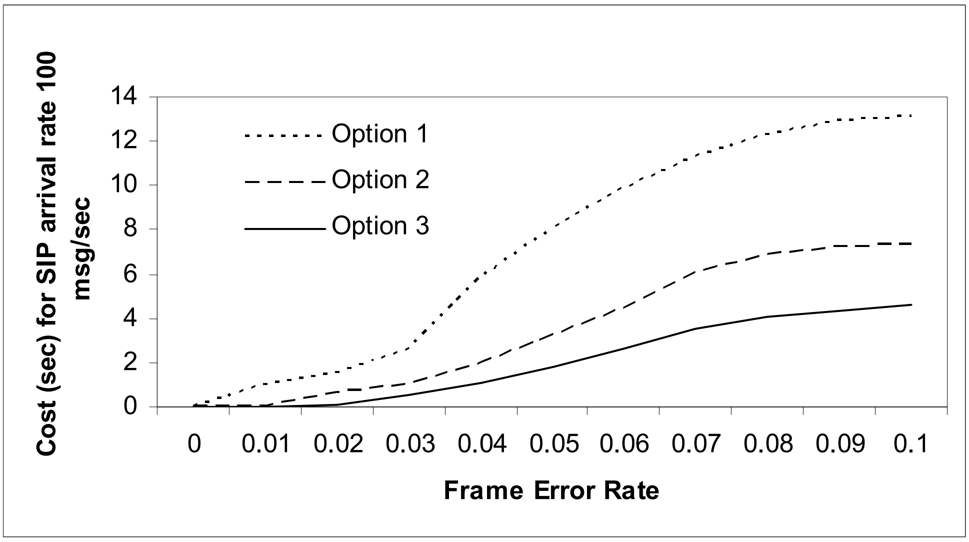


Figure 6-13: Cost comparison for 9.6 Kbps, arrival rate 100msg/s

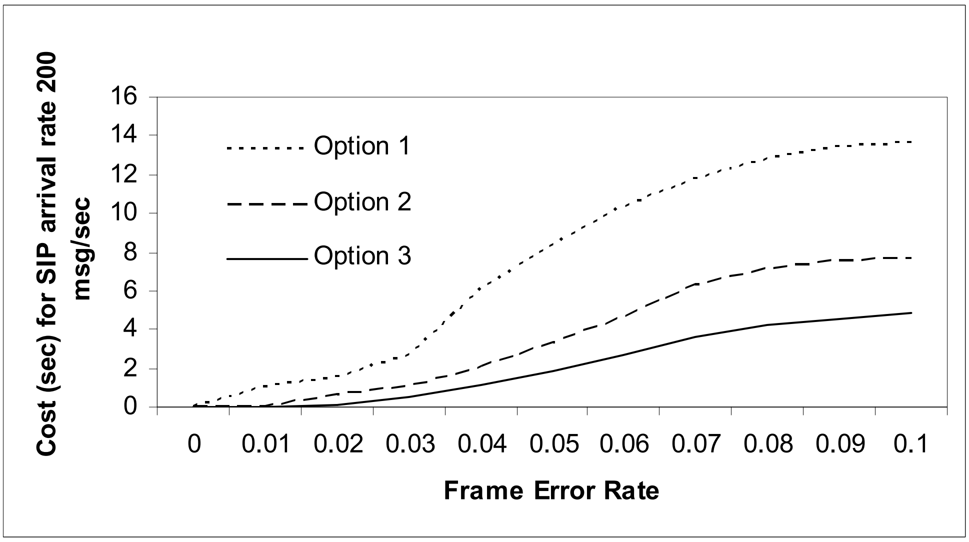


Figure 6-14: Cost comparison for 9.6 Kbps, arrival rate 200msg/s

We use the above results to draw Figure 6-15, Figure 6-16 and Figure 6-17 for Option 3 with successful session set up in the second trial i.e., for $C_3 = C_{3\_F} = 2C_{3\_S} = 2U$ using 9.6Kbps channel. We find that Option 2 performs the best among all for all different arrival rates for 9.6Kbps channel from and above 3% error rate. However, we find that cost of Option 3 still performs better with set up success at the second trial when the FER is below 3%. This is the impact observed for doubling the bandwidth.

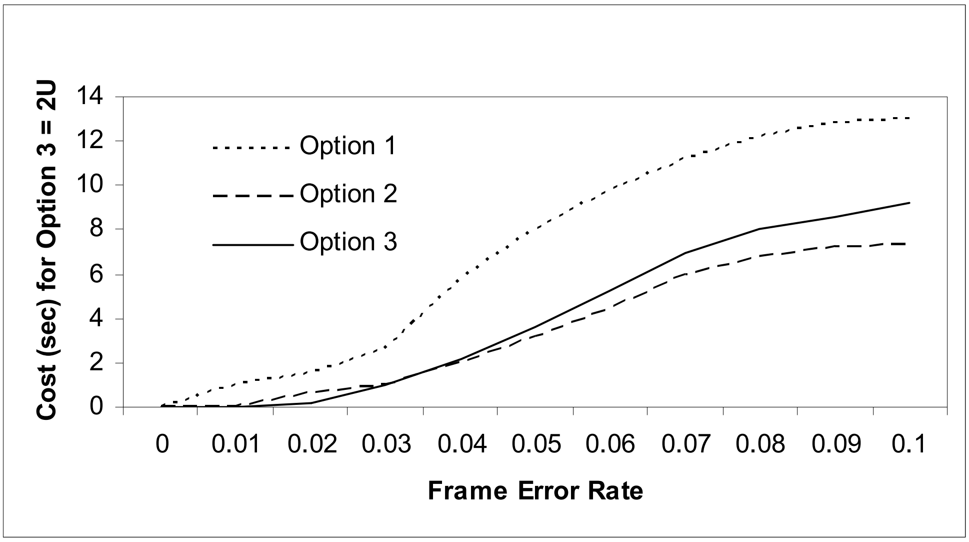


Figure 6-15: Cost comparison for Option 3 being successful in 2nd trial with 9.6Kbps channel and 50msg/s

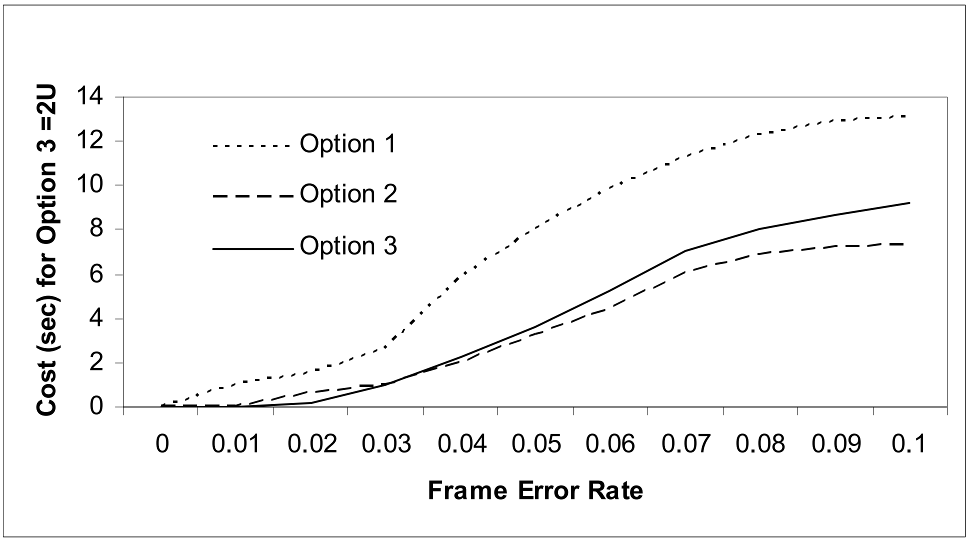


Figure 6-16: Cost comparison for Option 3 being successful in 2nd trial with 9.6Kbps channel and 100msg/s

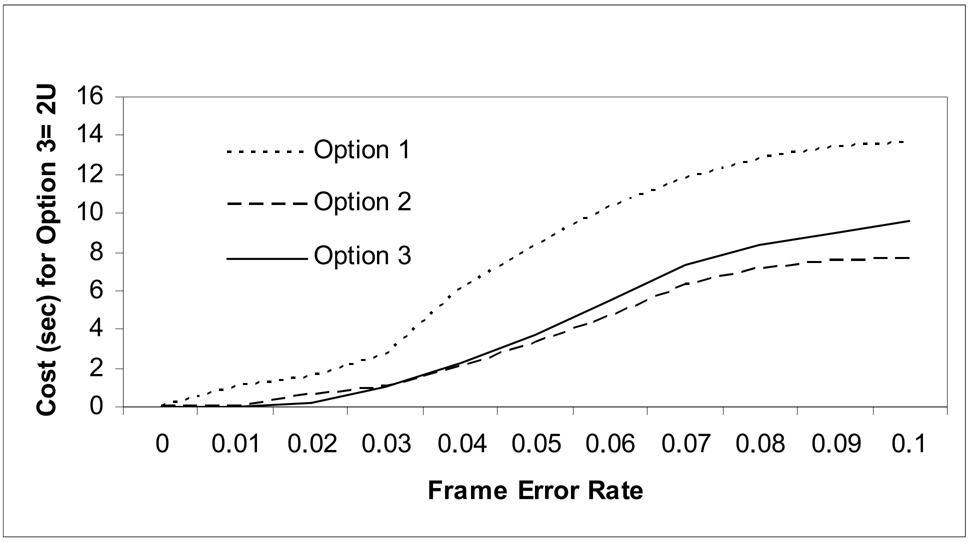


Figure 6-17: Cost comparison for Option 3 being successful in 2nd trial with 9.6Kbps channel and 200msg/s

Comparing the above figures it is trivial to see that doubling the bandwidth of the channel reduces the cost significantly for same arrival rate. The reduced delay in seconds and the percentage gain for each Option with respect to FER are furnished in Table 6-2, Table 6-3 and in Table 6-4 for the increased bandwidth shift from 4.8Kbps to 9.6Kbps. All the gains are positive which represents reduced cost for higher bandwidth channel (9.6Kbps) compared to the smaller bandwidth channel (4.8Kbps). A steady increase in the percentage gain can be observed against the later FER values in the tables. Looking at the percentage gain of last few rows in the tables, it can be stated that all the three options gain almost similar benefit from doubling the bandwidth. The cost is reduced more than 50% for all three options for over 7% FER. Thus the more the bandwidth, the lesser the cost incurred by the three methods.

Table 6-2: Percentage gain for doubling bandwidth with arrival rate 50msg/s

FER	Gain with Option 1		Gain with Option 2		Gain with Option 3	
	Second	percent	second	Percent	Second	Percent
0	0.002	50	0.001	50	0.0001	50
0.01	0.2	16.66667	0.01	25	0.001	14.28571
0.02	0.4	21.05263	0.3	33.33333	0.418	83.6
0.03	0.1	3.703704	1.01	50.5	0.7	58.33333
0.04	0.19	3.166667	2	50	1.3	54.16667
0.05	2	20	2.8	46.66667	2.2	55
0.06	4.2	30	4.6	51.11111	2.7	50.9434
0.07	7.7	40.74074	5	45.45455	3.7	51.38889
0.08	12.8	51.2	7.2	51.42857	5.1	56.04396
0.09	17.2	57.33333	9.3	56.36364	6.2	59.04762
0.1	21	61.76471	11.7	61.57895	8.1	63.77953

Table 6-3: Percentage gain for doubling bandwidth with arrival rate 100msg/s

FER	Gain with Option 1		Gain with Option 2		Gain with Option 3	
	second	percent	second	percent	second	Percent
0	0.002182	51.952381	0.001093	52.047619	0.000109	52.095238
0.01	0.251	19.920635	0.01179	28.071429	0.001314	17.877551
0.02	0.4815	24.135338	0.3408	36.063492	0.442508	84.287238
0.03	0.2116	7.4638448	1.10307	52.527143	0.757	60.079365
0.04	0.43771	6.9477778	2.186	52.047619	1.4134	56.087302
0.05	2.428	23.12381	3.0776	48.850794	2.3892	56.885714
0.06	4.8118	32.733333	5.0192	53.113228	2.9494	52.999102
0.07	8.5442	43.054674	5.508	47.688312	4.039	53.425926
0.08	13.9402	53.105524	7.8524	53.417687	5.531	57.885924
0.09	18.5848	58.999365	10.0746	58.150649	6.6992	60.763719
0.1	22.583	63.257703	12.5989	63.152381	8.7074	65.297338

Table 6-4: Percentage gain for doubling bandwidth with arrival rate 200msg/s

FER	Gain with Option 1		Gain with Option 2		Gain with Option 3	
	second	**percent**	**second**	**percent**	**second**	**Percent**
0	0.002353	52.85894	0.001179	52.95238	0.000118	52.9991
0.01	0.28624	21.43157	0.013102	29.42857	0.001514	19.42703
0.02	0.54066	25.56675	0.373332	37.26984	0.470708	84.58371
0.03	0.276764	9.20981	1.189193	53.42286	0.81248	60.83258
0.04	0.581218	8.70348	2.35744	52.95238	1.520336	56.91584
0.05	2.73512	24.5743	3.326704	49.81587	2.568768	57.69919
0.06	5.298272	34.00252	5.408968	53.99788	3.178676	53.88591
0.07	9.282868	44.12911	5.95932	48.67532	4.35176	54.30468
0.08	15.02281	53.99033	8.460496	54.2966	5.94334	58.68053
0.09	19.95819	59.77296	10.82408	58.94026	7.187668	61.50403
0.1	24.20032	63.95095	13.50186	63.84762	9.322396	65.95211

Figure 6-18, Figure 6-19 and Figure 6-20 show the cost behaviour of Option 3 for various session failure probability Q for 9.6 Kbps channel which are derived from Eq. (6-22) using the previous results. The higher the session failure probability, the higher the cost of Option 3.

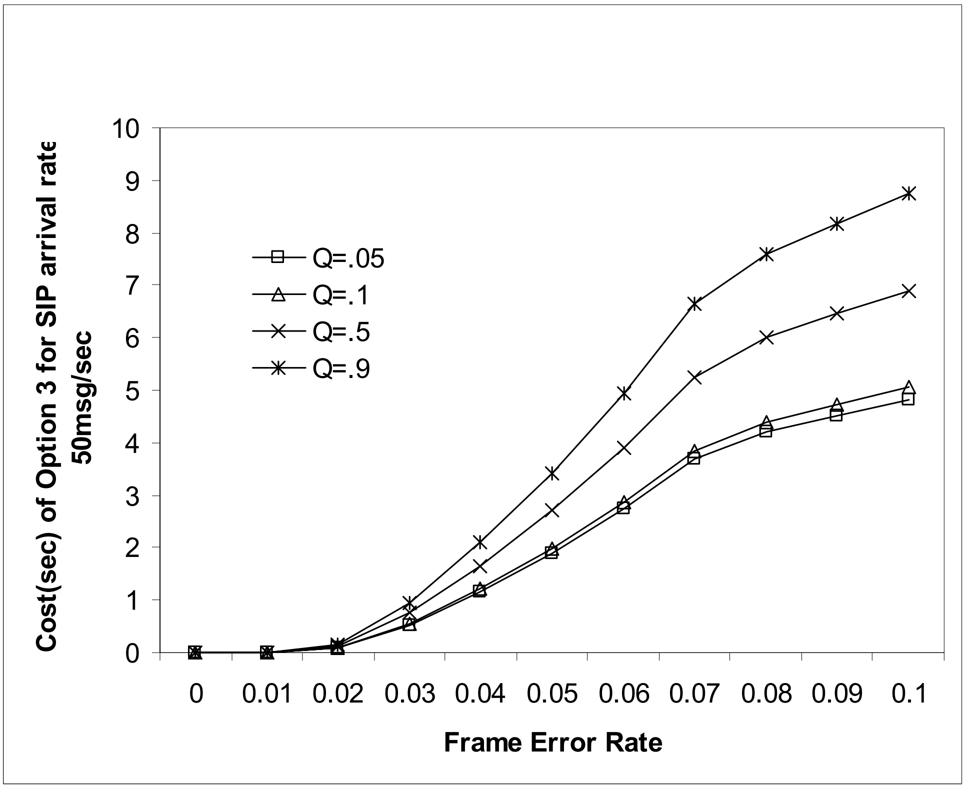


Figure 6-18: Option 3 cost for varying Q with arrival rate 50msg/s and 9.6Kbps channel

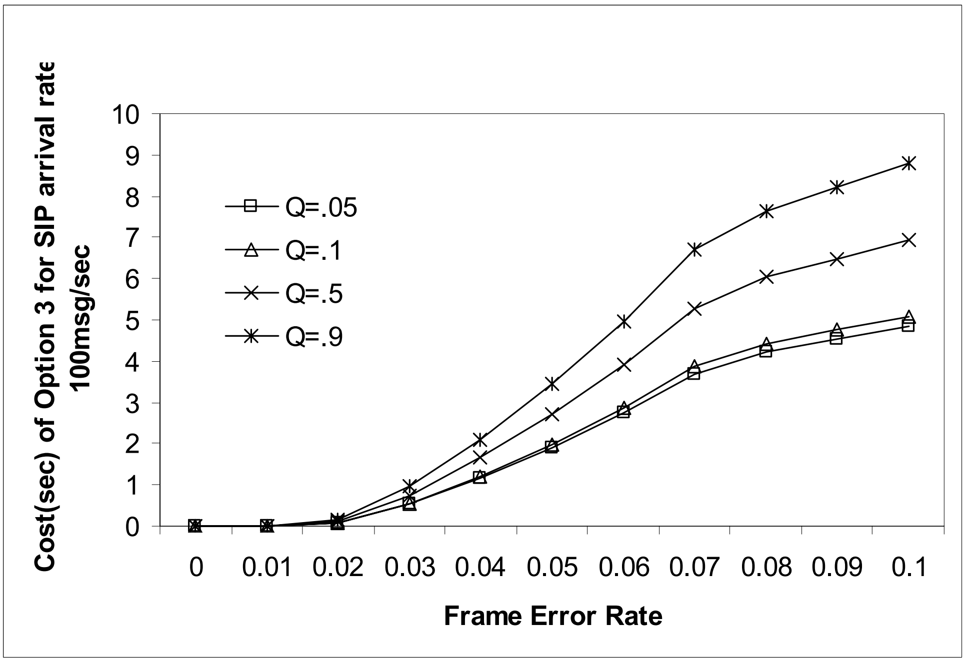


Figure 6-19: Option 3 cost for varying Q with arrival rate 100msg/s and 9.6Kbps channel

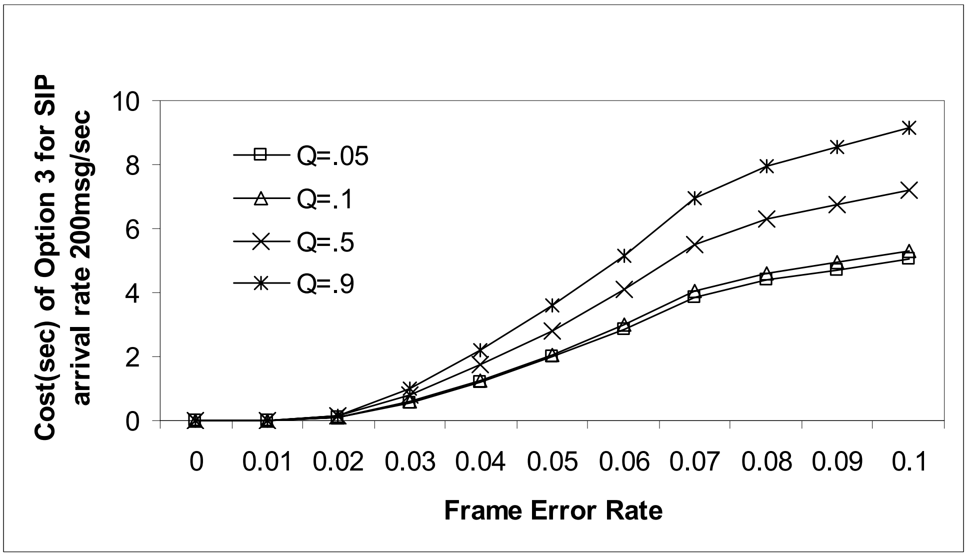


Figure 6-20: Option 3 cost for varying Q with arrival rate 200msg/s and 9.6Kbps channel

We have the session failure probability for various handoff probabilities in Figure 6-21. The curves are linear which implies both handoff and node failure have equal impact on the session failure probability. In our test-bed the value *r* in Eq. (6-20) is 30 as there are 6 intermediate nodes and according to Figure 6-4, 5 message flows are needed to complete a session set up after stage one.

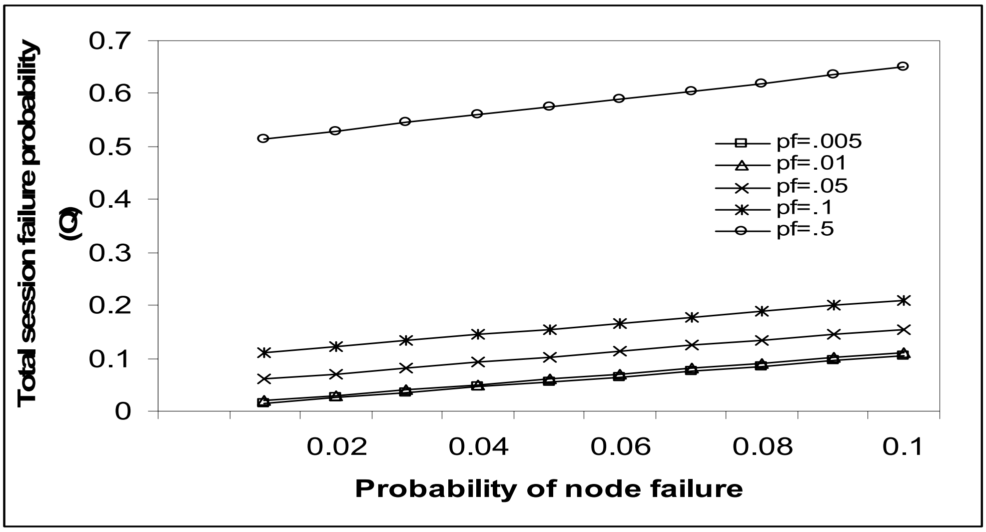


Figure 6-21: Q for various p_f

Figure 6-22 and Figure 6-23 show how little the message arrival rate affects the costs of all options for both channel-bandwidths. Although very low, the increase of cost in 4.8Kbps channel is found more than that in 9.6Kbps channel for FER of 5%. For 4.8 Kbps channel, cost increase for message arrival rate increase from 50msg/sec to 200msg/sec was found to be 1.13s, 0.678s and 0.452s only for Option 1, Option 2 and Option 3 respectively. For 9.6 Kbps channel, cost increase for message arrival rate increase from 50msg/sec to 200msg/sec was found to be 0.39488s, 0.151296s and 0.083232s only for Option 1, Option 2 and Option 3 respectively.

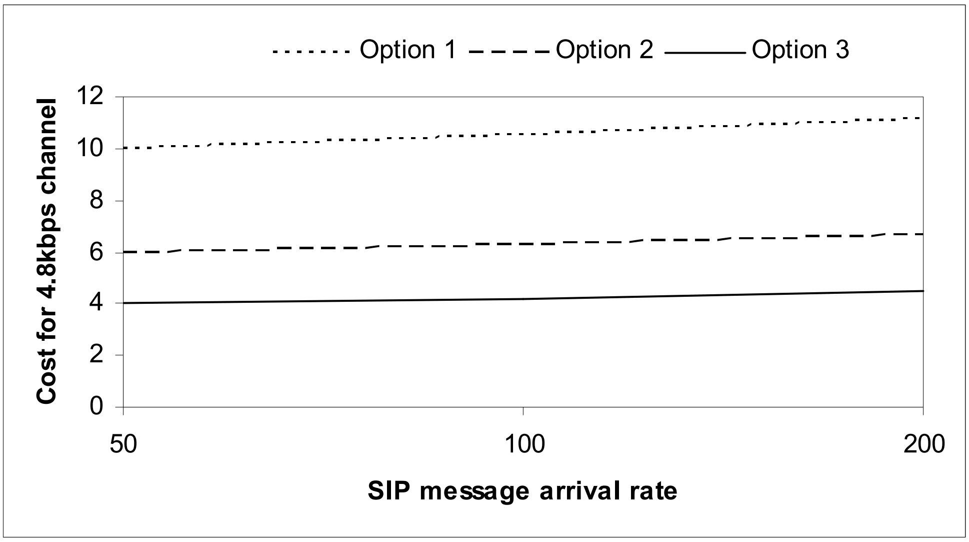


Figure 6-22: Cost for increased arrival rate with 4.8 Kbps channel

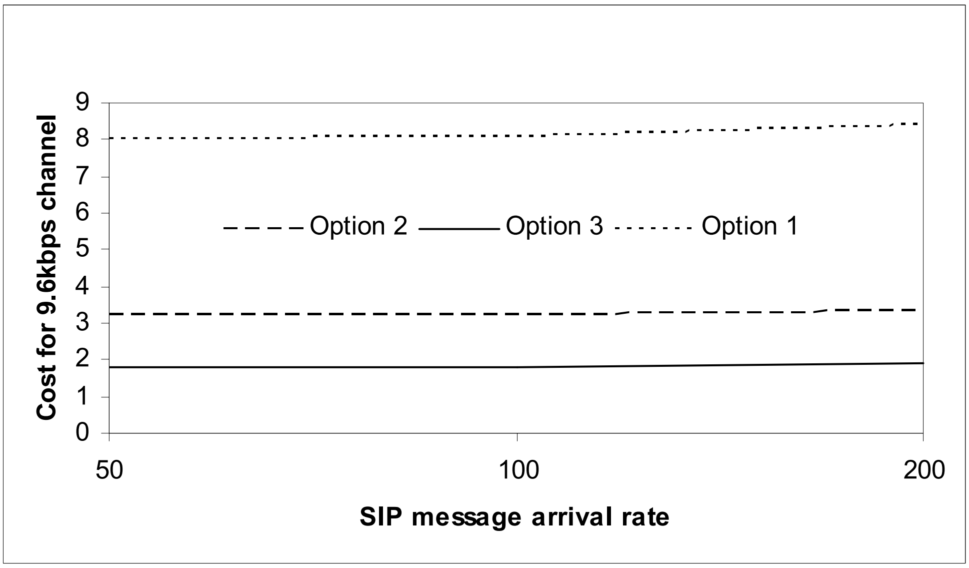


Figure 6-23: Cost for increased arrival rate with 9.6Kbps channel

Further we capture the packet loss rate for growing FER in Figure 6-24 and in Figure 6-25. We see that the big sized messages have higher loss rate. Also, the packet loss rates in 4.8Kbps channel are greater than those in 9.6Kbps channel.

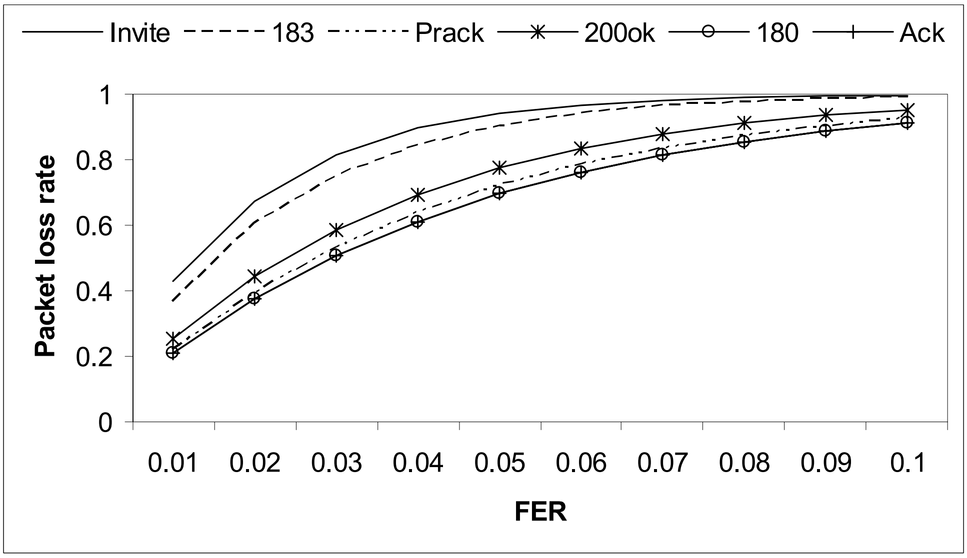


Figure 6-24: Packet loss rate in 4.8Kbps channel

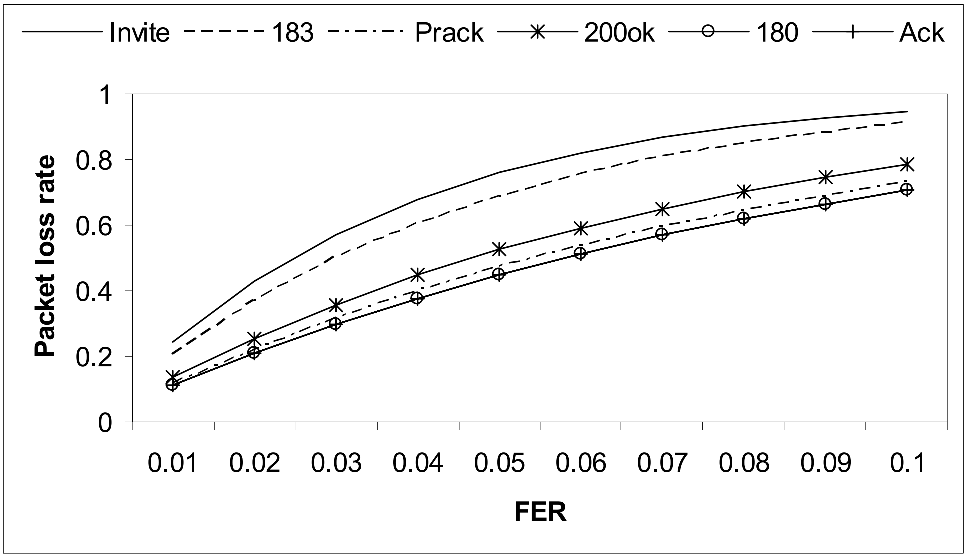


Figure 6-25: Packet loss rate in 9.6Kbps channel

6.6 Threshold from Simulation

The simulation results imply that Option 2 exerts better performance than Option 1. However, Option 3 incurs the least delay if the session succeeds to set up in the first trial. We need to decide when to use Option 3 since the cost would be high for the higher session failure probability. If a session fails, the cost of Option 3 simply doubles up which is much higher than the cost of Option 2 as shown in the above simulation for higher FER. We define a threshold, *P* for that and compare with *Q*. If *Q* exceeds the threshold, we do not perform Option 3 and invoke other option. After the round trip of stage one and the handoff, the number of nodes involved (*r, n and x*) in the session set up and in BU message is known.

Let the threshold probability be *P.* The cost for Option 3 in successful first trial session establishment is therefore defined to be:

$$\frac{C_{3\_S}}{P} \tag{6-24}$$

where *P* is the highest probability for a session to be successfully set up in the first trial.

Since either Option 1 or Option 2 is invoked in case the session fails to set up at the first trial, the total cost for possible session set up option is:

$$[C_1 + C_2] \tag{6-25}$$

Therefore from Eq. (6-24) and Eq. (6-25), we get:

$$P > \frac{C_{3\_S}}{[C_1 + C_2]} \tag{6-26}$$

Our simulation suggests that $C_1 > C_2$ i.e., setting P equal to $\frac{C_{3\_S}}{C_2}$ will achieve the threshold (which also saves the computation of C_1). Eq. (6-26) determines the criteria whether Option 3 is applicable. Thus after stage one in the session set up both P and Q are calculated. The parameters of P are available after stage one of a session set up as mentioned earlier. The probability of message failing to reach next hop, q_n can be calculated from the computation of packet loss rate as discussed earlier assuming packet loss rate is equal to the rate message failing to reach next hop. We assume that the handoff failure probability, p_f is provided in a network (alternatively methods discussed in the modelling section can be used to compute the handoff failure probability). With, the values of q_n and p_f, Q can easily be computed after stage one of a session establishment. Only if $Q<P$, then Option 3 is performed in IMS. Because of the simple nature of Eq. (6-26) and Eq. (6-21), the delay to compute both P and Q are negligible.

We determine the characteristics of the threshold P against packet loss rate of 200OK message in Figure 6-26 and in Figure 6-27 for two different channels from the simulated data. We compute P for both $\frac{C_{3\_S}}{[C_1 + C_2]}$ and $\frac{C_{3\_S}}{C_2}$. Obviously, the values of first ratio is smaller then the later i.e., $\frac{C_{3\_S}}{[C_1 + C_2]} < \frac{C_{3\_S}}{C_2}$. Both the curves get steady shape after 3% of FER rate. If we represent packet loss rate as one of the criteria of the session failure rate then we see that as FER increases the session set up failure rate increases and becomes greater than the threshold. We used 200OK message for packet loss rate as that is the biggest message after stage one. The smaller messages will have lower loss rate and thus will exceed the values of threshold for much higher FER. However, we will use Option 3 only there is less possibility of a session failure as the cost doubles up for a failure at the first trial. We also find that for 4.8 Kbps channel we should not use Option 3 after 3% FER and for 9.6 Kbps channel Option 3 should not be used after 6% FER. Thus with higher bandwidth the possibility of Option 3 usage is higher.

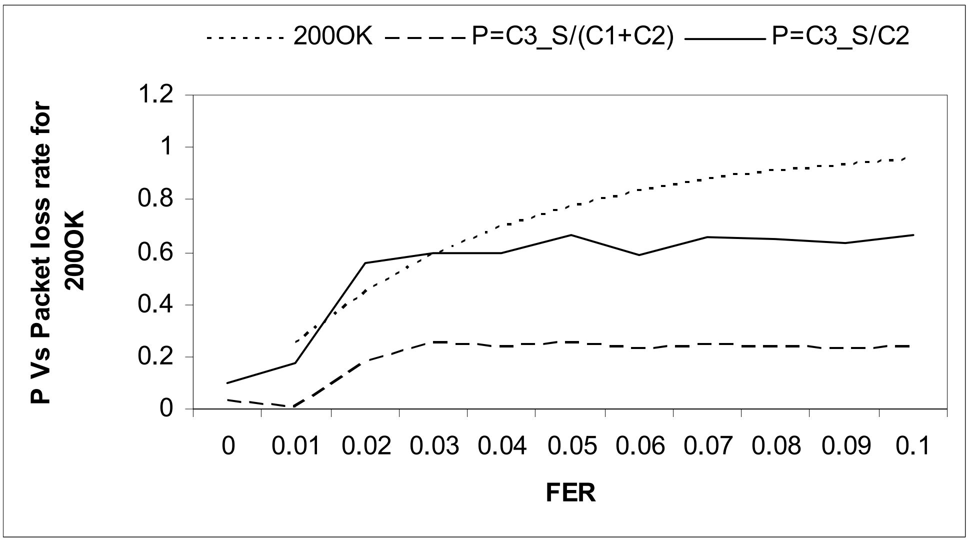


Figure 6-26: P Vs Packet loss rate for 4.8Kbps channel with arrival rate 50msg/s

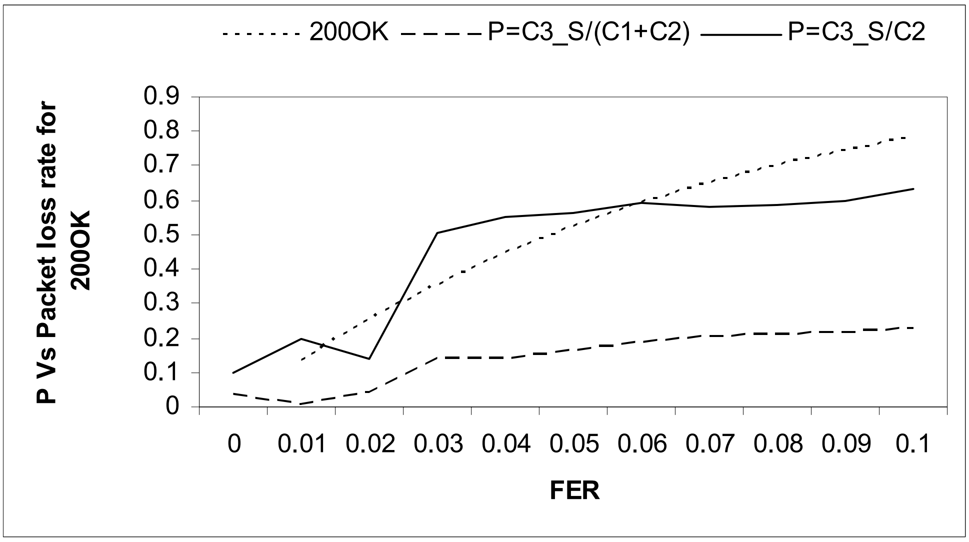


Figure 6-27: P Vs Packet loss rate for 9.6Kbps channel with arrival rate 50msg/s

The values of threshold *P* (under 3% FER for 4.8 Kbps channel and under 6% FER for 9.6 Kbps channel) in the above discussion was derived from simulation results i.e., they are channel specific. It is indeed important to derive such cut off point dynamically during session set up so that an end IMS terminal can decide which option to initiate when. In order to compute the parameters of *P* and *Q* manually, analysis is provided in the next section.

6.7 Queuing Analysis for Nodes

Since our cost functions depend on the addition of node delays to process messages/data, result of Open Jackson Network [133, 162] can be used to approximately count the mean number of packets delivered, total expected number of packets in the network and the expected sojourn time of packets. We utilized the M/M/1 behaviour of Opnet module for every node to perform performance analysis in the simulation section. The justification of choosing Jackson Network is due to the fact that it provides results for network that contains M/M/1 queues.

For an open Jackson network consisting K *M/M/1* queues with $\lambda_i < \mu_i$, for all $i = 1,2,...,K,$ the mean number of jobs at steady state in the network is

$$\pi(\underline{l}) = \prod_{i=1}^{K} \left(\frac{\mu_i - \lambda_i}{\mu_i} \right) \left(\frac{\lambda_i}{\mu_i} \right)^{l_i} \qquad \forall \underline{l} = (l_1,...,l_K) \in \mathrm{N}^K \qquad (6\text{-}27)$$

where, $(\lambda_1, \lambda_2, ..., \lambda_K)$ is the unique nonnegative solution of the system of linear equations:

$$\lambda_i = \lambda_i^0 + \sum_{j=1}^{K} p_{ij} \lambda_j \qquad i = 1,2...,K. \qquad (6\text{-}28)$$

Here, jobs arrive from outside the system joining queue *i* according to a Poisson process with rate λ_i^0 . After service at queue *i,* which is exponentially distributed with parameter μ_i, the job either leaves the system with probability p_{i0}, or goes to queue *j,* with probability p_{ij}. Clearly, $\sum_{j=0}^{K} p_{ij} = 1,$ since each job must go somewhere.

The above result can be applied to compute *U, y* and $N \sum_{i=1}^{x} D_i$ manually. For instance, in order to find end to end delay of packets to send BU, the expected number of packets in the route needs to be computed. For the simulation purpose, we generated message (for session set up) i.e., from CN and send packet (for data transfer) from one terminal only. Also, we kept the mean service rate same for all nodes for simplicity in the simulation section. Thus, to replicate the simulation environment here via modelling, if there is *n* number of nodes in between the MN and CN to send BU, then the route can be thought to be composed of *n* nodes in series, each modelled as M/M/1 queue with common service rate μ (see Figure 6-28). In a nutshell, we now have a Jackson network with *n* M/M/1 queues where $\lambda_i^0 = 0$ (for Eq. (6-28)) for *i=2,3,…n* (no external arrivals at nodes *2,3,…n* since packets are generated from one side only and packets enter to the next node in series after getting serviced),

$\mu_i = \mu \quad for \quad i = 1,2,...,n; \quad p_{ii+1} = 1 \quad for \quad i = 1,2,...n$. If p_e is the probability of packets being received correctly by the destination node, then the retransmission probability by the source is $(1 - p_e)\lambda_1$. Under these assumptions, we find,

$$\lambda_i = \lambda_{i-1} \quad for \ i = 2,3,...,n, \tag{6-29}$$

and

$$\lambda_1 = \lambda^0 + (1 - p_e)\lambda_1$$
$$\lambda_1 = \frac{\lambda^0}{p_e} \tag{6-30}$$

Here, λ^0 is the rate the CN sends data for MN or vice versa.

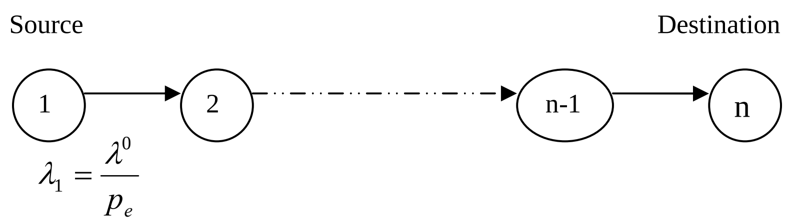


Figure 6-28: *n* M/M/1 queues in series

Hence, from the result of Jackson network, the number of packets in the route for $\lambda^0 < p_e\mu$ is given by

$$\pi(\underline{l}) = \left(\frac{p_e\mu - \lambda^0}{p_e\mu}\right)^n \left(\frac{\lambda^0}{p_e\mu}\right)^{l_1+...+l_n} \qquad \forall \underline{l} = (l_1, l_1, ..., l_n) \in \mathrm{N}^n \tag{6-31}$$

In particular, the probability of having z_i packets in node *i* and z_j packets in node *j*(>*i*) is given by:

$$\Pr_{ij}(z_i, z_j) = \sum_{l_U \geq 0, U \notin \{i,j\}} \pi\left(l_1,..,l_{i-1}, z_i, l_{i+1},...l_{j-1}, z_j, l_{j+1},..., l_n\right)$$

$$= \left(\frac{p_e\mu - \lambda^0}{p_e\mu}\right)^2 \left(\frac{\lambda^0}{p_e\mu}\right)^{z_i + z_j}. \tag{6-32}$$

The expected sojourn time of a packet in the route can be determined. Since, queue i has the same characteristics as an M/M/1 queue with arrival rate λ^0 / p_e and mean service time $\frac{1}{\mu}$, the mean number of packets (denoted as $\overline{X}_i$)is given by

$$\overline{X}_i = \frac{\lambda^0}{p_e\mu - \lambda^0} \quad for \quad i = 1,2,..., n. \tag{6-33}$$

Thus total expected number of packets in this route is

$$\sum_{i=1}^{n} E[X_i] = \frac{\lambda^0}{p_e\mu - \lambda^0} n. \tag{6-34}$$

Applying Little's formula [162], the expected sojourn time (denoted as $\overline{T}$) of packets from source to destination can be found as

$$\overline{T} = \frac{1}{\lambda_0} \sum_{i=1}^{n} E[X_i] = \frac{n}{p_e\mu - \lambda^0} \tag{6-35}$$

6.8 Summary of Analysis

In this chapter of the thesis, we have has studied the possible SIP session set up options in IMS when an end terminal is mobile. Based on the proposed model a threshold parameter has been derived to select Option 3 in order to achieve better session set up performance over IMS. We have presented an analysis of cost functions in terms of delay in IMS framework. The results show that

- Traditional option (Option 1) is not suitable for higher bit error rate to establish an IMS session in mobile environment.

- Our proposed option (Option 3) provides fastest set up when the session failure rate is null and low.
- Message arrival rate does not affect the session set up delay much as a parameter.
- Increasing channel bandwidth decreases delay for session set up.
- Total cost for session set up in Option 3 goes up sharply if the session fails to set up in first trial.
- Option 3 should be used for higher bandwidth channel with low packet loss rate.
- Option 2 should be used for higher packet loss rate.

In a nutshell, an IMS terminal should send BU in parallel to session set up when the session failure probability is low.

Chapter 7 Queuing Analysis for Instant Messages with Relay Nodes

7.1 Introduction

In this short chapter, we exploit the queuing mechanisms to apply to a special scenario discussed in the IETF draft [213] while instant messages traverse via maximum of two relay nodes. An IMS terminal can select a maximum of two relay nodes to send instant messages via MSRP. The different messages of MSRP (SEND, REPORT, VISIT) has been discussed in the literature review (section 3.1.9). For the large sized SEND messages in IM, MSRP delivers in several SEND messages, where each SEND contains one chunk of the overall message. The relay nodes may have different capacity and service rates. We analyse a special scenario illustrated in Figure 3-9 (section 3.1.9.1) to derive the blocking probability and stability condition in this chapter. We show how the scenario can be reduced for the applicability of queuing theories. The scenario (to be discussed next) can benefit from the analyses provided in this chapter in terms of setting the capacity and service rates of the relay nodes.

7.2 Chunking method of MSRP [91]

Long chunks are interrupted in mid-transmission to ensure fairness across shared transport connections. To support this, MSRP uses a boundary-based framing mechanism. The start line of an MSRP request contains a unique identifier that is also used to indicate the end of the request. Included at the end of the end-line, there is a flag that indicates whether this is the last chunk of data for this message or whether the message will be continued in a subsequent chunk. There is also a Byte-Range header field in the request that indicates that the overall position of this chunk inside the complete message.

For example, the following snippet of two SEND requests demonstrates a message that contains the text "abcdEFGH" being sent as two chunks.

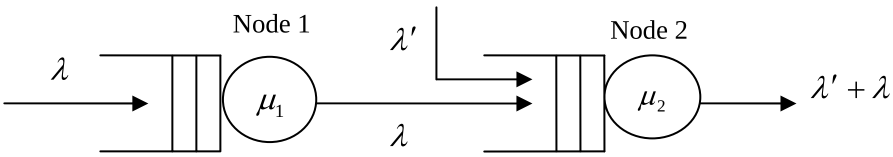
```
MSRP dkei38sd SEND
Message-ID: 4564dpWd
Byte-Range: 1-*/8
Content-Type: text/plain
abcd
-------dkei38sd+


MSRP dkei38ia SEND
Message-ID: 4564dpWd
Byte-Range: 5-8/8
Content-Type: text/plain
EFGH
-------dkei38ia$
```

Figure 7-1: Breaking a Message into Chunks [91]

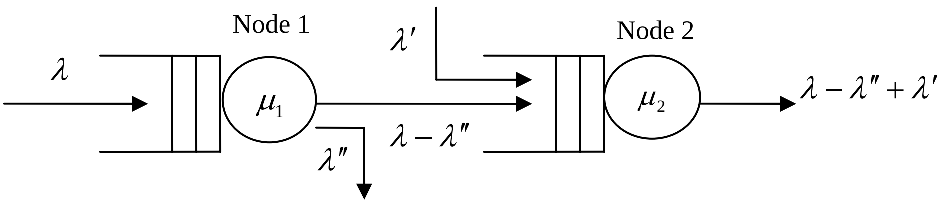
This chunking mechanism allows a sender to interrupt a chunk part of the way through sending it. The ability to interrupt messages allows multiple sessions to share a TCP connection, and for large messages to be sent efficiently while not blocking other messages that share the same connection, or even the same MSRP session. Any chunk that is larger than 2048 octets MUST be interruptible. While MSRP would be simpler to implement if each MSRP session used its own TCP connection, there are compelling reasons to conserve connection. For example, the TCP peer may be a relay device that connects to many other peers. Such a device will scale better if each peer does not create a large number of connections. The chunking mechanism only applies to the SEND method, as it is the only method used to transfer message content [91]. We call the chunking mechanism i.e., breaking one large SEND message into several SEND messages a SEND system. We use the mean size of SEND chunks in relation to the buffer capacity of relay nodes.

7.3 The Special Scenario and Related Work

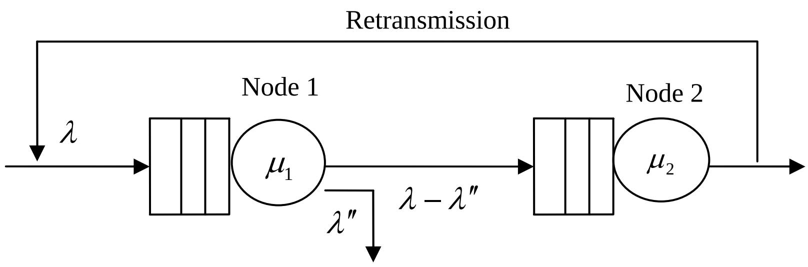
Here we discuss the scenario we consider for SEND systems to analyze. It has been mentioned that a relay node can re-chunk from the received chunks which depends on the service rate of the node. For instance, the first relay node may receive chunks 1 to 3 and then 4 to 7 and so on; upon receiving the chunks it can re-chunk to send 1 to 5 and then 6 to 10 and so on to the second relay node. If the second relay node is short of capacity then the chunks will be blocked and the first relay node will not service and send any more chunks to the second relay node till the second relay node is unblocked. Also, the blocking may impose if the second relay node has a slow outgoing link or the service rate at this node is low (which will eventually lead to a full buffer situation due to heavy load). In a nutshell, this scenario can be described as the first relay node having infinite capacity and the second relay node having finite capacity with arbitrary service rates. By capacity in this chapter, we mean the capacity of the relay nodes in terms of buffer size for accommodation of instant messages. We describe this type of two node system with our assumptions in the next section. Some related work on two node tandem network exists under different assumptions [126, 218, 219]. Kleinrock (1976) was one of the first authors to analyze such system [133, 221]. Chakravarthy (1992) in [219] showed a model for two nodes in series where jobs after getting serviced at the first node wait at the second node before being processed till the group size grows to become a certain given size. This kind of system has application in manufacturing system such as grinding, pinning and cutting etc. However, in instant messing scenario this is not suitable since the waiting before servicing will not provide near real time service. The other cases like additional jobs joining at the second relay node, jobs leaving after service from the first relay node and additional jobs joining at the second relay node are discussed in [126] by Cohen (1982) (see Figure 7-2(a) and Figure 7-2(b)). But, these cases consider infinite buffer in their models. Results of Jackson networks are always applicable if the nodes behave as M/M/1 machines. An optimization of a two node router network containing M/GI/1 finite buffer nodes with retransmission is provided in [218, 222] by Gulpinar *et al* (2007) (see Figure 7-2(c)).



(a)



(b)



(c)

Figure 7-2: Two-stage tandem network

In our scenario we consider only one type of chunks i.e., all the SEND chunks go through two specified relay nodes with one having infinite capacity and the other having finite capacity. For instance, if an IMS terminal sends a 4GB file (including header size) to a destination IMS terminal with explicit selection of two relay nodes using MSRP then, all of the 2097152 chunks (4GB/2048Byte) of the SEND system will go through the two specified relay nodes. The application of such scenario can be thought of a relay node with infinite capacity/buffer located at the home network of the source IMS terminal from the same operator where as the second relay with limited capacity is located outside the home network under different operator having all terminals selecting two relays (to overcome the fading problem). The scenario is a good fit if the destination is located outside the home network with the relay node associated in the visited network having low resources. We analyze the system with different service rates and see how it performs for varying capacity. The earlier version of the work presented in this chapter has been published in [220]. Our assumptions to further narrow down the system for provisions of queuing applications are provided next.

7.4 System Assumptions

As mentioned before that we are considering the first relay node to have infinite capacity and the second relay node to have finite capacity. We consider the situation where all chunks are of or close to the maximum allowed size of MTU. Since the large instant messages will be broken into chunks, these chunks will utilize the maximum allowable size limit and thus will be of almost similar size. Let the second relay node has the finite capacity of size m where m represents the number of the mean sized chunks of a SEND system (Figure 7-3). This means the second relay node can hold at most m number of chunks. Let the service rate at both the relay node be exponential. The chunks from the IMS terminals join the first relay node in a Poisson fashion at the rate of λ. We denote service rates as μ_1, μ_2 at the first and second relay node respectively. We assume chunks are served at each node in a FCFS (First Come First Served) manner. Also, let us assume the first relay node becomes blocked when a chunk completes its service at a time the second relay node is full i.e., blocking after service.

The first relay node remains in blocking state i.e., it cannot serve any other chunk in its queue until a departure occurs from the second relay node. Here, we assume that the chunk propagation delay between the nodes is negligible. Also, in this scenario we do not consider chunks to be corrupted for a retransmission. We refer the issues related to retransmission due to corrupted data, additional chunks joining at the second relay etc. as the scope of the future work of this research.

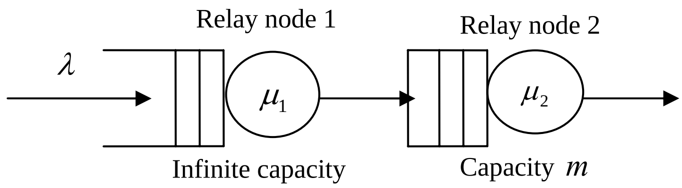


Figure 7-3: SEND system with blocking for 2 relays open queuing

According to the IETF draft of MSRP relay extensions [213], relays only keep transaction states for a short time for each chunk. Delivery over each node should take no more than 32 seconds after the last byte of data is sent. Client applications define their own implementation-dependent timers for end-to-end message delivery. It is assumed that the second relay node provides its blocking status via the REPORT or 200OK message to the first relay node if it is in blocked state or in case its buffer is full or how many more chunk it can accommodate. This may require the 2nd relay node to write at different fields of these messages. The first relay node blocks any chunk if the second relay node is in blocked state.

7.5 Modelling

In this section we first discuss how to achieve the steady state solutions of the scenario under consideration and then later we derive the blocking probability and stability condition for two specific cases of the scenario. The state of the above introduced queuing network can be described by the random variables (n_1, n_2), where n_1 indicates the number of chunks in the first relay node and n_2 indicates number of chunks in the second relay node. Thus we have $n_1 = 0,1,...,$ $n_2 = 0,1,...m, m+1$. $n_2 = m+1$ indicates that the second relay node is full and thus the first relay node is blocked. So, we denote any state $(i, m+1)$, $i = 1,2,...,$ interpreting as the first relay node in blocking state having i number of chunks in the first relay node. Of these i chunks, one has received its service but it is blocked from entering the second relay node and the remaining $i-1$ chunks are waiting to be served at the first relay node.

Perros (1994, [118]) showed that the derivation of such system is not trivial. We mention the case based example provided by Perros where, $m = 2$ and later generalize for a SEND system. Let $p(n_1, n_2)$ be the steady state probability that the system is in state (n_1, n_2). Thus for $m = 2$, we have the possible states: (0,0), (0,1), (1,0), (2,0), (1,1), (2,1), (0,2), (3,0), (1,2), (2,2), (1,3), (3,1), (2,3), (2,3), (3,3), (3,2). We provide the break down of a few transitions in Figure 7-4 due to the arrivals and services of chunks.

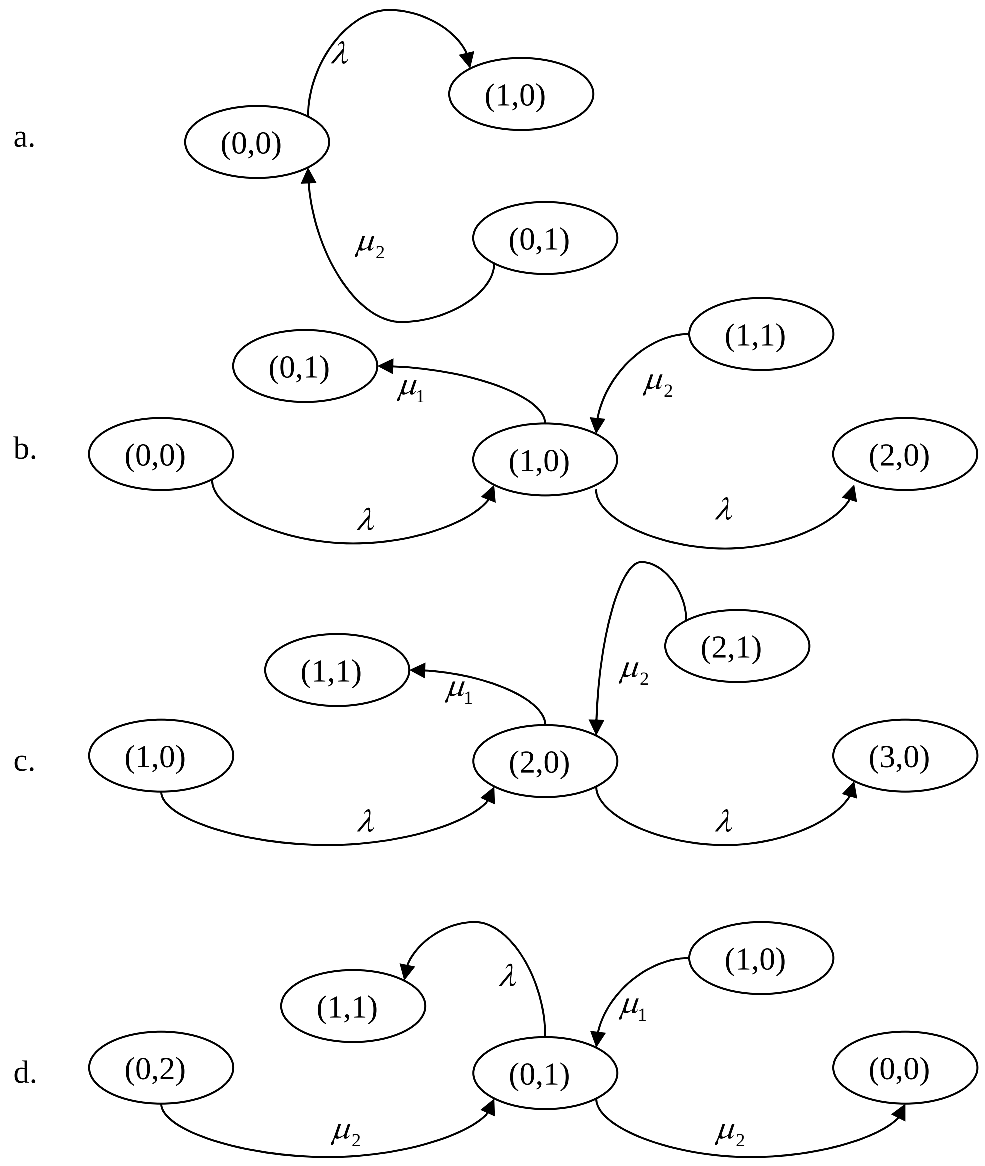


Figure 7-4: State changes of the relay nodes: (a) for state (0, 0), (b) for state (1, 0), (c) for state (2, 0), (d) for state (0, 1)

An arrival at the first relay node will take a SEND system to state (1, 0) from state (0, 0) and a service of a chunk at the second relay node will take the SEND system from state (0, 1) to state (0, 0). This is reflected in Figure 7-4 (a). An arrival at the first relay node will take the SEND system to state (2, 0) from state (1, 0); a service of a chunk at the second relay node will take the SEND system from state (1, 1) to state (1, 0); a service of a chunk at the first relay node will take the system from state (1,0) to state (0, 1) since after service at the first relay node, a chunk enters into the second relay node. This is reflected in Figure 7-4 (b). The other state transitions can be described in similar fashion. Perros showed that the total number of independent equations is 6 (derived from the state transitions) while the total number of unknowns is 7 which make the case of $m = 2$ to be short of one equation. In the case of general *m*, the number of unknowns is $2m + 3$ [118]. This consists of $m + 2$ unknown generating functions and $m + 1$ unknown probabilities $p(0,0)$, $p(0,1)$, $....p(0, m)$. The number of available independent equations is $m + 4$, thus we are $m - 1$ equations short. In view of this, it is not possible to solve the above system of equations for generating functions $g_k(z), k = 1,2,...m + 1$, except when $m = 1$. As mentioned by Perros (1994) in [118], one way to obtain the additional equations required for the solution of the above system is to exploit the fact that a generating function is analytic within the unit circle. In particular let us consider a generating function $g(z)$ and let us assume that it can be written as a ratio of two polynomials, i.e., $g(z) = f_1(z) / f_2(z)$. Now in order for *g(z)* to be analytic within the unit circle, all zeroes of $f_2(z)$ within the unit circle have to be zeroes of $f_1(z)$ as well. Let the zeroes of $f_2(z)$ within the unit circle be $\zeta_1, \zeta_2, ..., \zeta_k$. Then we have that $f_1(\zeta_i) = 0,\ i = 1,2,...k$ which gives us *k* additional equations. This argument has been used successfully to analyse different queuing systems. In general, the zeroes are calculated numerically, though in several cases, one can obtain their exact closed-form expressions. In this way, the steady state equations of a SEND system can be solved.

Now let us examine two opposite cases to the context of our system and see how the queuing systems evolve: (1) when the service rate of the first relay node is infinity and (2) when the first relay node is saturated.

7.5.1 Service rate of the first relay node is infinite

Here, we consider that the first relay node receives an infinitesimal amount of service. If a chunk arrives at the first relay node when the second node contains less than *m* chunks, the chunk goes through the first node and it immediately joins the second relay node. If the second relay node contains *m* chunks, the arriving chunk is immediately blocked and the first node is blocked as well. The first relay node receives the capacity/blocking status of the second relay node via acknowledgment message (as assumed). When a chunk departs from the second node, the first relay node receives the REPORT message and becomes unblocked for an infinitely small amount of time and then it gets blocked again, if there are chunks waiting in the first relay node. If we ignore the propagation time of REPORT or 200OK messages, then the queuing system reduces to M/M/1 queue with traffic intensity, $\rho = \frac{\lambda}{\mu_2}$. With this regard, the steady state probability that there are $n,\ n = 0,1,....,$ chunks in the queuing network is:

$$p(n) = (1-\rho)\rho^n \qquad (7\text{-}1)$$

The probability that there are n_2 chunks in the second relay node is simply given by:

$$p_2(n_2) = (1-\rho)\rho^{n_2}, \quad n_2 = 0,1,...,m-1 \qquad (7\text{-}2)$$

$$p_2(m) = \sum_{n \geq m} (1-\rho)\rho^n = \rho^m \qquad (7\text{-}3)$$

The probability that there are $n_1, n_1 = 1,2,...,$ chunks in the first relay node is

$$p_1(n_1) = (1-\rho)\rho^{m+n_1} \qquad (7\text{-}4)$$

And

$$p_1(0) = \sum_{n=0}^{m} (1-\rho)\rho^n = 1-\rho^{m+1} \qquad (7\text{-}5)$$

As mentioned before, the blocking probability is defined as the probability that a chunk upon service completion at the first relay node will be blocked if the second relay node is full in buffer. This is considered since the first relay node can immediately send a serviced chunk to the second as soon as it receives REPORT message from the second relay node. Also, we consider that the message propagation delay between relay nodes is negligible. With this regard, the waiting of a chunk at the first relay node or at the second relay node is same as a whole and there would be no loss of messages/chunks in a SEND system (since the first relay node will not send a chunk if the second is full). Thus, when the service rate of the first relay node is infinite, the queuing network is simply an M/M/1 queue as discussed above. A chunk will get blocked with probability 1 if upon arrival it finds *m* or more chunks in the M/M/1 queue and vice versa i.e., a chunk will not get blocked if upon arrival of a chunk there are less than *m* chunks. Therefore, we are to define the probability of *m* or more chunks in the queue when the service rate of the first relay node is infinite. Applying the PASTA argument [124, 162], we have the following.

The blocking probability due to the infinite service rate at the first relay node is,

$$\begin{aligned} \beta &= \sum_{n \geq m} p(n) \\ &= (1-\rho)\sum_{n \geq m} \rho^n \qquad (7\text{-}6) \\ &= \rho^m \end{aligned}$$

where, $\rho = \dfrac{\lambda}{\mu_2}$ (7-7)

The throughput of the system (denoted as *U*) can be obtained by direct analysis of the corresponding capacity of the M/M/1 queue as follows:

$$U = \frac{\rho(1-\rho^m)\mu_2}{1-\rho^{m+1}} \qquad (7\text{-}8)$$

The expected response time (denoted as *W*) at the queuing network for a SEND system is M/M/1 response time approximation as:

$$W = \frac{1-(m+1)\rho^{m} + m\rho^{m+1}}{\mu_2(1-\rho)(1-\rho^{m})} \tag{7-9}$$

7.5.2 First relay node is saturated

We now consider the queuing network when the first relay node is saturated. The definition of a node being saturated to our context is when there is always at least one chunk waiting for service [126] i.e., the relay node is never empty. This case can be thought of the first relay node with unlimited supply of chunks from IMS terminals in busy time (opposite case of the previous section). For given λ, μ_2 and m, let μ_1^s be the service rate at the first relay node below for which the first relay node gets saturated. Thus the first relay node is either busy serving chunks or blocked under the following condition:

$$\mu_1 \le \mu_1^s \tag{7-10}$$

A chunk upon completion of its service at the first relay node gets blocked if at that moment the second relay node is full. The blocking chunk remains in front of the first relay node until a departure occurs from the second relay node. At that instant, it moves to the *m*th position of the second relay node and the first node becomes unblocked. Thus, during the blocking period the first node can be seen as providing an additional storage space to the second node since it holds chunks till the second is in not full state and receives the REPORT message from the second relay node. Furthermore, during the blocking period, the first relay node does not serve any other chunks so that no more arrivals occur at the second relay node. With our assumption of exponential arrival and service, it is equivalent to say that arrivals occur at the second relay node at the rate of μ_1 but they are lost during the blocking period. Under this situation, the second relay node behaves as an M/M/1/m+1 queue with an overall arrival rate of μ_1 and service rate of μ_2. Therefore, the probability that there are n_2 chunks in the second relay node is given by:

$$p_2(n_2) = (1-\sigma)\frac{\sigma^{n_2}}{1-\sigma^{m+1}} \tag{7-11}$$

where, $\sigma = \dfrac{\mu_1}{\mu_2}$ (7-12)

This type of two stage model has been studied extensively in the production system [126-128]. We provide derivation of blocking probability and conditions for stability next.

With the above M/M/1/m+1 queue assumption, the effective arrival rate $\bar{\lambda}$ at the second node can be defined as:

$$\bar{\lambda} = \mu_1\left[1 - p_2(m+1)\right] \tag{7-13}$$

where,

$$p_2(m+1) = \frac{(1-\sigma)\sigma^{m+1}}{1-\sigma^{m+2}} \tag{7-14}$$

Applying Little's formula to the (m+1)st position in the queue we can obtain the rate, $\bar{\lambda}\beta$ at which chunks enter the (m+1)st position as follows:

$$\frac{\bar{\lambda}\beta}{\mu_2} = p_2(m+1) \tag{7-15}$$

where, $\dfrac{1}{\mu_2}$ is the mean time a chunk spends in this position and $p_2(m+1)$ is the mean number of chunks occupying this position. From Eq. (7-13) and Eq. (7-15) we get:

$$\mu_1[1 - p_2(m+1)]\frac{\beta}{\mu_2} = p_2(m+1) \tag{7-16}$$

Substituting $\sigma = \dfrac{\mu_1}{\mu_2}$ and $p_2(m+1) = \dfrac{(1-\sigma)\sigma^{m+1}}{1-\sigma^{m+2}}$ we get:

$$\sigma\left(1-\frac{(1-\sigma)\sigma^{m+1}}{1-\sigma^{m+2}}\right)\beta=\frac{(1-\sigma)\sigma^{m+1}}{1-\sigma^{m+2}} \tag{7-17}$$

or

$$\left(1-\sigma^{m+2}-\sigma^{m+1}+\sigma^{m+2}\right)\beta=(1-\sigma)\sigma^{m} \tag{7-18}$$

i.e., the blocking probability that a chunk upon service completion at the first relay node is blocked (at that instant the second relay is full) is:

$$\beta=\frac{(1-\sigma)\sigma^{m}}{1-\sigma^{m+1}} \tag{7-19}$$

It can be seen that β coincides with the time average probability that the second relay node is full when it is analysed as an M/M/1/m queue with traffic intensity, $\sigma=\frac{\mu_1}{\mu_2}$. In that case, the blocking probability defined for M/M/1/m queuing system can be used directly. Applying the result of geometric series in Eq. (7-19) we have (a similar derivation was provided in section 6.3 in chapter 6):

$$\beta=\frac{\sigma^{m}}{1+\sigma+...+\sigma^{m}} \tag{7-20}$$

The quantity $1-\beta$ can be seen as the percentage of time that the first relay node is busy serving chunks. For the special case where $\mu_1=\mu_2$, we have:

$$1-\beta=1-\frac{1^{m}}{1+1+....+1^{m}}=1-\frac{1}{m+1}=\frac{m}{m+1} \tag{7-21}$$

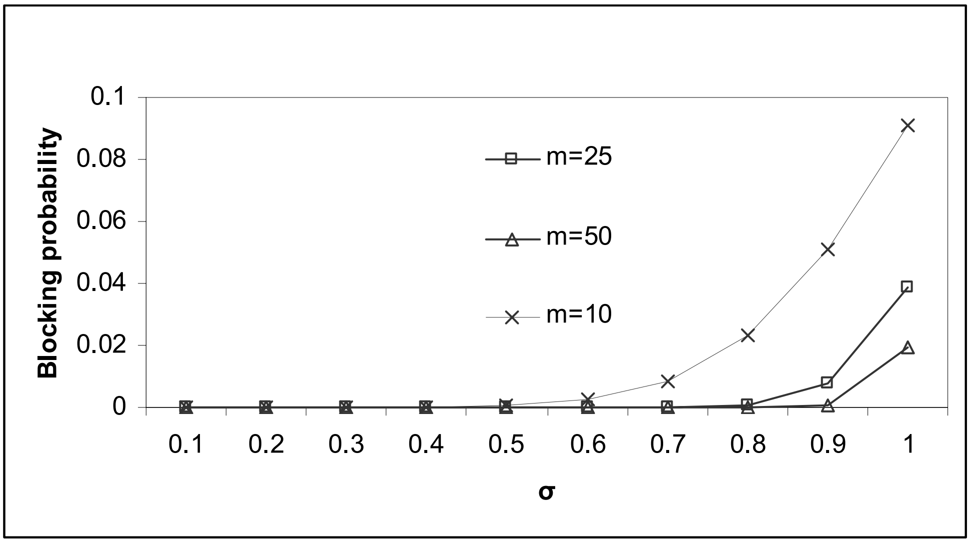


Figure 7-5: Plot of Eq. (7-20)

The plot for blocking probability is provided in Figure 7-5. It can be seen that the higher the capacity of the second relay node, the lesser the blocking probability for the system. Also, when the ratio between the service rates of the relay nodes are low, the blocking probability of the system is low. This means that the blocking of chunks will go up if the service rate of the second relay node is low when the first relay node is saturated.

7.5.2.1 Condition for stability

For given λ, μ_2 and m, the service rate μ_1^s was defined as the critical service rate of the first relay node below for which the first queue becomes saturated/unstable. In order first node to be stable, the effective rate into the second relay node has to be λ. However, for $\mu_1 \le \mu_1^s$, it has been shown in Eq. (7-13) that $\overline{\lambda} = \mu_1[1 - p_2(m+1)]$. Now for, $\mu_1 = \mu_1^s$, we have:

$$\lambda = \overline{\lambda}$$

$$or \qquad (7\text{-}22)$$

$$\lambda = \mu_1^s\left(1 - \frac{(1-\sigma_0)\sigma_0^{m+1}}{1-\sigma_0{}^{m+2}}\right)$$

where, $\sigma_0 = \frac{\mu_1^s}{\mu_2}$. Numerical analysis can be used to compute μ_1^s from Eq. (7-22). The condition for stability of the SEND systems is simply the condition for stability of the first relay node. We observe when the first node is saturated, the maximum departure rate from this node is $\mu_1[1 - p_2(m+1)]$ and that $[1 - p_2(m+1)]$ is the percentage of time that the first relay node is not blocked. Therefore, in order for the first node to be stable we should have as stability condition:

$$\lambda < \mu_1 [1 - p_2(m+1)] \tag{7-23}$$

or

$$\lambda < \mu_1 \left(1 - \frac{(1-\sigma)\sigma^{m+1}}{1-\sigma^{m+2}} \right) \tag{7-24}$$

or

$$\frac{\lambda}{\mu_1} < \frac{1+\sigma+.....+\sigma^m}{1+\sigma+...+\sigma^{m+1}} \tag{7-25}$$

The amount of load at the first relay node that will make the system unstable can be evaluated from the above equation. The plot for Eq. (7-25) is provided in Figure 7-6. We observe that for higher capacity at the second relay node, the system is stable for higher load. Also, when the ratio between the two service rates is low i.e., the service rate for the second relay node is much higher than the first relay node then, the system is stable even for much higher load. But if the two service rates are almost equal, then the stability condition starts dropping quickly for low capacity.

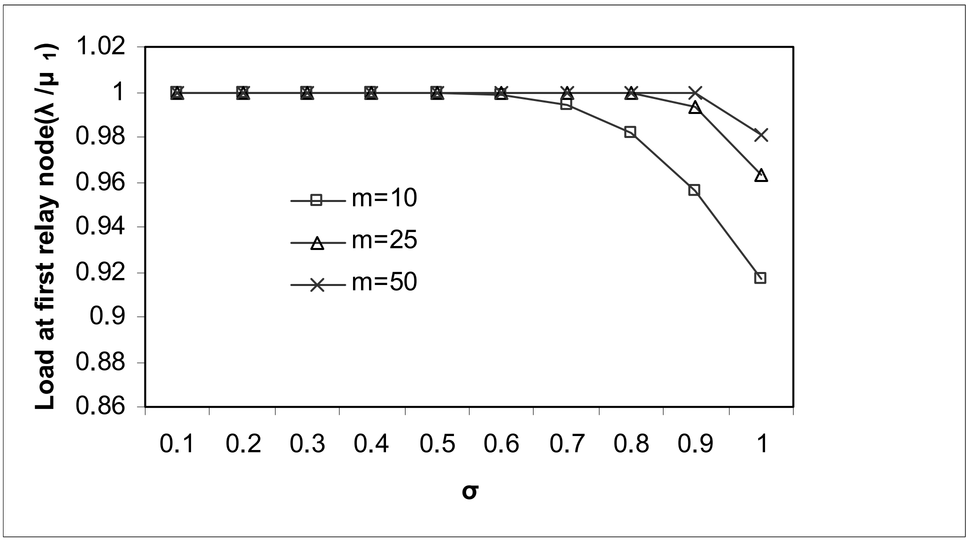


Figure 7-6: Plot for stability condition

We considered the capacity of the system in terms of mean chunk-size which makes the analysis easier to relate to the number of chunks accommodation into the system. The expressions provided in Eq. (7-6), Eq. (7-20), Eq. (7-22) and Eq. (7-25) can be used to synchronise a SEND system if the first relay node has infinite buffer and the second relay node has finite capacity. Using these expressions the blocking probability can be computed and the stability conditions can be upper bounded under given arrival and service rates. We believe a SEND system can benefit applying the above calculation where chunks for instant messages has to go through two relay nodes having infinite and finite capacity respectively before reaching the destination.

7.6 Summary

We have presented the analysis of a special queuing system including two relay nodes for instant messages. The special case we consider here i.e., the first relay node having infinite capacity and the second relay node having finite capacity is a typical case discussed in the IETF draft of relay extensions on MSRP. We showed with two different cases when the queuing network behaves like M/M/1 and when like

M/M/1/m+1 queue. If the service rate of the first relay node is infinite, then the queuing network under consideration reduces to M/M/1 whereas when the first relay node is saturated, then the system reduces to M/M/1/m+1 queue. In the first case, the traffic intensity depends on λ and μ_2 i.e., $\left(\rho = \frac{\lambda}{\mu_2}\right)$ whereas in the second case it depends on the service rate of the two relay nodes i.e., $\left(\sigma = \frac{\mu_1}{\mu_2}\right)$. We derived the blocking probabilities for both cases and the stability condition for the system. We showed that the ratio of the service rates of relay nodes and buffer size have direct impact to the blocking probability of the system. Based on the expressions derived in this chapter, the computation becomes straightforward for a SEND system with two relay nodes. The service rate of relay nodes can be adjusted using the analysis provided.

Chapter 8 Conclusions and Future Work

Our work in this thesis is a pioneer progress to improve IMS services for future network. The parameter-values used in our work can be modified accordingly to achieve desired performance.

- We have introduced a scheduling scheme for the presence server based on message arrival rate and associated watchers. Messages are dropped if the inter arrival time is less than a derived threshold time. The derivation of threshold time depends on the server channel allocation, traffic intensity, number of presentities publishing and message arrival rate. Although availability of ample channel bandwidth and server speed will reduce the load at the presence server, the IMS terminals which are low in capacity will still have to process abundant data during heavy traffic in the presence system. Our message dropping mechanism not only reduces load at the presence server, but also saves handful number of messages to be generated by the presence server and consequently not to be processed by the IMS end terminals. We provided a complete set of admission control methods centred through our introduced weighted class based queuing (WCBQ) scheme. We have shown the effectiveness of WCBQ by performance analysis with regards to blocking probability and message generation. The dropping of messages improves server performance and decreases message generation cost. We believe the WCBQ scheduling and late message dropping technique can be applied to any publish/subscribe based service as a generic solution. We also proposed a theoretical algorithm to optimize the watcher subscription time. The overhead consumption has been studied as well for constant subscription time in the IMS presence service.

- We further explained how to dimension a PoC service with regards to fixed grade of service (GoS). The available resources need to be utilized optimally in order to achieve enhanced performance with a fixed grade of service. We analysed how to restrict long PoC sessions while number of Transmit/Receive units at the base station is limited. This reduces message flows to some extend in the system. Methods to reduce traffic overflow is also studied. Moreover, the optimal timer and number of simultaneous sessions for long PoC sessions in busy hour have been derived in this thesis. These derivations are indeed useful for a service provider that is short in resources at the base station.
- The other area we explored is the IMS session set up scenario in mobile environment. The previous work is mainly centred at the end-to-end quality of service of IMS session after it is set up. The SIP interaction with MIP gains useful outcome in mobility management. In this thesis, we showed how delay can be reduced by sending the binding update message in parallel while a session is being set up if the end IMS terminals are mobile. The scenario has been studied by simulating an IMS session set up prototype. Results suggest that for early successful session ups, sending binding update in parallel will exert the most performance.
- Moreover, we showed how an IM system evolves when the SEND chunks go through two consecutive relay nodes. With the applications of queuing theories, we defined a few important parameters as blocking

probability, stability condition, system throughput and response time. The buffer capacity of the relay nodes was represented in terms of number of SEND chunks. The derived expressions are useful outcome for setting service rates of the relay nodes while the first having infinite capacity and the second having finite buffer.

Future work: The method we provided for optimizing watcher subscription time in section 4.7 needs to be tested with overhead consumption. The work presented in chapter 4 focuses on single server only. There are other ways to improve presence service for instance, the overload problem can be addressed at the distributed computing layer i.e., distribution of presence servers. Performance measure in such situation can be analysed via distributed hashing or peer-to-peer information sharing etc. for the presence look up and distribution.

The PoC service is yet to undergo further refinement to over come all its shortcomings. New algorithm for efficient dimensioning is required to reduce congestion in the network. If PoC is to carry real-time data over packet switching networks, the Internet standard protocol real time protocol (RTP) can be used. In that case low latency and in sequence delivery need to be guaranteed by some algorithm that would synchronize the clock that time-stamps the packets. The PoC signalling in circuit switched networks is limited to SMS (short messaging service) only. In order to use SMS for Push-to-talk, signalling will increase the server load and consume more resources. Thus to use SMS capacity for PoC signalling is expensive as the traditional use brings direct revenues that are now used for PoC. The cost for this signalling should be compared to the price of sending SMS messages for the end-user.

The voice quality, presence functionality etc. need to be tested under IMS technologies. The different codec facilities for instance, AMR (Adaptive multi rate voice codec), EFR (Enhanced full rate) etc. are to be studied both in error free and error prone states.

New models could be derived based on the cost incurred by the radio access network to implement PoC service in IMS. Some necessary cost items are radio network planning activities, transition network, core network planning, number of application servers used, service integration and marketing activities etc. Efficient scheduling is always preferable below the MAC (media access control) layer for the PoC sessions to share the time-slots of the server. The future releases of the OMA and IMS should refer to the abovementioned areas at least.

We have discussed the applicability of queuing theories into IMS IM service to define blocking probability etc. However, the scenario in chapter 7 did not consider chunk retransmission due to corrupted data and chunk propagation delay between relays. Also, chunks may leave after service at the first relay and, similarly new chunks may join at second relay. These cases need to be explored as well. Another important area in IM service is to dimension a chat server (i.e., the Media Resource Function Processor or MRFP) that provides multi-party session based conferences for instant messages. Similar optimization aspects need to be addressed for an IM chat server as of PoC services (as derived in chapter 5).

It is well known that, when new technology is introduced into the telecommunication industry, it is logistically impossible to deploy it everywhere at the same time. Yet, plenty of areas need to be explored to shape IMS completely. The current 3GPP specifications do not adequately address how online session charging may be accomplished in IMS networks. Several 3GPP technical specifications describe online charging in IMS networks [100–102]. For example, the 3GPP TS 32.240 and TS 32.260 standards describe an online charging server (OCS) with a session charging function [101-102]. The OCS is coupled to a call session control function (CSCF) through an IMS service control (ISC) interface. The CSCF can control a call session for either a calling party or a called party, and it needs to communicate with the OCS over the ISC interface to provide online charging for the call session. However, while as a service interface the ISC defines a reference point between a serving call session control function (S-CSCF) and an application server for the Session Initiation Protocol (SIP)-based session signalling control, it does not provide session credit authorization and real-time credit control. The ISC interface cannot support online charging. Therefore, in order to use the ISC interface between the CSCF and the OCS for online charging, additional functionality needs to be added to the OCS [101-102]. A complete policy-based IMS call control system that will fill the online charging gaps in the 3GPP standards is essential.

P-CSCF discovery is the procedure by which an IMS terminal obtains the IP address of a P-CSCF. This is the P-CSCF that acts as an outbound/inbound SIP proxy server toward the IMS terminal (i.e. all the SIP signalling sent by or destined for the IMS terminal traverses the P-CSCF). P-CSCF discovery may take place in two different ways:

(a) Integrated into the procedure that gives access to the IP-CAN (IP Connectivity Access Network).

(b) As a stand-alone procedure.

The integrated version of P-CSCF discovery depends on the type of IP Connectivity Access Network. If IP-CAN is a GPRS (General Packet Radio Service) network, once the GPRS attach procedures are completed the terminal is authorized to use the GPRS network. Then the IMS terminal does a so-called Activate PDP (Policy Decision Point) Context Procedure. The main goal of the procedure is to configure the IMS terminal with an IPv6 address, but in this case the IMS terminal also discovers the IPv6 address of the P-CSCF to which to send SIP requests.

The stand-alone version of the P-CSCF discovery is based on the use of DHCPv6 (Dynamic Host Configuration Protocol for IPv6) specified in RFC 3315 [58] and DNS (Domain Name System, specified in RFC 1034 [59]). A suitable procedure is required in order to identify the faster mechanism to discover P-CSCF in IMS. The performance of network access connectivity in IMS was never analysed before. An efficient algorithm is needed for an IMS terminal to promptly select the faster method in order to achieve the IP address of a P-CSCF to send SIP messages.

Once the terminal has got connectivity to the IP-CAN the IMS terminal sends a DHCPv6 information request where it requests the DHCPv6 Options for SIP servers (specified in [60]). In the case of the IMS the P-CSCF performs the role of an outbound/inbound SIP proxy server, so, the DHCP server returns a DHCP Reply message that contains one or more domain names and/or IP addresses of one or more P-CSCFs. Eventually, the IMS terminal discovers the IP address of its P-CSCF and can send SIP signalling to its allocated P-CSCF. However, these procedures do not mention which one is efficient in which environment. For example, if P-CSCF is located in the home network and the CN (IMS terminal) in the visited network, which would be the quicker method for P-CSCF discovery!

Concisely, the necessary changes that IMS needs to undergo are:

- Cost measurement of PoC signalling under SMS capacity in the circuit-switching networks.
- Evaluation/detection of early/faster P-CSCF discovery method during the process of receiving IP address for an IMS terminal.
- Dimensioning of IM chat servers.
- Robust rule/policy based call control system for online charging servers (OCS).

Besides, there are other potential areas that need to be addressed. IMS provides SIP-PSTN Inter-working services. The traditional audio calls in SIP-PSTN inter-working focuses on audio-only calls. Although, the video services are included in the inter-working, significant work is required in the encoding and gateway architecture to facilitate video streaming in the IMS. The future releases of the IMS should refer to the abovementioned areas at least. One of the reasons for creating the IMS was to provide the Quality of Service (QoS) required for enjoying, rather than suffering, real time multimedia sessions. Thus its drawbacks are essential to be overcome in near future.

Appendix A Steady State of BCMP Model

This appendix has been referred in section 4.4.1.

We state the following so-called BCMP result from [203] introduced by Baskett, K. M. Chandy, R. R. Muntz and F. G. Palacios (1975).

For a BCMP network with K nodes and R classes of customers, which is open, closed or mixed in which each node is of type FCFS or IS (infinite sever), the equilibrium state probabilities are given by

$$\pi(\underline{n}) = \frac{d(\underline{n})}{G}\prod_{i=1}^{K} f_i(\underline{n}_i) \tag{A.1}$$

The above formula holds for any state $\underline{n} = (\underline{n}_1,...,\underline{n}_K)$ in the state-space S (that depends on the network under consideration) with $\underline{n}_i = (n_{i1},...,n_{iR})$, where n_{ir} is the number of customers of class r in node i. Moreover (with $|\underline{n}_i| == \sum_{r=1}^{R} n_{ir}$ for $i=1,2,...,K$),

If node i is of type FCFS then,

$$f_i(\underline{n}_i) = |\underline{n}_i|!\prod_{j=1}^{|\underline{n}_i|}\frac{1}{\alpha_i(j)}\prod_{r=1}^{R}\frac{\rho_{ir}^{n_{ir}}}{n_{ir}!} \tag{A.2}$$

If node i is of type IS then,

$$f_i(\underline{n}_i) = \prod_{r=1}^{R}\frac{\rho_{ir}^{n_{ir}}}{n_{ir}!} \tag{A.3}$$

In Eq. (A.1), $G < \infty$ is the normalizing constant chosen such that

$\sum_{\underline{n}\in S}\pi(\underline{n}) = 1$, $d(\underline{n}) = \prod_{j=0}^{M(\underline{n})-1}\gamma(j)$ if the arrivals in the system depend on the total number of customers $M(\underline{n}) = \sum_{i=1}^{K}|\underline{n}_i|$ when the system is in state $\underline{n}$, and $d(\underline{n}) = 1$ if the network is closed.

Appendix B Effective Bandwidth of a Flow in WCBQ

This appendix has been referred in section 4.4.4.

We define, $\phi(\theta) = E[\exp(\theta(s_n - \tau_n)]$, The Laplace transformation of the random variable $s_n - \tau_n$, where s_n is the service time of the *nth* message and τ_n is the time between arrivals of messages *n* and *n+1* for the GI/GI/1 system.

In a GI/GI/1 queue we assume that:

1. $(s_n)_n$ is a sequence of independent random variables with the common cumulative distribution function (c.d.f.) *G(x)*, namely, $P(s_n \leq x) = G(x)$ for all $n \geq 1, x \geq 0$;

2. $(\tau_n)_n$ is a sequence of independent random variables with the common c.d.f. *F(x)*, namely, $P(\tau_n \leq x) = F(x) = 1 - \exp(-\lambda x)$ for all $n \geq 1, x \geq 0$. λ has been defined before with the superposition of all independent Poisson processes (in chapter 4).

We also assume that there exists $m > 0$ such that $\phi(m) < \infty$. From Kingman's result [195] we know that if $\theta > 0$ such that $\phi(\theta) \leq 1$, then

$$P(W_n \geq x) \leq e^{-\theta x} \qquad \forall x > 0, n \geq 1 \tag{B.1}$$

and $$P(W \geq x) \leq e^{-\theta x} \quad \forall x > 0 \tag{B.2}$$

where, W_n is the waiting time in queue of the *nth* message.

The above results may be used in our multiclass M/G/1 WCBQ if we can reduce our queue to G/G/1 queue.

With the probability ${}^{\lambda_k}\!/_{\lambda}$, the *nth* message will be a message of class *k,* Let X_k be the time that elapses between the arrival of the *nth* message and the first arrival of a message of class *k.* Since the arrival process of each class is Poisson, and therefore memoryless, we know that X_k is distributed according to an exponential random variable with rate λ_k.

Therefore,

P((n+1)st message is of class k) = $P(X_k < \min_{j \neq k} X_j))$

$$= \int_0^\infty P(x < \min_{j \neq k} X_j)\lambda_k e^{-\lambda_k x} dx$$

$$= \int_0^\infty \prod_{j \neq k} P(x < X_j)\lambda_k e^{-\lambda_k x} dx \qquad \text{(B.3)}$$

$$= \lambda_k \int_0^\infty \prod_j e^{-\lambda_j x} dx$$

$$= \lambda_k \int_0^\infty e^{-\lambda x} dx = \frac{\lambda_k}{\lambda}$$

Let us now determine the c.d.f. *G(x)* of the service time of an arbitrary message.

G(x)=P (*service time of new message* $\leq x$)

$$= \sum_k P\,(\textit{service time of new message} \leq x \textit{ and new message of type } k)$$

$$= \sum_k P\ \ (\textit{service time of new message} \leq x \mid \textit{new message of type } k)\frac{\lambda_k}{\lambda}$$

From the above and from the law of total probability and Bayes' formula,

$$G(x) = \sum_k \frac{\lambda_k}{\lambda} G_k(x) \qquad \text{(B.4)}$$

In particular, the mean service time $1/\mu$ of our multiclass M/G/1 WCBQ is given by

$$\frac{1}{\mu} = \int_0^\infty x dG(x) = \sum_k \frac{\lambda_k}{\lambda} \int_0^\infty x dG_k(x) = \sum_k \frac{\rho_k}{\lambda} \qquad \text{(B.5)}$$

with $\rho_k = \dfrac{\lambda_k}{\mu_k}$.

Thus, for stability condition of $\sum_k \rho_k < 1$,

$$\phi(\theta) = E[e^{\theta(s_n - \tau_n)}]$$

$$= E[e^{\theta s_n}]E[e^{-\theta\tau_n}] \tag{B.6}$$

$$= \left(\frac{\lambda}{\lambda+\theta}\right)E[e^{\theta s_n}]$$

Now,

$$E[e^{\theta s_n}] = \int_0^\infty e^{\theta y}\,dG(y) = \sum_k \frac{\lambda_k}{\lambda}\int_0^\infty e^{\theta y}\,dG_k(y) \tag{B.7}$$

Therefore,

$$\phi(\theta) = \sum_k \frac{\lambda_k}{\lambda+\theta}\int_0^\infty e^{\theta y}\,dG_k(y) \tag{B.8}$$

Thus, $P(W \ge a) \le c$ if $\phi(-(\log c)/a) \le 1$ which is of the form

$$\sum_k \frac{\lambda_k}{\lambda-(\log c)/a}\int_0^\infty e^{-(\log c)y/a}\,dG_k(y) \le 1. \tag{B.9}$$

Applying the additive property of independent Poisson process on the arrival rates, Eq. (B.9) can be rewritten as Eq. (4-34) of chapter 4.

Appendix C M/M/m Queuing System

This appendix has been referred in section 5.4.

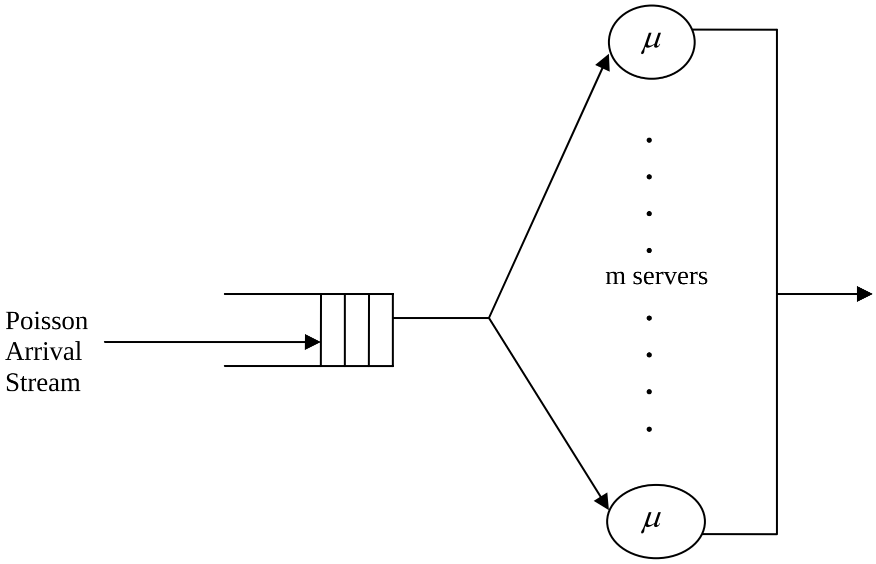


Figure C-1: M/M/m queuing system [162]

The queuing system with Poisson arrival rate λ, number of servers $m \geq 1$, and mean service rate of μ, each sharing a common queue (see Figure C-1) can be illustrated as the birth-death model with the rates:

$$\lambda_k = \lambda, \qquad k = 0,1,2...$$
$$\mu_k = \begin{cases} k\mu, & 0 \leq k < m. \\ m\mu, & m \leq k, \end{cases} \tag{C.1}$$

The state diagram of this system is shown in Figure C-2. The steady state probabilities are given by [162]:

$$p_k = p_0 (\frac{\lambda}{\mu})^k \frac{1}{k!}, \qquad k < m,$$
$$p_k = p_0 (\frac{\lambda}{\mu})^k \frac{1}{m!\, m^{k-m}}, \quad k \geq m, \tag{C.2}$$

Defining $\rho = \frac{\lambda}{m\mu}$, the condition for stability is given by $\rho < 1$. Using the fact that $\sum_{k=0}^{\infty} p_k = 1$, the expression for p_0 is obtained as follows [162]:

$$p_0 = [\sum_{k=0}^{m-1} \frac{(m\rho)^k}{k!} + \frac{(m\rho)^m}{m!} \frac{1}{1-\rho}]^{-1} \tag{C.3}$$

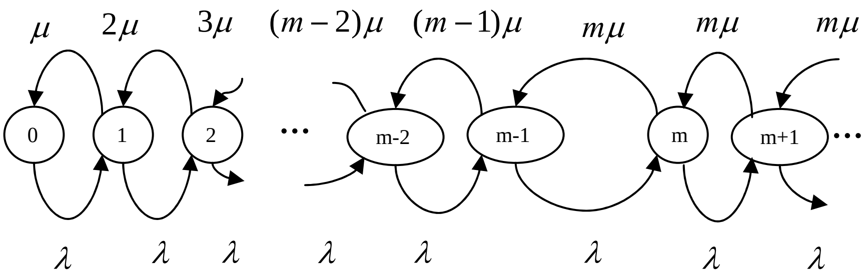


Figure C-2: The state diagram of the M/M/m queue [162]

Appendix D Load Sharing Expression

This appendix has been referred in section 5.5.

The load sharing optimal Eq. (5-9) of chapter 5 becomes (from [183]) :

$$
\begin{aligned}
L(A,u,v) = & \sum_{l,n} a_{l,n} B_{l,n} + \sum_{n,m} a_{n,m} B_{n,m} \\
& + \sum_{l,m} v^{l,m} \left(\sum_{n} A_n^{l,m} - A^{l,m} \right) - \sum_{l,m,n} u_n^{l,m} A_n^{l,m}
\end{aligned} \tag{D.1}
$$

Taking the derivative with respect to $A_k^{i,j}$, we get

$$
\begin{aligned}
u_k^{i,j} = & v^{i,j} + \sum_{l,n} \frac{\partial a_{l,n}}{\partial A_k^{i,j}} \left[B_{l,n} + a_{l,n} \frac{\partial B_{l,n}}{\partial a_{l,n}} \right] \\
& + \sum_{n,m} \frac{\partial a_{n,m}}{\partial A_k^{i,j}} \left[B_{n,m} + a_{n,m} \frac{\partial B_{n,m}}{\partial a_{n,m}} \right]
\end{aligned} \tag{D.2}
$$

Using the fact that

$$
\frac{\partial a_{l,m}}{\partial A_k^{i,j}} = \sum_{m} \frac{\partial A^{l,m}}{\partial A_k^{i,j}} = \delta_{i,l} \delta_{k,n} \tag{D.3}
$$

and

$$
\begin{aligned}
\frac{\partial a_{n,m}}{\partial A_k^{i,j}} & = \sum_{l} \frac{\partial A^{l,m} (1 - B_{l,n})}{\partial A_k^{i,j}} \\
& = \sum_{l} \left[(1 - B_{l,n}) \delta_{j,m} \delta_{n,k} \delta_{i,l} - A_n^{l,m} \frac{\partial B_{l,n}}{\partial a_{l,n}} \frac{\partial a_{l,n}}{\partial A_k^{i,j}} \right] \\
& = (1 - B_{i,n}) \delta_{j,m} \delta_{k,n} - A_n^{i,m} \frac{\partial B_{i,n}}{\partial a_{i,n}} \delta_{k,n.}
\end{aligned} \tag{D.4}
$$

where, $\delta_{i,j}$ = The Kronecker symbol = 1 if i=j, and 0 otherwise

Replacing these derivatives in Eq. (D.2) and doing the appropriate sums, we get the optimal expression in Eq. (5-17) of chapter 5.

Appendix E Poisson Inter-Arrival Time & Density Function

This appendix has been referred in section 5.3.

By the definition, in the Poisson model, the inter-arrival time of data units follows the negative exponential distribution (NED), i.e., the probability density function (PDF) is

$$f(t) = \lambda e^{-\lambda t} \quad (t > 0) \tag{E.1}$$

And the cumulative distribution function (CDF) is

$$F(t) = 1 - e^{-\lambda t} \quad (t > 0) \tag{E.2}$$

From Eq.(E.1) and Eq.(E.2), it can be derived that the inter-arrival time of session follows the negative exponential distribution (NED), then the probability density function (PDF) with arrival rate λ takes the form: $p(w) = \frac{\lambda}{w^2} e^{-\lambda/w}$; and the corresponding cumulative distribution function is: $C(w) = e^{-\lambda/w}$.

The mathematical detail of the derivation is furnished below. Let X be a continuous random variable with the probability density function $f(x)$ such that $f(x) \neq 0$ in interval [a,b]. Let $y = g(x)$ be a real function differentiable everywhere. If $g'(x)$ does not change its sign in [a,b], then $Y = g(X)$ is a continuous random variable with the probability density function:

$$\psi(y) = \begin{cases} f\big(h(y)\big)\,|\,h'(y)\,| & if \quad \alpha < y < \beta \\ 0 & otherwise \end{cases} \tag{E.3}$$

where *h(y)* is the inverse function of *g(x)*, and

$$\begin{aligned} \alpha &= \min\{g(a), g(b)\} \\ \beta &= \max\{g(a), g(b)\} \end{aligned} \tag{E.4}$$

If $y = g(x) = 1/x$, then the process connects both the inter-arrival time and inter-arrival rate. As a result, we have

$$\begin{aligned} & x = h(y) = 1/y \\ \textit{and} \quad & h'(y) = -1/y^2 \end{aligned} \tag{E.5}$$

Thus for probability density function of Eq. (E.1), i.e., for the inter-arrival time following the negative exponential distribution, the PDF of the inter-arrival rate would be

$$\psi(y) = \frac{1}{y^2} f\left(\frac{1}{y}\right) = \frac{\lambda}{y^2} e^{-\lambda/y} \qquad (y > 0) \tag{E.6}$$

And the corresponding cumulative distribution function (CDF) is

$$C(y) = e^{-\lambda/y} \qquad (y > 0) \tag{E.7}$$

Bibliography

[1] M. T. Alam, Z. D. Wu, “Dimensioning and Optimization of Push-to-Talk over Cellular Server“, *International Journal of Network Management,* John Wiley & Sons Ltd. (In press), DOI: 10.1002/nem.652, 2007.

[2] M. T. Alam, Z. D. Wu, "On Analysing Cost for Optimizing the Watcher Subscription Time in the IMS Presence Service", *Engineering Letters,* Vol: 13(1), 2006, pp: 1-10.

[3] C. Perkins, "IP mobility support for IPv4," RFC 3344, *Internet Engineering Task Force,* 2002.

[4] C. Perkins and D. Johnson, "Route optimization in mobile IP," *Internet Draft, Internet Engineering Task Force*. Work in progress, 2000.

[5] C. Perkins, "Minimal encapsulation within IP," RFC 2004, *Internet Engineering Task Force*, 1996.

[6] A. Neumann, X. Fu, H. Karl, “Prototype Implementation and Performance Evaluation of a QoS-Conditionalized Handoff Scheme for Mobile IPv6 Networks”. *Computer Communications, Proceedings.* IEEE 18th Annual Workshop on, 20-21 Oct 2003, pp: 24 – 29.

[7] A. Ringapin, J. Ben-Othman, P. Urien, “Quality Of Service and Mobility in IP Network.” *The 56th IEEE Proceedings on Vehicular Technology Conference*, Vol: 4, 24-28 Sep 2002, pp: 2430 – 2434.

[8] I. Vivaldi, M. Habaebi, B. Ali, V. Prakesh, “Fast handover algorithm for hierarchical mobile IPv6 macro-mobility management.” *The 9th Asia-Pacific Conference on Communications,* Vol: 2, 21-24 Sep 2003, pp: 630 – 634.

[9] Q. Zhou; H. Yu;D. Makrakis, O. Kabranov, L. Orozco-Barbosa, ‘‘Performance evaluation for real-time traffic in differentiated services capable IP network”. *Electrical*

and Computer Engineering, IEEE CCECE, Canadian Conference on, Vol: 3, 12-15 May 2002, pp: 1522 – 1526.

[10] K. Yusof, N. Fisal, "Provisioning QoS in Differentiated Service Domain for MIPv6". *Telecommunication Technology, NCTT Proceedings.* IEEE 4th National Conference on, 14-15 Jan. 2003, pp: 98 – 101.

[11] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, E. Schooler, "SIP: Session Initiation Protocol." *Internet Engineering Task Force, RFC 3261,* 2002.

[12] D. B. Johnson, C. E. Perkins, J. Arkko, "Mobility Support in IPv6". *IETF Mobile IP Working Group,* RFC 3775, 2004.

[13] Shou-Chih Lo, G. Lee, Wen-Tsuen Chen, Jen-Chi Liu, "Architecture for Mobility and QoS Support in All-IP Wireless Networks". *IEEE Journal on selected areas in communications,* Vol: 22 (4), 2004, pp: 691-705.

[14] C. Shan, "A new scalable and efficient packet scheduling method in high-speed packet switch networks". *High Performance Switching and Routing,* IEEE Workshop on , 29-31 May 2001, pp: 16 – 20.

[15] S. Hwang, B. Lee, Y. Han; C. Hwang, "An adaptive hierarchical mobile IPv6 with route optimization." *Vehicular Technology Conference,* The 57th IEEE Semi-annual , Vol: 3, 22-25 Apr 2003, pp: 1502 – 1506.

[16] N. Montavont, T. Noel, "Analysis and Evaluation of Mobile IPv6 Handovers over Wireless LAN". *ACM Mobile Networks and Applications.* Vol: 8, 2003, pp: 643-653.

[17] A. Garyfalos, K. Almeroth, J. Finney, "A Comparison of Network and Application Layer Multicast for Mobile IPv6 Networks". *ACM Proceedings of the 6th international workshop on Modelling analysis and simulation of wireless and mobile systems.* 2003, pp: 58-66.

[18] H. Takatama, H.Tan, "Intelligent SIP system for mobile Internet" *IEEE Intelligent Network Workshop,* 6-9 May 2001, pp: 83 – 93.

[19] Chun-Ting Chou, K. Shin, **"**Analysis of adaptive bandwidth allocation in wireless networks with multilevel degradable quality of service". *Mobile Computing,* IEEE Transactions on, Vol: 3 (1), Jan-Mar 2004, pp: 5 – 17.

[20] M. Bozinovski, L. Gavrilovska, R. Prasad, "Performance evaluation of a SIP-based state-sharing mechanism". *Vehicular Technology Conference, Proceedings.* IEEE 56th , Vol: 4, 24-28 Sep 2002, pp: 2041 – 2045.

[21] G. Hasegawa, T. Matsuo, M. Murata, H. Miyahara, "Comparisons of packet scheduling algorithms for fair service among connections on the Internet" *INFOCOM,* Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies, Proceedings. Vol: 3, 26-30 Mar 2000, pp: 1253 – 1262.

[22] Z. Zhanjun, H. Chengde, Y. Xueliang, "The guarantee of QoS for wireless multimedia streams based on adaptive session". *Personal Wireless Communications,* IEEE International Conference on, 17-20 Dec 2000, pp: 283 – 287.

[23] B. Moon, H. Aghvam, "RSVP extensions for real-time services in wireless mobile networks". *Communications Magazine,* IEEE , Vol: 39 (12), Dec 2001, pp: 52 – 59.

[24] 3GPP, TSG SSA, IP Multimedia Subsystem (IMS) – Stage 2 (Release 7), TS 23.228 v.7.3.0, 2006-03.

[25] 3GPP, Technical Specification Group Services and System Aspects, IP Multimedia Subsystem (IMS) - Stage 2 (Release 5), TS 23.228 v5.6.0, 2002-09.

[26] 3GPP, Technical Specification Group Services and System Aspects: QoS Concept and Architecture (Release 5), TS 23.207 v5.6.0, 2002-09.

[27] 3GPP, Technical Specification Group Core Network; Signalling flows for the IP multimedia call control based on SIP and SDP; Stage 3 (Release 5), TS 24.228 v5.2.0, 2002-09.

[28] E. Wedlund and H. Schulzrinne, "Mobility support using SIP," in *Second ACM/IEEE International Conference on Wireless and Mobile Multimedia (WoWMoM'99),* Seattle, Washington, 1999, pp: 76-82.

[29] H. Schulzrine, E. Wedlund, "Application-layer mobility using SIP". *Mobile Computing and Communications Review,* Vol: 4 (3), 2000, pp: 47-57.

[30] M. Moh, G. Berquin, Y Chen, "Mobile IP telephony: mobility support of SIP". *IEEE Computer Communications and Networks, Proceedings.* Eight International Conference, 1999, pp: 554 – 559.

[31] G. Stojsic, R. Radovic, S. Srbljic, "Formal definition of SIP proxy behaviour." *IEEE International Conference on Trends in Communications,* Vol: 2, 4-7 Jul 2001, pp: 289 – 292.

[32] Z. J. Haas, Y. Lin, **"**On optimizing the location update costs in the presence of database failures." *ACM Wireless Networks,* Vol: 4(5), Aug 1998, pp: 419 – 426.

[33] A. Pang, Y. Lin, Y. Fang, "Implicit deregistration with forced registration for PCS mobility management." *ACM Wireless Networks,* Vol: 7(1), 2001, pp: 99 – 104.

[34] T. Choi, J. Nah, J. Song, L. Kim, **"**Combinatorial mobile IP: a new efficient mobility management using minimized paging and local registration in mobile IP environments." *ACM Wireless Networks,* Vol: 10(3), May 2004, pp: 311 – 321.

[35] Y. Birk, Y. Nachman, "Using direction and elapsed-time information to reduce the wireless cost of locating mobile units in cellular networks." *ACM Wireless Networks***,** Vol: 1(4), 1995, pp: 403 – 412.

[36] I. Dalgic, M. Borella, R. Dean, J. Grabiec, J. Mahler, G. Schuster, I. Sidhu, "True number portability and advanced call screening in a SIP-based IP telephony system." *Communications Magazine, IEEE*, Vol: 37(7), Jul 1999, pp: 96 – 101.

[37] J. Wu, P. Wang, "The performance analysis of SIP-T signaling system in carrier class VoIP network." *Advanced Information Networking and Applications,* 17th International Conference on, 27-29 Mar 2003, pp: 39 – 44.

[38] O. Haase, K. Murakami, T. Laporta, "Unified mobility manager: Enabling efficient SIP/UMTS mobile network control." *Wireless Communications, IEEE,* Vol: 10 (4), Aug 2003, pp: 66 – 75.

[39] K. Murakami, O. Haase, S. JaeSheung, T. F. La Porta, "Mobility management alternatives for migration to mobile Internet session-based services." *Selected Areas in Communications, IEEE Journal on*, Vol: 22(5), Jun 2004, pp: 818 – 833.

[40] R. Sparks, "Problems identified associated with the Session Initiation Protocol's non-INVITE Transaction." *Network Working Group,* Internet Draft: http://www.softarmor.com/wgdb/docs/draft-sparks-sip-nit-problems-01.html, 2004.

[41] P. Calhoun, J. Loughney, E. Guttman, G. Zorn, and J. Arkko, "Diameter Base Protocol." RFC 3588, *Internet Engineering Task Force*, Sep 2003.

[42] M. Rahman, C. Akinlar, I. Kamel, "On secured end-to-end appliance control using SIP". *Networked Appliances, Liverpool. Proceedings.* IEEE 5th International Workshop on, 30-31 Oct 2002, pp: 24 – 28.

[43] S. Salsano, L. Veltri, "QoS control by means of COPS to support SIP-based applications." *Network, IEEE*, Vol: 16(2), Mar-Apr 2002, pp: 27 – 33.

[44] A. Kist, R. Harris, "Dynamic resource allocation in 3GPP SIP overlay networks." *Information, Communications and Signal Processing, and the Fourth Pacific Rim Conference on Multimedia.* Proceedings of the 2003 Joint Conference of the Fourth International Conference on, Vol: 3, 15-18 Dec 2003, pp: 1952 – 1956.

[45] M. Bozinovski, T. Renier, H. Schwefel, R. Prasad, ”Transaction consistency in replicated SIP call control systems.” *Information, Communications and Signal Processing, and the Fourth Pacific Rim Conference on Multimedia.* Proceedings of the 2003 Joint Conference of the Fourth International Conference on, Vol: 1, 15-18 Dec 2003, pp: 314 – 318.

[46] K. K. Tam, H. Goh, **“**Session Initiation Protocol.” *IEEE International Conference on Industrial Technology,* Vol: 2, 11-14 Dec 2002, pp: 1310 – 1314.

[47] 3GPP, Customized Applications for Mobile network Enhanced Logic (CAMEL); CAMEL Application Part (CAP) specification for IP Multimedia Subsystems (IMS). TS 29.278, 3rd Generation Partnership Project 2004.

[48] ITU-T, Gateway control protocol: Version 2. Recommendation H.248, International Telecommunications Union, May 2002.

[49] 3GPP, Presence service; Architecture and functional description; TS 23.141, v7.0.0, 2005-09

[50] H. Sugano, S. Fujimoto, G. Klyne, A. Bateman, W. Carr, and J. Peterson, Presence Information Data Format (PIDF). Internet-Draft RFC 3863, *Internet Engineering Task Force* 2004.

[51] A. Kist, R. Harris, “Using virtual SIP links to enable QoS for signalling”. *The 11th IEEE International Conference on Networks,* 28 Sep - 1 Oct 2003, pp: 301 – 306.

[52] H. Schulzrinne, V. Gurbani, P. Kyzivat and J. Rosenberg, “RPID – Rich Presence Extensions to the Presence Information Data Format (PIDF).” RFC 4480, *Internet Engineering Task Force*, Jul 2006.

[53] H. Schulzrinne, “Timed Presence Extension to the Presence Information Data Format (PIDF) to indicate Presence Information for Past and Future Time Intervals.” RFC 4481, *Internet Engineering Task Force*, Jul 2006.

[54] A. B. Roach, J. Rosenberg, B. Campbell, “A Session Initiation Protocol (SIP) Event Notification Extension for Resource Lists.” RFC 4662, *Internet Engineering Task Force,* Aug 2006.

[55] G. Camarillo, “Compressing the Session Initiation Protocol (SIP).” RFC 3486, *Internet Engineering Task Force*, Feb 2003.

[56] H. Hannu, J. Christoffersson, S. Forsgren, K.-C. Leung, Z. Liu, and R. Price, “Signalling Compression (SigComp) – Extended Operations.“ RFC 3321, *Internet Engineering Task Force*, Jan 2003.

[57] H. Hannu, J. Rosenberg, C. Bormann, J. Christoffersson, Z. Liu, and R. Price, “Signalling Compression (SigComp).” RFC 3320, *Internet Engineering Task Force*, Jan 2003

[58] R. Droms, J. bound, B. Volz, T. Lemon, C. Perkins, and M. Carney, “Dynamic Host Configuration Protocol for IPv6 (DHCPv6).” RFC 3315, *Internet Engineering Task Force,* Jul 2003.

[59] P. V. Mockapetris, “Domain names – concepts and facilities.” RFC 1034, *Internet Engineering Task Force,* Nov 1987.

[60] H. Schulzrine and B. Volz, “Dynamic Host Configuration Protocol (DHCPv6) Options for Session Initiation Protocol (SIP) Servers.” RFC 3319, *Internet Engineering Task Force,* Jul 2003.

[61] J. Rosenberg and H. Schulzrine, “Session Initiation Protocol (SIP): Locating SIP Servers.” RFC 3263, *Internet Engineering Task Force,* Jun 2002.

[62] W. Stallings, Data & Computer Communications. Sixth Edn. Prentice Hall, NJ, 2002, pp: 1-740.

[63] D. Vali, S. Paskalis, A. Kaloxylos, L. Merakos, ”A SIP-based method for intra-domain handoffs”. *IEEE Vehicular Technology Conference,* VTC 58th, Vol: 3, 6-9 Oct 2003, pp: 2068 – 2072.

[64] D. VaIi, S. Paskalis, A. Kaloxylos, L. Merakos, ”An efficient micro-mobility solution for SIP networks”. *Global Telecommunications Conference,* GLOBECOM '03. IEEE, Vol: 6, 1-5 Dec 2003, pp: 3088 – 3092.

[65] H. Soliman, C. Castelluccia, K. El-Malki, L. Bellier, “Hierarchical Mobile IPv6 Mobility Management (HMIPv6)”, RFC 4140, *IETF*, Aug 2005.

[66] A. Misra, S. Das, A. Datta and S. K. Das, “IDMP-based Fast Handoffs and Paging in IP-Based 4G Mobile Networks”, *IEEE Communications Magazine,* Vol: 40 (3), pp: 138-145, Mar 2002.

[67] R. Braden ed., L. Zhang, S. Berson, S. Herzog, S. Jamin, “Resource ReSeRvation Protocol – Version 1 Functional Specification”, RFC 2205, Sep 1997.

[68] G. Camarillo and M. Garcia-Martin, The 3G IP Multimedia Subsystem. John Wiley & Sons, Ltd. 2004, pp: 5-348.

[69] V. Kueh, R. Tafazolli, B. Evans, “Performance evaluation of SIP-based session establishment over satellite-UMTS.” *Vehicular Technology Conference, 2003. VTC 2003-Spring. The 57th IEEE Semiannual,* Vol: 2, 22-25 Apr 2003, pp: 1381 – 1385.

[70] I. Curcio, M. Lundan, “SIP call setup delay in 3G networks.” *Computers and Communications, Proceedings*. ISCC, Seventh International Symposium on, 1-4 Jul 2002, pp: 835 – 840.

[71] N. Banerjee, K. Basu, S. Das, “Hand-off delay analysis in SIP-based mobility management in wireless networks”. *IEEE Parallel and Distributed Processing Symposium,* Proceedings. International, 22-26 Apr 2003, Digital Object Identifier 10.1109/IPDPS.2003.1213412.

[72] T. Borosa, B. Marsic, S. Pocuca, “QoS support in IP multimedia subsystem using DiffServ.” *Telecommunications, Proceedings of the 7th International Conference on,* Vol: 2, 11-13 Jun 2003, pp: 669 – 672.

[73] M. Grech, M. Torabi, M. Unmehopa, "Service control architecture in the UMTS IP multimedia core network subsystem." *Third International Conference on 3G Mobile Communication Technologies*, 8-10 May 2002, pp: 22 – 26.

[74] 3GPP, End-to-end Quality of Service (QoS) concept and architecture. TS 23.207, 3rd Generation Partnership Project, 2005.

[75] 3GPP, End-to-end Quality of Service (QoS) signalling flows. TS 23.208, 3rd Generation Partnership Project 2005.

[76] 3GPP, General Packet Radio Service (GPRS) Service description; Stage 2. TS 23.060, 3rd Generation Partnership Project, 2005.

[77] M. Baugher, D. McGrew, M. Naslund, E. Carrara, and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)." RFC 3711, *Internet Engineering Task Force,* 2004.

[78] J. Rosenberg, "A Session Initiation Protocol (SIP) Event Package for Registrations." RFC 3680, *Internet Engineering Task Force,* March 2004.

[79] J. Rosenberg, "Extensible Markup Language (XML) Format for Representing Resource Lists." Internet-Draft draft-ietf-simple-xcap-list-usuage-05.txt, *Internet Engineering Task Force,* 2005, Work in Progress.

[80] J. Rosenberg, H. Schulzrinne and P. Kyzivat, "Indicating User Agent Capabilities in the Session Initiation Protocol (SIP)." RFC 3840, *Internet Engineering Task Force,* 2004.

[81] S. M. Faccin, P. Lalwaney, B Patil, "IP Multimedia Services: Analysis of Mobile IP and SIP Interactions in 3G Networks". *IEEE Communications Magazine,* Vol: 8(1), 2004, pp: 113-118.

[82] Y. Kawarasaki, T. Shibata, T. Takahashi, **"**IPv4/IPv6 SIP interworking methods in dual-stack network". *IEEE Communications, APCC.* The 9th Asia-Pacific Conference, Vol: 3, 21-24 Sep 2003, pp: 1124 – 1128.

[83] D. Sisalem, J. Fiedler, R. Ruppelt, "SIP and IPv6: why and how?" *Applications and the Internet Workshops,* IEEE Proceedings. Symposium on, 27-31 Jan 2003, pp: 222 – 225.

[84] P. Flykt, T. Alakoski, "SIP services and interworking with IPv6". *IEEE 3G Mobile Communication Technologies,* Second International Conference on (Conf. Publ. No. 477), 26-28 Mar 2001, pp: 186 – 190.

[85] T. Kwon, M. Gerla, S. Das, "Mobility management for VoIP service: Mobile IP VS. SIP". *IEEE Wireless Communications,* Vol: 9 (5), Oct 2002, pp: 66-75.

[86] M. Holdrege, P. Srisuresh, "Protocol Compilations with the IP Network Address Translator." RFC 3027, *IETF*, Jan 2001.

[87] E. Fogelstroem, A. Jonsson, C. Perkins, "Mobile IPv4 Regional Registration." Internet draft, draft-ietf-mip4-reg-tunnel-04, *IETF,* work in progress, 2007.

[88] N. Nakajima, A. Dutta, S. Das, H. Schulzrinne, "Handoff delay analysis and measurement for SIP based mobility in IPv6." *Communications, ICC '03. IEEE International Conference on*, Vol: 2, 11-15 May 2003, pp: 1085-1089.

[89] H. Omar, T. Saadawi and M. Lee, "Multicast Support for Mobile-IP with the Hierarchical Local Registration Approach." ACM *WOWMOM*, 2000, pp: 55-64.

[90] Z. Wu, "An approach for optimizing binding lifetime with mobile IPv6". *Local Computer Networks, LCN '03.* Proceedings. 28th Annual IEEE International Conference on, 20-24 Oct 2003, pp: 82 – 88.

[91] B. Campbell, R. Mahy, C. Jennings, "The Message Session Relay Protocol". Internet-Draft draft-ietf-simple-message-sessions-19, *Internet Engineering Task Force,* 2007, Work in Progress.

[92] Y. Cai, X. Y. Li, Y. Wang, J. B. Reid, and P. Wang*,* "IP Multimedia Subsystem Online Session Charging Call Control**"** *Bell Labs Technical Journal*, Vol: 10(4), 2006,

pp: 117–132, Lucent Technologies Inc. Published by Wiley Periodicals, Inc. DOI: 10.1002/bltj.20129.

[93] D. M. Ward, D. P. Strand, W. A. Senneke, B. L. Knolinski, R. J. Scheer, and A. J. McBride, "Domain Management of IMS" *Bell Labs Technical Journal*, Vol: 10(4), 2006, pp: 233–254, Lucent Technologies Inc. Published by Wiley Periodicals, Inc. DOI: 10.1002/bltj.20136.

[94] E. E. Anderlind, D. W. Faucher, E. H. Grosse, D. N. Heer, A. R. McGee, D. P. Strand, and R. J. Thornberry Jr., "IMS Security" *Bell Labs Technical Journal*, Vol: 11(1), 2006, pp: 37–58, Lucent Technologies Inc. Published by Wiley Periodicals, Inc. DOI: 10.1002/bltj.20143.

[95] R. Gaglianello and D. Liu, "IMS Shared Streaming Video" *Bell Labs Technical Journal*, Vol: 10(4), 2006, pp: 71–75, Lucent Technologies Inc. Published by Wiley Periodicals, Inc. DOI: 10.1002/bltj.20124.

[96] M. R. G. Azada, R. P. Ejzak, J. J. MacNamara, D. Sand, and R. Thompson, "Seamless Mobility Across IMS and Legacy Circuit Networks" *Bell Labs Technical Journal*, Vol: 10(4), 2006, pp: 25–38, Lucent Technologies Inc. Published by Wiley Periodicals, Inc. DOI: 10.1002/bltj.20121.

[97] J. Brok, B. Kumar, E. Meeuwissen, and H. J. Batteram, "Enabling New Services by Exploiting Presence and Context Information in IMS" *Bell Labs Technical Journal*, Vol: 10(4), 2006, pp: 83–100, Lucent Technologies Inc. Published by Wiley Periodicals, Inc. DOI: 10.1002/bltj.20126.

[98] D. F. Lieuwen, T. C. Morgan, H. L. Raether, S. K. Ramamoorthy, M. Xiong, and R. B. Hull, "Subscriber Data Management in IMS Networks" *Bell Labs Technical Journal*, Vol: 10(4), 2006, pp: 197–215, Lucent Technologies Inc. DOI: 10.1002/bltj.20134.

[99] T. Bessis, "Improving Performance and Reliability of an IMS Network by Co-Locating IMS Servers" *Bell Labs Technical Journal*, Vol: 10 (4), 2006, pp: 167–178,

Lucent Technologies Inc. Published by Wiley Periodicals, Inc. DOI: 10.1002/bltj.20132.

[100] 3rd Generation Partnership Project, “Charging Principles,” 3GPP TS 32.200, V 5.8.0, Mar. 2005.

[101] 3rd Generation Partnership Project, “Charging Architecture and Principles,” 3GPP TS 32.240, V 6.1.0, Mar. 2005.

[102] 3rd Generation Partnership Project, “IP Multimedia Subsystem (IMS) Charging,” 3GPP TS 32.260, V 6.1.0, Mar. 2005.

[103] N. Rajagopal, M. Devetsikiotis, "Modeling and Optimization for the Design of IMS Networks," *39th Annual Simulation Symposium* 2006, pp: 34-41.

[104] F. C. de Gouveia, T. Magedanz, "A Framework to Improve QoS and Mobility Management for Multimedia Applications in the IMS," *ism*, 2005, pp: 216-222, *Seventh IEEE International Symposium on Multimedia.*

[105] D. Geer, "Building Converged Networks with IMS Technology," *Computer*, Vol: 38 (11), Nov 2005, pp. 14-16.

[106] K. L. Larsen, E. V. Matthiesen, H.-P. Schwefel, G. Kuhn, "Optimized Macro Mobility within the 3GPP IP Multimedia Subsystem," *icwmc*, 2006, *International Conference on Wireless and Mobile Communications.*

[107] M. Sher, T. Magedanz, W. T. Penzhorn, "Inter-Domains Security Management (IDSM) Model for IP Multimedia Subsystem (IMS)," *ares*, 2006, pp: 502-509, *First International Conference on Availability, Reliability and Security*.

[108] J. Rosenberg, “The Extensible Markup Language (XML) Configuration Access Protocol (XCAP)”, Internet Draft draft-ietf-simple-xcap-12.txt, *Internet Engineering Task Force*, Work in Progress, 2007.

[109] Retrieved from URL: http://www.ericsson.com/products/

[110] L. A. DaSilva, G. E. Morgan, C. W. Bostian, D. G. Sweeney, S. F. Midkiff, J. H. Reed and C. Thompson, "The Resurgence of Push-to-Talk Technologies." *IEEE Communication Magazine,* Vol: 44 (1), 2006, pp: 48-55.

[111] T. Ali-Vehmas, S. Luukkainen, "Service Diffusion Strategies for Push to Talk Over Cellular." *IEEE International Conference on Mobile Business, ICMB,* 2005, pp: 427-433.

[112] OMA, Open Mobile Alliance. Push to talk over Cellular (PoC)-Architecture. Push to Talk Over Cellular Working Group. 2005; URL: http://www.openmobilealliance.org.

[113] W. C. Chan, Performance analysis of telecommunications and local area networks, Boston, Kluwer Academic, 2002.

[114] D. L. Jagerman, Difference equations with applications to queues, NY: M. Dekker, 2005.

[115] L. Sennott, Stochastic dynamic programming and the control of queuing systems, NY: Wiley 1999, pp: 15-164.

[116] D. Gross, C. M. Harris, Fundamentals of queuing theory, 3rd edition, NY: Wiley 1998, pp: 1-401.

[117] F. P. Kelly and R. J. Wiliams, Stochastic networks, NY: Springer-Verlag, 1995.

[118] H. G. Perros, Queuing networks with blocking: exact and approximate solutions, NY: Oxford University Press, 1994, pp: 57-85.

[119] T. G. Robertazzi, Computer networks and systems: queuing theory and performance evaluation, 2nd edition, NY: Springer-Verlag, 1994.

[120] C. D. Meyer and R. J. Plemmons, Linear algebra, Markov chains, and queuing models, NY: Springer-Verlag, 1993.

[121] H. Takagi, Queuing analysis: a foundation of performance evaluation, Vol: 1, NY: Elsevier Science Pub. Co. 1993, pp: 1-447.

[122] M. E. Woodward, Communication and computer networks: modelling with discrete-time queues, London" Pentech, 1993, pp: 1-128.

[123] U. N. Bhat and I. V. Basawa, Queuing and related models, Oxford: Clarendon Press; NY: Oxford University Press, 1992.

[124] J. Mehdi, Stochastic models in queuing theory, Boston: Academic Press, 1991, pp: 1-416.

[125] D. Stoyan, Comparison methods for queues and other stochastic models, Chichester; NY: Wiley 1983.

[126] J. W. Cohen, The single server queue, Revise edition, NY: Elsevier, 1982, pp: 34-633.

[127] G. F. Newell, Applications of queuing theory, 2nd edition, London; NY: Chapman and Hall, 1982, pp: 24-261.

[128] R. B. Cooper, Introduction to queuing theory, 2nd edition, NY: North Holland, 1981, pp: 1-307.

[129] S. C. Bruell, Gianfranco Balbo, Computational algorithms for closed queuing networks, NY: North Holland, 1980.

[130] A. O. Allen, Probability, statistics and queuing theory: with computer science applications, NY: Academic Press, 1978.

[131] V. K. Garg, Wireless network evolution: 2G to 3G, Upper Saddle River, NJ: Prentice Hall, 2001, pp: 3-714.

[132] R. Lioyd-Evans, QoS in integrated 3G networks, Boston: London: Artech House, 2002.

[133] L. Kleinrock, Queuing Systems, Vol: 2, Computer Applications, John Wiley & Sons, New York, 1976, pp: 10-318.

[134] C. Urrutia-Veldes, A. Mukhopadhyay, M. El-Sayed, "Presence and availability with IMS: Applications Architecture, Traffic Analysis, and Capacity Impacts" *Bell Labs Technical Journal*, Vol: 10(4), 2006, pp: 101-107, Lucent Technologies Inc. Wiley Periodicals, DOI: 10.1002/bltj.20127

[135] H. K. Telio, E. Leppanen, M. Lonnfors, J. C. Requena, "Functional Description of Event Notification Filtering." RFC 4660, *Internet Engineering Task Force,* 2006.

[136] A. Niemi, "Session Initiation Protocol (SIP) Event Notification Extension for Notification Throttling." Internet-Draft draft-niemi-sipping-event-throttle-05, *Internet Engineering Task Force* 2007; Work In progress.

[137] A. Balazs, "QoS in wireless networks: Push-to-talk performance over GPRS" *Proceedings of the 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems*, 2004, pp: 182-187.

[138] Northstream Report, "Overview and comparison of Push-to-talk" URL: http://www.northstream.se, 2004.

[139] S. Raktale, "3PoC: An Architecture for enabling Push To Talk services in 3GPP Networks" *IEEE International Conference on Personal wireless communications* (ICPWC), 2005, pp: 202-206.

[140] A. Parthasarathy. "Push to Talk over Cellular (PoC) Server" *IEEE International Conference on Networking*, Sensing and Control, 2005, pp: 772-776.

[141] P. Kim, A. Balazs, E. Broek, G. Kieselmann, W. Bohm, "IMS-based Push-to-Talk over GPRS/UMTS" *IEEE Wireless Communications and Networking Conference*, Vol: 4, 2005, pp: 2472-2477.

[142] A. Hafslund, T. T. Hoang and Q. Kure, "Push-to-talk applications in mobile ad hoc networks." *Vehicular Technology Conference*, 2005. VTC 2005-Spring. IEEE 61st, Vol: 4(30), pp: 2410 – 2414.

[143] Lin-Yi Wu, Meng-Hsun Tsai, Yi-Bing Lin and Jen-Shun Yang, "A client-side design and implementation for push to talk over cellular service" *Wireless Communications and Mobile Computing,* Published online in Wiley InterScience, 2006, www.interscience.wiley.com. DOI: 10.1002/wcm.369.

[144] A. W. Berger, W. Whitt, "Effective bandwidths with priorities." *IEEE/ACM Transactions on Networking*; Vol: 6 (4), 1998, pp: 447-460.

[145] S. Jordan, K. Jogi, C. Shi, I. Sidhu, "The variation of optimal bandwidth and buffer allocation with the number of sources." *IEEE/ACM Transactions on Networking*; Vol: 12 (6), 2004, pp: 1093-1104.

[146] D. Ayyagari, A. Ephremides, "Admission control with priorities: approaches for multi-rate wireless system." *Mobile Networks and Applications*; Vol: 4 (3), 1999, pp: 209-218.

[147] T. Janevski, B. Spasenovski, "QoS provisioning for wireless IP networks with multiple classes through flexible fair queuing." *Global Telecommunications Conference*, GLOBECOM '00. IEEE 2000; Vol: 3 (27), pp:1677 – 1701.

[148] A. Roach, "Session Initiation Protocol (SIP)-Specific Event Notification." RFC 3265, *Internet Engineering Task Force* 2002.

[149] P. Marbach, "Analysis of a static pricing scheme for priority services." *IEEE/ACM Transactions on Networking* (TON); Vol: 12 (2), 2004, pp: 312 – 325.

[150] J. R. Moorman, J. W. Lockwood, "Multiclass priority fair queuing for hybrid wired/wireless quality of service support." *Proceedings of the 2nd ACM international workshop on Wireless mobile multimedia,* 1999, pp**:** 43 – 50.

[151] “Traffic Management Specification v4.0.” ATM Forum Document AF-TM-0056.000, 1996.

[152] C. C. Beard, V. S. Frost, “Prioritized Resource Allocation for Stressed Networks.” *IEEE/ACM Transactions on Networking,* 2001; Vol: 9 (5), pp: 618-633.

[153] M. Ghaderi and R. Boutaba, “Call Admission Control for Voice/Data Integration in Broadband Wireless Networks.” *IEEE Transactions on Mobile Computing* 2006; Vol: 5 (3), pp: 193-207.

[154] S. Floyd and V. Jacobson, “Link-sharing and resource management models for packet networks,” *IEEE/ACM Transactions on Networking* 1995; Vol: 3(4), pp: 365-386.

[155] F. Risso, “Decoupling Bandwidth and Delay Properties in Class Based Queuing,” *Computers and Communications*, Proceedings. Sixth IEEE Symposium on, 3-5 Jul 2001, pp: 524 – 531.

[156] D. Hong and S. S. Rappaport, “Traffic model and performance analysis for cellular mobile radio telephone systems with prioritized and nonprioritized handoff procedures,“ *IEEE Transactions on Vehicular Technology*, Vol: 35(3), 1986, pp: 77-92.

[157] Y. Fang, I. Chlamtac, and Y. B. Lin, “Channel Occupancy Times and Handoff Rate for Mobile Computing and PCS Networks,” *IEEE Transactions on computers*, Vol: 47(6), 1998, pp: 679-692.

[158] Y. Fang and I. Chlamtac, “Analytical Generalized Results for Handoff Probability in Wireless Networks” *IEEE Transactions on communications*, Vol: 50(3), 2002, pp: 396-399.

[159] M. M. Zonoozi and P. Dassanayake, “User mobility modelling and characterization of mobility patterns,” *IEEE journal on selected areas in communications,* Vol: 15(7), Sep 1997, pp: 1239-1252.

[160] L. Gavrilovska, T. Janevski, “Modelling Techniques for Mobile Communications Systems.” *IEEE GLOBECOM’98* Sydney, Australia, Nov 1998, pp: 2623-2628.

[161] T. Janevski, Traffic analysis and design of wireless IP networks Artech house, Boston, London, 2003, pp: 91-347.

[162] K. S. Trivedi, Probability & Statistics with Reliability, Queuing, and computer Science Applications, New Delhi, Prentice’ Hall of India, 2003, pp: 1-464.

[163] M. Cortes, J. Robert Ensor, and J. O. Esteban, “On SIP Performance” *Bell Labs Technical Journal*, Vol: 9(3), 2004, pp: 155-172, Wiley Periodicals, Inc.

[164] H. Batteram, E. Meeuwissen, and J. V. Bemmel, “SIP Message Prioritization and Its Applications” *Bell Labs Technical Journal*, Vol: 11(1), 2006, pp: 21-36, Wiley Periodicals Inc.

[165] M. A Siddiqui, K. Guo, S. Rangarajan, and S. Paul, “End-to-End QoS Support for SIP Sessions in CDMA 2000 Networks” *Bell Labs Technical Journal*, Vol: 9(3), 2004, pp: 135-153, Wiley Periodicals, Inc.

[166] M. Molina, J. Quittek, M. Brunner, and Telemaco Melia, “Scalable and efficient QoS support for SIP-signalled voice calls” *International Journal of Communication Systems*, Vol: 19 (4), 2006, pp: 407-424, Wiley InterScience, DOI: 10.1002/dac.794.

[167] S.-M. Huang, Q. Wu, Y. Lin, and C.-H. Yeh, “SIP mobility and IPv4/IPv6 dual-stack supports in 3G IP multimedia subsystem” *Wireless Communications and Mobile Computing*, Vol: 6 (5), 2006, pp: 585-599, Wiley InterScience

[168] Q. Wang and M. A. Abu-Rgheft, “Signalling analysis of cost-efficient mobility support by integrating mobile IP and SIP in all IP wireless networks” *International Journal of Communications Systems*, Vol: 19 (2), 2006, pp: 225-247, Wiley InterScience.

[169] H. Lee, J.-Y. Song, S.-H. Lee, S. Lee, and D.-H. Cho, “Integrated Mobility Management Methods for Mobile IP and SIP in IP based Wireless Data Networks” *Wireless Personal Communications*, Vol: 35 (3), 2005, pp: 269-287, Springer.

[170] H. Fathi, S. S. Chakraborty, R. Prasad, “Optimization of SIP Session Setup Delay for VoIP in 3G Wireless Networks” *IEEE Transactions on Mobile Computing,* Vol: 5(9) Sep 2006, pp: 1121-1132.

[171] I. F. Akyildiz and W. Wang, “A Predictive User mobility Profile for Wireless Multimedia Networks,” *IEEE/ACM Transactions on Networking*, Vol: 12 (6), 2004, pp: 1021-1035.

[172] M. D. Austin and G. L Stuber, “Velocity Adaptive Handoff Algorithms for Microcellular Systems,” *IEEE Transactions on Vehicular Technology*, Vol: 43 (3), 1994, pp: 549-561.

[173] J. M. Holtzman and A. Sampath, “Adaptive Averaging Methodology for Handoffs in Cellular Systems,” *IEEE Transactions on Vehicular Technology*, Vol: 44 (1), 1995, pp: 59-66.

[174] S. Mohanty and I.F. Akyildiz, “A Cross-Layer (Layer 2+3) Handoff Management Protocol for Next-Generation wireless Systems,” *IEEE Transactions on Mobile Computing*, Vol: 5 (10), 2006, pp: 1347-1360.

[175] 3GPP TSG GERAN, “A Comparison between Packet-Switched Call Setup Using SIP and GSM Circuit-Switched Call Setup Using RIL3-CC, RIL3-MM, RIL3-RR, and DTAP,” GP-000508, Nortel Networks, Nov 2000.

[176] S. Das, E. Lee, K. Basu, and S. Sen, “Performance Optimization of VoIP Calls over Wireless Links Using H.323 Protocol,“ *IEEE Transactions on Computers*, Vol: 52 (6), 2003, pp: 742-752.

[177] H. Petander, E. Perera, K.-C. Lan, and A. Seneviratne, “Measuring and Improving the Performance of Network Mobility Management in IPv6 Networks,”

IEEE Transactions on Selected Areas in Communications, Vol: 24 (9), 2006, pp: 1671-1681.

[178] C.-M. Huang, C.-H. Lee, and J.-R. Zheng, “A Novel SIP-Based Route Optimization for Network Mobility,” *IEEE Transactions on Selected Areas in Communications,* Vol: 24 (9), 2006, pp: 1682-1691.

[179] Y.-H. Han. J. Choi, and S.-H. Hwang, “Reactive Handover Optimization in IPv6-Based Mobile Networks” *IEEE Transactions on Selected Areas in Communications,* Vol: 24 (9), 2006, pp: 1758-1772.

[180] M. T. Alam, Z. D. Wu, “Comparison of Session Establishment Schemes Over IMS in Mobile Environment.” *Fifth IEEE International Conference on Information, Communications and Signal Processing, ISBN:* 0-7803-9283-3, IEEE Catalog Number: 05EX1118C, Dec 6-9, Bangkok, Thailand, 2005, pp: 638-642, Paper ID: 0581

[181] F. P. Kelly, “Blocking probabilities in large circuit switched networks.” *Advances in Applied Probability,* Vol: 18, 1986, pp: 473-505.

[182] F. P. Kelly, “Adaptive routing in circuit-switched networks,” *Statistical Laboratory,* Cambridge University, Cambridge, England, 1986.

[183] F. P. Kelly, “Routing in circuit-switched networks: optimization, shadow prices and decentralization.” *Advances in Applied Probability*, Vol: 20, 1988, pp: 112-144.

[184] X. Liu, “Network optimization with stochastic traffic flows.” *International Journal of Network Management,* Vol: 12, 2002, pp: 225-234.

[185] E. N. Gilbert, “Capacity of a burst-noise channel,” *Bell Syst. Tech. J.*, Vol: 39, 1960, pp: 1253-1265.

[186] F. Swarts and H. C. Ferreira, “Markov characterization of digital fading mobile VHF channels.” *IEEE Trans. Veh. Technol.,* Vol: 43, 1994, pp**:** 977–985.

[187] H. S. Wang and N. Moayeri, "Finite-state Markov channel: A useful model for radio communications channels." *IEEE Trans. Veh. Technol.*, Vol: 44, 1995, pp**:** 163–171.

[188] E. Altman, K. Avrachenkov and C. Barakat, "TCP in presence of bursty losses." *in Proc. ACM SIGMETRICS,* 2000, pp: 124–133.

[189] J. Bolot, S. Parisis and D. Towsley, "Adaptive FEC-based error control for Internet telephony." *in Proc. IEEE INFOCOM,* Vol: 3, 1999, pp**:** 1453–1460.

[190] B. Akan, I. F. Akyildiz, "ARC: The Analytical Rate Control Scheme for Real-Time Traffic in Wireless Networks." *IEEE/ACM Transactions on Networking,* Vol: 12 (4), 2004, pp: 634-544.

[191] V. Ronnholm, "Push-to-Talk over Bluetooth" *Proceedings of the 39th Hawaii International Conference on System Sciences*, IEEE, 2006, pp: 232-241.

[192] R. Guerin, "Channel occupancy time distribution in a cellular radio system." *IEEE Transactions on Vehicular Technology,* Vol: 35 (3), Aug 1987.

[193] M. Hellebrandt, R. Mathar, M. Scheibenbogen, ''Estimating position and velocity of mobiles in a cellular radio network.'' *IEEE Transactions on Vehicular Technology,* Vol: 46, 1997, pp: 65-71.

[194] C. Wu, H. Lin, L. Lan, ''A New Analytic Framework for Dynamic Mobility Management of PCS Networks." *IEEE Transactions on Mobile Computing,* Vol: 1 (3), 2002, pp: 208-219.

[195] J. F. C. Kingman JFC. "Inequalities in the Theory of Queues." *Journal of Royal Statistical Society*; Vol: 32 (B), 1970, pp: 102-110.

[196] A. W. Berger and Y. Kogan, "Dimensioning Bandwidth for Elastic Traffic in High-Speed Data Networks." *IEEE/ACM Transactions on Networking*; Vol: 8 (5), 2000, pp: 643-654.

[197] A. Striegel and G. Manimaran, **"**Dynamic class-based queue management for scalable media servers," *Real-Time Technology and Applications Symposium*, RTAS, Proceedings, Sixth IEEE 31 May-2 Jun 2000, pp: 228 – 236.

[198] Y. Guo and H. Chaskar, "Class-based quality of service over air interfaces in 4G mobile networks" *Communications Magazine*, IEEE, Vol: 40 (3), Mar 2002, pp: 132 – 137.

[199] F. A. Cruz-Perez and L. Ortigoza-Guerrero, "Flexible resource allocation strategies for class-based QoS provisioning in mobile networks" *Vehicular Technology, IEEE Transactions on,* Vol: 53 (3), May 2004, pp: 805 – 819.

[200] D. I. Kim, E. Hossain, V. K. Bhargava, "Dynamic rate and power adaptation for provisioning class-based QoS in Cellular Multirate WCDMA systems" *Wireless Communications, IEEE Transactions on,* Vol: 3 (5), Sept. 2004, pp:1590 – 1601.

[201] C. Fragouli, V. Sivaraman, M. B. Srivastava, **"**Controlled multimedia wireless link sharing via enhanced class-based queuing with channel-state-dependent packet scheduling" IN*FOCOM '98. Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies.* Proceedings. IEEE, Vol: 2, 29 Mar - 2 Apr 1998, pp: 572 – 580.

[202] Y. Huang, H. Garcia-Molina, "Publish/subscribe in a mobile environment" Wireless Networks, Kluwer Academic Publisher, Vol: 10, 2004, pp: 643-652.

[203] F. Baskett, K. M. Chandy, R. R. Muntz and F. G. Palacios "Open, Closed and Mixed Networks of Queues with Different Classes of Customers", *Journal of the Association for the Computing Machinery*, Vol: 22(2), Apr 1975, pp: 248-260.

[204] M. T. Alam, An Optimal Timer for Push-To-Talk Controller, In D. Taniar (Ed.), Encyclopedia of Mobile Computing and Commerce, Hershey, PA: Information Science Reference., Vol: 2, 2007, pp: 724-728.

[205] 3GPP, TSG SSA, IP Multimedia Subsystem (IMS) – Stage 1 (Release 8), TS 22.228 v.8.1.0, 2006-09.

[206] 3GPP, TSG SSA, IP Multimedia Subsystem (IMS) – Stage 2 (Release 6), TS 23.228 v.6.3.0, 2006-03.

[207] B. Campbell, J. Rosenberg, H. Schulzrine, C. Huitema, and D. Gurle, "Session Initiation Protocol (SIP) Extension for Instant Messaging", RFC 3428, *Internet Engineering Task Force,* 2002.

[208] 3GPP, Internet Protocol (IP) multimedia call control protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage 3, TS 24.229.

[209] E. Burger**, "**A Mechanism for Content Indirection in Session Initiation Protocol (SIP) Messages" RFC 4483, *Internet Engineering Task Force,* 2006.

[210] M. Debbabi, M. Rahman, W. Lin, J. Gopal, S. Sridhar, "Standard SIP-based instant messaging and presence APIs for networked devices", *Networked Appliances, Proceedings. IEEE 5th International Workshop on*, 30-31 Oct 2002, pp: 59 - 64.

[211] M. Gomez, J. L. Megias, C. Bueno, C. Brocal, "Interworking between the Multimedia Messaging Service (MMS) and the 3G IP Multimedia subsystem (IMS) Instant Messaging Service", *IEEE 16th International Symposium on Personal, Indoor and Mobile Radio communications*, 2005, pp: 2274-2278.

[212] L. Zhou, "An empirical Investigation of Deception Behaviour in Instant Messaging", *IEEE Transactions on Personal Communication*, Vol: 48 (2), 2005, pp: 147-160.

[213] C. Jennings, R. Mahy, A. B**.** Roach, "Relay Extensions for the Message Sessions Relay Protocol (MSRP)", *Internet Engineering Task force*, draft-ietf-simple-msrp-relays-09.txt**,** 2007, Work in Progress.

[214] M. Gastpar, M. Vetterli, **"**On the capacity of wireless networks: the relay case" *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings.* IEEE, Vol: 3, 23-27 Jun 2002, pp: 1577 – 1586, Digital Object Identifier 10.1109/INFCOM.2002.1019409.

[215] A. Host-Madsen, “On the capacity of wireless relaying”, *Vehicular Technology Conference, 2002. Proceedings, IEEE 56th*, Vol: 3, 24-28 Sep 2002, pp: 1333 – 1337, Digital Object Identifier 10.1109/VETECF.2002.1040432.

[216] B. Wang, J. Zhang, A. Host-Madsen, “On the capacity of MIMO relay channels”**,** *Information Theory, IEEE Transactions on*, Vol: 51 (1), Jan 2005, pp: 29 – 43, Digital Object Identifier 10.1109/TIT.2004.839487.

[217] H. B¨olcskei, R. U. Nabar, O. Oyman, A. J. Paulraj, “Capacity Scaling Laws in MIMO Relay Networks”, *IEEE Transactions on Wireless Communications*, Vol: 5 (6), Jun 2006, pp: 1433-1443.

[218] N. Gulpinar, P. Harrison, B. Rustem, “Optimization of a tandem M/GI/1 router network with batch arrivals”, *Parallel and Distributed Processing Symposium, 2005. Proceedings*. 19th IEEE International, 4-8 Apr 2005, Digital Object Identifier 10.1109/IPDPS.2005.312.

[219] S. Chakravarthy, “A finite capacity queuing network with single and multiple processing nodes”, *Proceedings of the Second International Conference on Queuing Networks with Finite Capacity,* 28-29 May, 1992, NC, USA.

[220] M. T. Alam, Z. D. Wu, “End-to-End Delay Measurement for Instant Messaging Relay Nodes,” *Ubiquitous Computing and Communication Journal,* Vol: 2 (2), Mar 2007, ISSN 1992-8424.

[221] L. Kleinrock, Queuing Systems, Vol: 1, Theory, Wiley, New York, 1975, pp: 3-313.

[222] N. G¨ulpınar, U. Harder, P. Harrison, T. Field, B. Rustem, L.-F. Pau. “Mean-Variance Performance Optimization of Response Time in a Tandem Router Network with Batch Arrivals,” *Cluster Computing,* Springer, 2007, DOI: 10.1007/s10586-007-0016-9.