Spring 3-22-2013

STOCHASTIC MODELING AND TIME-TO-EVENT ANALYSIS OF VOIP TRAFFIC

Imad Al Ajarmeh
DePaul University, iajarmeh@gmail.com

STOCHASTIC MODELING AND TIME-TO-EVENT ANALYSIS OF VOIP TRAFFIC

BY

IMAD AL AJARMEH

A DISSERTATION SUBMITTED TO

THE COLLEGE OF COMPUTING AND DIGITAL MEDIA OF

DEPAUL UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS OF THE DEGREE

OF

DOCTOR OF PHILOSOPHY

CHICAGO, IL

2013

DePaul University

College of Computing and Digital Media

School of Computing

Stochastic Modeling and Time-to-Event Analysis of VoIP Traffic

Abstract

Voice over IP (VoIP) systems are gaining increased popularity due to the cost effectiveness, ease of management, and enhanced features and capabilities. Both enterprises and carriers are deploying VoIP systems to replace their TDM-based legacy voice networks. However, the lack of engineering models for VoIP systems has been realized by many researchers, especially for large-scale networks. The purpose of traffic engineering is to minimize call blocking probability and maximize resource utilization. The current traffic engineering models are inherited from the legacy PSTN world, and these models fall short from capturing the characteristics of new traffic patterns. The objective of this research is to develop a traffic engineering model for modern VoIP networks. We studied the traffic on a large-scale VoIP network and collected several billions of call information. Our analysis shows that the traditional

traffic engineering approach based on the Poisson call arrival process and exponential holding time fails to capture the modern telecommunication systems accurately. We developed a new framework for modeling call arrivals as a non-homogeneous Poisson process, and we further enhanced the model by providing a Gaussian approximation for the cases of heavy traffic condition on large-scale networks. In the second phase of the research, we followed a new time-to-event survival analysis approach to model call holding time as a generalized gamma distribution and we introduced a Call Cease Rate function to model the call durations. The modeling and statistical work of the Call Arrival model and the Call Holding Time model is constructed, verified and validated using hundreds of millions of real call information collected from an operational VoIP carrier network. The traffic data is a mixture of residential, business, and wireless traffic. Therefore, our proposed models can be applied to any modern telecommunication system. We also conducted sensitivity analysis of model parameters and performed statistical tests on the robustness of the models' assumptions.

We implemented the models in a new simulation-based traffic engineering system called VoIP Traffic Engineering Simulator (VSIM). Advanced statistical and stochastic techniques were used in building VSIM system. The core of VSIM is a simulation system that consists of two different simulation engines: the NHPP parametric simulation engine and the non-parametric simulation engine. In addition, VSIM provides several subsystems for traffic data collection, processing, statistical modeling, model parameter estimation, graph generation, and traffic prediction. VSIM is capable of extracting traffic data from a live VoIP network, processing and storing the extracted information, and then feeding it into one of the simulation engines which in turn provides resource optimization and quality of service reports.

This Doctoral dissertation is lovingly dedicated to my parents for their continuous encouragement and prayers that sustained me throughout my entire life and to my wife and children for their passion, love, patience and support.

Acknowledgements

First and foremost, I would like to thank my advisor Dr. James Yu for his continuous support and guidance. It has been a great honor to be his Ph.D. student. He has taught me, both consciously and unconsciously, how theoretical and experimental research is done. I highly appreciate all his contributions of effort, time, ideas and demand of excellence to make my Ph.D. experience productive and stimulating. I am also thankful for the excellent example he has provided as a successful and brilliant professor with razor-sharp wit.

I will forever be thankful for Mr. John Bullock and Mr. Paul Liu from Inteliquent for their generous support, encouragement and sponsorship to my research that has positively affected my academic as well as my professional life. The support, friendship, trust and enthusiasm I received from Inteliquent family made this research both possible and rewarding.

Also I would like to acknowledge the inspiration and guidance I received from Dr. Mohamed Amezziane. His brilliant mind and unflagging willingness to help did make this research fruitful.

In addition, I would like to thank the College of Computing and Digital Media (CDM) at DePaul University for their generous scholarship that has changed my entire life. The quality of education, advice, and encouragement I received for the CDM faculty gave me the tools to successfully move ahead into this research. I especially want to thank Dr. Anthony Chung and Dr. Gregory Brewster for their guidance, time and effort they have spent reviewing my work.

I also thank my siblings and friends (too many to list here but they know who they are) for providing the support and friendship that I needed.

I thank all of you.

CONTENTS

1 Introduction 22

1.1 Motivation 24

1.2 Scope 24

1.3 Goals 25

1.4 Contributions 25

1.5 Publications 28

1.6 Thesis Outline 29

2 VoIP Networks 32

2.1 VoIP Traffic 32

2.2 VoIP Network Types 34

2.3 Call Admission Control (CAC) 39

2.4 VoIP Call Resources and Admission Control 42

2.4.1 Empirical Results and Analysis 43

2.4.2 Packet Throughput and Maximum Call Load 46

2.5 Summary 48

3 Literature Review of Traffic Models 50

3.1 Telecommunication System Modeling 50

3.1.1 Call Arrival Process 52

3.1.2 Call Holding Time 54

3.2 Traffic Engineering Models 55

3.2.1 Traffic Measurement 55

3.2.2 Erlang-B model 56

3.2.3 Erlang-B model Extensions 59
3.3 Other Research on Traffic Models 62
3.3.1 Modeling Call Arrival Process 63
3.3.1.1 Batch (Session) Based Call Arrivals 64
3.3.1.2 Traditional Stationary Poisson Arrival Rate 66
3.3.1.3 Erlang-jk 66
3.3.1.4 BCMP 67
3.3.1.5 Non-homogeneous Poisson Process 68
3.3.1.6 Packet-level Arrival Modeling 69
3.3.2 Modeling Call Holding Time 70
3.3.2.1 Traditional Exponential Call Holding Time 71
3.3.2.2 Lognormal Model 71
3.3.2.3 Mixture of Lognormals 72
3.3.2.4 Phase Type Distributions 73
3.3.2.5 Weibull and Piecewise Weibull Distributions 75
3.3.2.6 Pareto Distribution 76
3.4 Summary 77
4 Research Methodology 82
4.1 VoIP Traffic Data from IP Tandem Network 82
4.1.1 IP Tandem Network 83
4.1.2 Data Collection and Processing 84
4.2 Mathematical and Statistical Modeling and Analysis 88
4.3 Simulation 92
5 Traffic Engineering Modeling Results and Analysis 96
5.1 Modeling Call Holding Time for VoIP Tandem Networks 96
5.1.1 Data Exploration 97

5.1.2 Introducing the Call Cease Rate Function 99
5.1.3 Model Estimation 105
5.1.4 Goodness of Fit and Model Validation 109
5.1.5 Final Model 112
5.1.6 Summary 113
5.2 Modeling Call Arrival Rate as NHPP 114
5.2.1 Call Arrival Patterns 115
5.2.2 Model Formulation and Validation 116
5.2.3 Traffic Prediction 123
5.2.4 Summary 124
5.3 Normality Approximation of Call Arrivals under Heavy Traffic Condition 124
5.3.1 Model Building 126
5.3.2 Parameter Estimation 127
5.3.3 Model Validation 134
5.3.4 Prediction and Model Comparison 137
5.3.5 Summary 139
6 VSIM Description and System Design 140
6.1 Development Approach 141
6.2 VSIM Design 142
6.2.1 Traffic Data Collector: 143
6.2.2 Traffic Processing and Aggregation Engine 145
6.2.3 Model Generation engine 146
6.2.4 VSIM Graph Generator: 151
6.2.5 VSIM Traffic Prediction Engine: 151
6.2.6 VSIM simulation Engine 153
7 VSIM Simulation Implementation and Experiments 154

7.1 Introduction 155
7.2 Telecommunication System Simulation 157
7.3 VSIM simulation engines 158
7.3.1 Parametric VSIM G/G/c/c simulator 163
7.3.2 Non-parametric VSIM simulator 163
7.4 VSIM model verification 164
7.4.1 Internal simulation random variables 164
7.4.2 VSIM simulation engine algorithms 166
7.5 VSIM Simulation Model Validation 168
7.6 Simulation results and Analysis 170
7.7 Summary 175
8 Conclusions and Future Work 176

LIST OF FIGURES

Figure 1. Residential Internet-based VoIP 35

Figure 2. Enterprise VoIP architecture 36

Figure 3. SIP Trunking VoIP solution 37

Figure 4. VoIP carrier network 38

Figure 5. Call Admission control for VoIP system 40

Figure 6. Call utilization for various links 45

Figure 7. Telecommunication system model 51

Figure 8. Typical IP-based tandem network 83

Figure 9. Number of collected Call Detail Records 84

Figure 10. Data collection process 85

Figure 11. Tandem traffic categories 87

Figure 12. Exponential distribution against truncated data 98

Figure 13. Fitting call duration data to a Mixture of Lognormals 99

Figure 14. Exponential and extreme value distribution test 103

Figure 15. Weibull distribution test 103

Figure 16. Log Normal distribution test 104

Figure 17. Log-logistic distribution test 104

Figure 18. Cox-Snell residuals for Log-logistic, Log-normal, and Generalized gamma 109

Figure 19. Call Cease Rate function (t <1000) 111

Figure 20. Call Cease Rate function (t <300) 113

Figure 21. Call arrival pattern for a typical week 115

Figure 22. Fitting actual call arrivals to the suggested model 122

Figure 23. Predicted against actual call arrivals for two random weeks 123

Figure 24. Fitted Gaussian model μ(t)against collected data 133

Figure 25. Residuals against time 135

Figure 26. Residuals against the predicted values 136

Figure 27. Predicted against actual call arrivals for two random weeks 137

Figure 28. Comparison between the Poisson and Gaussian Models 138

Figure 29. Functional Modules of VSIM 142

Figure 30. VSIM high level design 143

Figure 31. Sample collected call data 145

Figure 32. Sample modeling output (NHPP model estimation) 147

Figure 33. Sample modeling output (NHPP model graph) 148

Figure 34. Residuals Vs Time 149

Figure 35. Histogram of call holding time 150

Figure 36. Call holding time modeling 150

Figure 37. Prediction using NHPP model 151

Figure 38. Traffic prediction using the Gaussian/normal model approximation 152

Figure 39. Call arrival data analysis and modeling 159

Figure 40. VSIM output: effect of the number of IP trunks on GoS 160

Figure 41. VSIM simulation algorithm 162

Figure 42. Simulated call arrival rate 165

Figure 43. Simulated Vs Estimated GoS (Exponential special case) 167

Figure 44. Number of IP trunks as a function of time (resource time function) 170

Figure 45. Blocking probability Vs system capacity (switch A) 172

Figure 46. Blocking probability Vs system capacity (switch B) 173

Figure 47. Blocking probability Vs system capacity (switch C) 174

LIST OF TABLES

Table 1. VoIP frame 33

Table 2. Vocoding and VoIP overhead 34

Table 3. VoIP Quality Measurement 40

Table 4. Theoretical limit of VoIP call capacity (Max Call Load) 44

Table 5. Call arrival process modeling approaches 78

Table 6. Call holding time modeling approaches 79

Table 7. Hazard and Density functions 102

Table 8. Estimated parameters for $\lambda(t)$ 121

Table 9. Parameters' estimation and std. errors 132

Table 10. ANOVA model significance 133

Table 11. Normality test results 134

Table 12. Simulated Vs actual IP trunk requirements 168

LIST OF ACRONYMS

AIC Akaike Information Criteria

ANOVA ANalysis Of Variance

ARCH AutoRegressive Conditional Heteroscedasticity

BCMP (Queuing network named after the authors of the paper who first described it)

BHT Busy Hour Traffic

BIC Bayesian Information Criteria

BP-EMLM/BR Batched Poisson EMLM with Bandwidth Reservation

bps bit per second

BR Bandwidth Reservation

BSBH Busy Season Busy Hour

CAC Call Admission Control

CC Call Congestion

CDF Cumulative Distribution Function

CDMA Code Division Multiple Access

CDR Call Detail Record

CEF	Cisco Express Forwarding
CHT	Call Holding Time
CPU	Central Processing Unit
DS0	Digital Signal 0
DSL	Digital Subscriber Line
EDGE	Enhanced Data rates for GSM Evolution
EM	Expectation-Maximization
EMLM	Erlang Multirate Loss Model
EnMLM	Engset Multirate Loss Model
FCS	Frame Check Sequence
GoS	Grade of Service
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
i.i.d	independent and identical distributed
IFG	Inter Frame Gap
IP	Internet Protocol
IRV	Interactive Voice Response

ISDN	Integrated Services Digital Network
ITU-T	International Telecommunication Union-Telecommunication
LS	Least Square
MLE	Maximum Likelihood Estimation
MMPP	Markov Modulated Poisson Process
MOS	Mean Opinion Score
MTU	Maximum Transmission Unit
NFS	Network File System
NHPP	Non-Homogeneous Poisson Process
PCM	Pulse Code Modulation
PMF	Probability Mass Function
PPP	Point to Point Protocol
pps	packet per second
PRB	Physical Resource Blocks
PRI	Integrated Services Digital Network
PSTN	Public Switched Telephone Network
QoS	Quality of Service

RSVP	Resource-Reservation Protocol
RTP	Real-time Transport Protocol
SIP	Session Initiation Protocol
SLA	Service Level Agreement
SOHYP	Sum of Hyper-exponential
SS7	Signaling System No. 7
TC	Time Congestion
TCP	Transmission Control Protocol
TE	Traffic Engineering
TFTP	Trivial File Transfer Protocol
TOA	Time of Arrival
UDP	User Datagram Protocol
VoIP	Voice over Internet Protocol
VSIM	VoIP traffic engineering Simulator
WAN	Wide Area Network

1 Introduction

The wide deployment of high-bandwidth, reliable, and cost-effective IP networks is pushing towards a major paradigm shift in the telecommunications world. The wired and wireless industries are heading towards an All-IP backbone for their future networks. According to Wikipedia [1], Voice over Internet Protocol (VoIP) is "*a general term for a family of transmission technologies for delivery of voice communications over IP networks*". According to a report from "*US Business VoIP Overview: Optimization Trumps Expansion"* [2] in January 2010, 42% of US businesses at the end of 2009 had a VoIP solution in at least one business location. Furthermore, VoIP growth among US businesses will increase rapidly over the coming few years, reaching 79% by 2013. The same report predicts that the revenues of Broadband IP Telephony will continue to grow and will be more than double by 2013. In-Stat released a report in March 2010 that sheds light on the penetration of VoIP through the US government sector [3]. The report indicates that 48% of the government agencies under survey have VoIP solution deployed in at least one location.

Telecommunication system traffic engineering, also known as "Teletraffic engineering", is defined in Wikipedia [4] as the "*application of traffic engineering theory to telecommunications. Teletraffic engineers use their basic knowledge of statistics including*

queuing theory, the nature of traffic, their practical models, their measurements and simulations to make predictions and plan for telecommunication networks at minimum total cost". Throughout this research, we will use the general term Traffic Engineering (TE) instead of Teletraffic Engineering to discuss traffic on PSTN, and VoIP networks. Traffic engineering provides the tradeoff between service and cost. Traffic engineering of the traditional circuit-switched PSTN networks passed through many phases and reaches the maturity with mathematical models that could efficiently capture the parameters and behavior of the traditional telecom process.

IP networks are packet-switched rather than circuit-switched. In order to transport voice conversations over IP networks, the voice stream must be broken down into transportable units which are then encapsulated into IP packets. The resources and characteristics of IP networks are different from these of circuit-switched networks. For example the major resource in a circuit-switched network is the number of circuits (trunks). Each phone call is assigned to a separate circuit (trunk) for the duration of the call. This scheme does not provide the optimal utilization of resources since the circuit is reserved for the call regardless of whether or not voice is being exchanged on that circuit. The circuit switched scheme is easier to engineer since the resource requirements for each call can easily be calculated and allocated. Therefore, the resources required for a VoIP call are not clearly identified. We provide a deep analysis for VoIP resource requirements and provided a new metric that can be used to quantify the number of calls that can be carried on a VoIP network.

1.1 Motivation

It has been realized that the traditional traffic engineering models fall short from capturing the traffic characteristics on modern telecommunication networks. The popularity of IP telephony, the wide spread of wireless services, and the drop in phone call prices have significantly affected the phone usage and hence resulted in different traffic patterns. Many other research studies and our traffic data show that the traditional traffic engineering approaches are inadequate for modern telecommunication systems.

The most common traffic engineering model is the Erlang-B model. This model was developed in 1920 and was widely used to engineer PSTN networks. The Erlang-B model is based on Poisson call arrival process and negative exponential call holding time. It has been proven through many studies that the Poisson and exponential distributions cannot capture the characteristics of traffic on modern telecommunication systems. However, none of the previous studies examined the traffic on a large-scale VoIP system and provided a complete model that can be used to design such systems. In this research we use live traffic data to build and validate call arrival and call holding time models, and we plan to develop a complete traffic engineering model for such large VoIP networks.

1.2 Scope

The scope of this research is limited to providing a traffic engineering model for performance analysis, traffic prediction and resource optimization of VoIP networks. The models of call arrival rate and call holding time can also be applied to any Internet-based telephony, such

as Skype. We use Call Administration Control (CAC) to allow/reject incoming calls, and this is a common approach on the private enterprise and carrier networks. The approach of CAC does not apply to Internet telephony for resource management; however, the call arrival rate and call holding time models will still be the same.

1.3 Goals

The main goal of this research is to provide a modern traffic engineering model for large-scale VoIP networks that enables us to conduct performance analysis, and resource optimization of VoIP systems under different load conditions. In order to do so, we need to study the characteristics of VoIP traffic, and develop new models that can fit the complex modern traffic patterns. After these models are developed, we need to provide simulation-based solution for the traffic problem.

1.4 Contributions

The major contributions of this research are summarized as follows:

- **Literature review of VoIP and PSTN traffic engineering models**

  In this thesis we provide comprehensive literature review for PSTN, and VoIP systems and the traffic engineering and modeling work that has been done in this field. In addition, we provide analyses of pros and cons of various approaches and modeling for VoIP networks.

- **Applying a new traffic measure, Maximum Call Load for VoIP systems.**

  We propose to use the max call load for VoIP networks as a comparable measure to network TDM trunks. Therefore, traffic engineering models can be used to determine the call capacity of VoIP networks. The new Maximum Call Load metric can support the Call Admission Control (CAC) to accept or reject an incoming call request.

- **Using packet per second (pps) in addition to bit per second (bps) in determining network capacity (Maximum Call Load).**

  The traditional calculation of the maximum number of calls is based on network bandwidth, and our experimental research shows that this approach fails to work on some routed networks with high speed links. Our experiments show that packet throughput of network devices (pps) could be a constraint for VoIP traffic as well. When doing traffic engineering for VoIP networks, network engineers should calculate not only the physical bandwidth of network interfaces but also the capacity (measured in pps) of network devices.

- **Framework for modeling call arrival rate as None Homogeneous Poisson Process (NHPP).**

  Our study of hundreds of millions of call data showed that the traditional Poisson approach of modeling call arrival rate include high amount of approximation and errors because it depends on assuming a fixed call arrival rate over the engineering period. We propose to use a NHPP with a variable arrival rate. Furthermore, we present a framework for finding a function of time that accurately captures the call arrival rate. We present statistical and mathematical background and validation for our work.

- **Normality (Gaussian) approximation of call arrival rate under heavy traffic condition**

  We propose a new model for call arrival rate on VoIP tandem networks under heavy traffic conditions. Based on empirical evidence, such call arrival rate can be modeled as linear Gaussian processes instead of NHPP. We show that the Gaussian approach provides intuitive and accurate representation for the call arrival process. The Gaussian approximation allows finding explicit mathematical equations for the model parameters, and provides effective model validation and significance testing. Our work is validated by using hundreds of millions of call records collected from a large-scale VoIP network in the U.S. Our statistical analysis of a few data samples shows that the coefficient of determination, R^2, for the proposed Gaussian model is 0.9973 which means that 99.73% of the variability in the data is explained by the proposed model.

- **Framework for using a survival-analysis approach to model call holding time. The approach introduces a "*Call Cease Rate*" function to model call holding time**

  We present a new approach for modeling call holding time on VoIP networks. Our study of hundreds of millions of call information shows that Erlang B model's exponential assumption is not valid for the modern VoIP networks. We propose a new approach based on time-to-event analysis. We introduce the concept of "call cease rate function" and find a mathematical model for this function based on the captured call data. After studying several models, we found that both the log-logistic and the generalized gamma distributions provide a good fit for the data.

- **Development of a parametric and non-parametric Traffic engineering simulation models for VoIP systems**

  We provide a new VoIP simulation suite that consists of a parametric simulator based on Non-homogeneous Poisson Process (NHPP) call arrival model, and a non-parametric simulator based on real traffic data. Our simulators are validated against real call data obtained from multiple offices of a production VoIP carrier network. The simulation results show that our simulator can provide up to 28% better resource utilization than the legacy Erlang B model. Our simulator can also help carriers dynamically allocate network resources to meet various traffic demands.

1.5 Publications

Journal Articles

- Imad Al Ajarmeh, James Yu and Mohamed Amezziane, "***Modeling VoIP Traffic on Converged IP Networks with Dynamic Resource Allocation***", INTERNATIONAL JOURNAL of COMMUNICATIONS, ISSN: 1998-4480, Issue 1, Volume 4, 2010, page 47-55
- James Yu and Imad Al Ajarmeh, "***Design and Traffic Engineering of VoIP for Enterprise and Carrier Networks***", International Journal On Advances in Telecommunications, vol. 1 no 1, year 2008

Conference Papers

- Imad Al Ajarmeh, James Yu and Mohamed Amezziane, "***G/G/c/c Simulation Model for VoIP Traffic Engineering with non-Parametric Validation"***, The Eighth International Conference on Digital Telecommunications ICDT 2013 April 21 - 26, 2013 - Venice, Italy
- Imad Al Ajarmeh, James Yu and Mohamed Amezziane, "***Modeling Call Arrivals on VoIP Networks as Linear Gaussian Process under Heavy Traffic Condition***", the 7th International Conference on Networks (ICON 2011), Singapore, December 2011
- Imad Al Ajarmeh, James Yu and Mohamed Amezziane, "***Framework for Modeling Call Holding Time for VoIP Tandem Networks***", GLOBECOM 2011, Houston, Texas, December 2011
- Imad Al Ajarmeh, James Yu and Mohamed Amezziane, "***Framework of Applying a Non-Homogeneous Poisson Process to Model VoIP Traffic on Tandem Networks***", 10th WSEAS International Conference on Informatics and Communications, Taipei, Taiwan, August 2010
- James Yu and Imad Al Ajarmeh, **"*Call Admission Control and Traffic Engineering of VoIP*"**, Second International Conference on Digital Communications, San Joes, CA, July 2007, **"Best Paper Award"**

1.6 Thesis Outline

This Thesis is organized as follows: Chapter 2 contains a high-level description of VoIP networks and characteristics of VoIP traffic. The literature review and previous work is

presented in Chapter 3. Chapter 4 contains the research methodology and environment for data collection and analysis. Chapter 5 provides the detailed modeling results and analysis. Chapter 6 presents VSIM design. Chapter 7 includes VSIM simulation details and analysis. And Chapter 8 includes the conclusions and future work.

CHAPTER 2

2 VoIP Networks

In this chapter we present a study for VoIP traffic characteristics and analysis. Also we study different VoIP networks and their resource constraints. Based on the analysis of traffic and networks, we discuss the need of Call Administration Control (CAC) to ensure voice quality.

2.1 VoIP Traffic

VoIP Systems create two types of messages on the IP networks: (a) control traffic (signaling), and (b) bearer traffic (IP encapsulated voice payload). The control traffic is generated by the call setup and management protocols and is used to initiate, maintain, manage, and terminate connections between users. VoIP control traffic consumes little bandwidth and does not require to be included in the traffic engineering modeling. The focus of our analysis is on the bearer traffic.

VoIP encapsulates digitized voice in IP packets. The standard Pulse Code Modulation (PCM) uses 256 quantization level and 8,000 samples per seconds. As a result, we have a

digitized voice channel of 64 kbps (DS0). If we use 20ms sampling interval, each sample will be:

64,000 bps × 20 ms = 1,280 bits = 160 bytes

This digitized voice is then encapsulated in an RTP/UDP/IP packet as illustrated in Table 1

Table 1. VoIP frame

Layer-2 header	IP header (20 Bytes)	UDP header (8 bytes)	RTP header (12 bytes)	Payload (160 bytes)

If the layer-2 is Ethernet, the 802.3 frame header, Frame Check Sequence (FCS), preamble, and Inter-Frame Gap (IFG) add additional 38 bytes. If the layer-2 is Point-to-Point Protocol (PPP), its header and FCS are 7 bytes.

PCM is the standard codec scheme for G.711, and it does not use any voice compression algorithm. If a compression algorithm is used, the bandwidth for a voice channel is reduced to 8 kbps for G.729A and 5.3-6.3 kbps for G.723.1. Some codec schemes employ a silence compression mechanism where the bit rate is significantly reduced if no voice activity is detected. Furthermore, look-ahead algorithms are used in order to anticipate the difference between the current frame and the next one. A summary of voice codec schemes is shown in Table 2.

Table 2. Vocoding and VoIP overhead

	G.711 (10 ms sampling interval)	**G.711** (20 ms sampling interval)	**G.729A** (20 ms sampling interval)	**G.723.1** (30 ms sampling interval)
Raw BW in bps [1]	64,000	64,000	8,000	5,300
VoIP Payload (bytes)	80	160	20	20
VoIP overhead (802.3)	78	78	78	78
VoIP overhead (PPP)	47	47	47	47
BW in bps (802.3) [1]	126,400	95,200	39,200	26,133
BW in bps (PPP) [2]	101,600	82,800	26,800	17,867

2.2 VoIP Network Types

VoIP traffic can be divided into five different categories based on users

1. Residential: This is VoIP service for home users, and it is also referred to as Broadband phone. This service is enabled by the wide deployment of broadband Internet connections such as DSL or cable at homes as shown in Figure 1. Traffic generated by residential VoIP calls tends to peak during the evenings when people are back home. The Internet service provider subscribes a certain number of simultaneous SIP sessions (SIP trunk) to the VoIP provider (S1 in Figure 1). The limiting resource in this network is the number of trunks (usually ISDN-PRI links) between the provider and the PSTN.

[1] The bandwidth (BW) is for one voice channel

[2] Required Bandwidth including the overhead based on the codec packet sampling rate

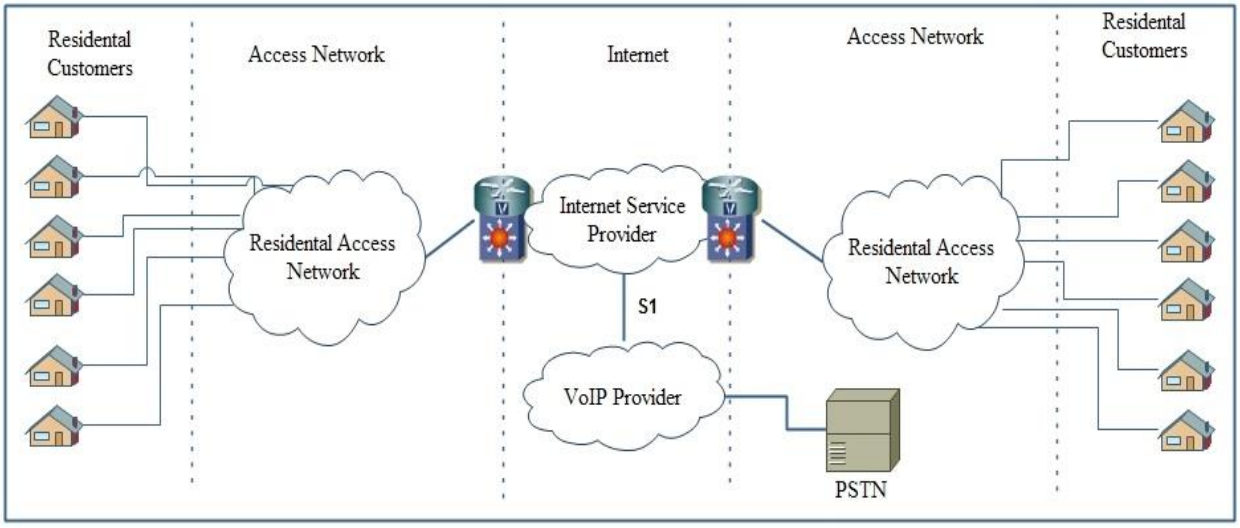

Figure 1. Residential Internet-based VoIP

2. Enterprise [intranet]: This is a VoIP network within the company intranet. Many companies deploy local VoIP solutions over their existing intranet aiming to reduce the cost of phone calls within the company. Such solution is more efficient for larger businesses with multiple locations especially if some of these locations are located outside the country. Traffic generated by such systems is easily predicted and tends to peak in the mornings. Figure 2 shows a basic Enterprise VoIP system. The limiting resource in this network is the WAN connections to the remote offices (W1, W2 and W3). The number of trunks to the PSTN (N1 and N2) could be another limiting resource for the traffic from and to PSTN.

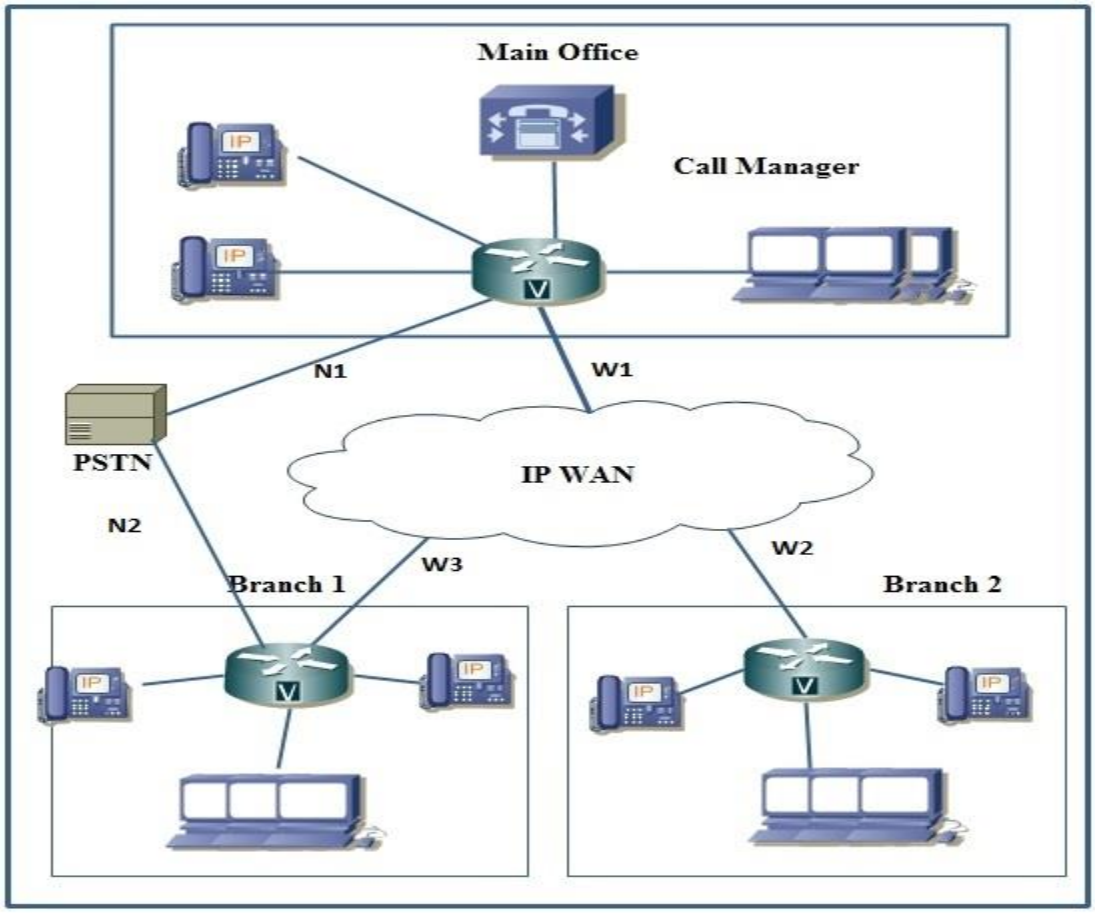

Figure 2. Enterprise VoIP architecture

3. Carrier-to-Enterprise Network (aka SIP Trunking service): This VoIP network provides business customers subscription for SIP trunk groups with a VoIP carrier or service provider. A typical business VoIP solution is illustrated in Figure 3. The service provider owns and runs the VoIP system and network, and the business customer subscribes via trunk groups with a certain capacity. Traffic generated by such systems is easily predicted and tends to peak in the mornings of the business days.

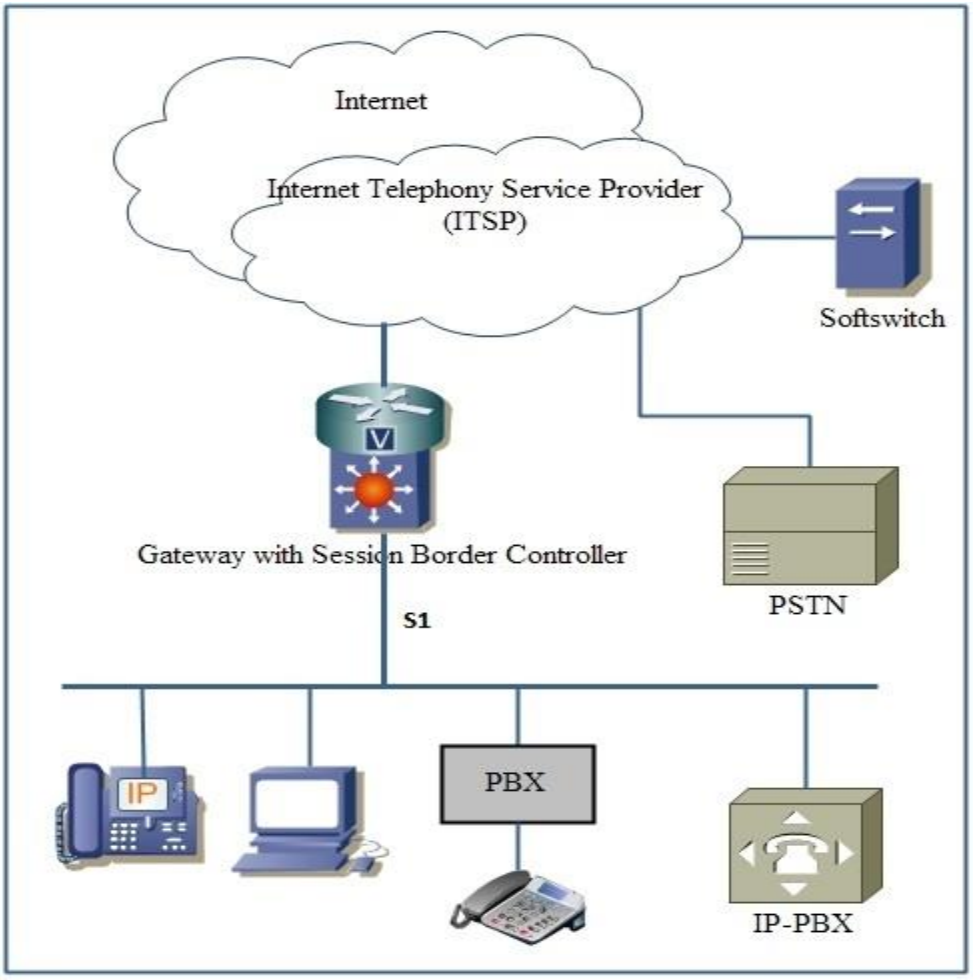

Figure 3. SIP Trunking VoIP solution

The limiting resource in this configuration is the number of sessions in the SIP trunk (capacity of the trunk group, shown in Figure 3 as S1). If an enterprise subscribes too few trunks, the end-user would experience a high probability of blocking, for both incoming and outgoing calls. If the enterprise subscribes too many trunks, many of them will not be used resulting in poor resource utilization and waste of money.

4. Carrier Networks: Networks built to carry voice calls over IP networks. Usually, such networks are large and carry huge amount of traffic. Traffic on these networks tends to be a

mixture of business and residential, so it is harder to predict and needs more complex mathematical modeling. Many carrier networks provide Service Level Agreement (SLA) guarantees for the voice traffic and hence proper traffic engineering for these networks becomes very important and business-critical. Figure 4 shows a typical VoIP carrier network that spans multiple cities.

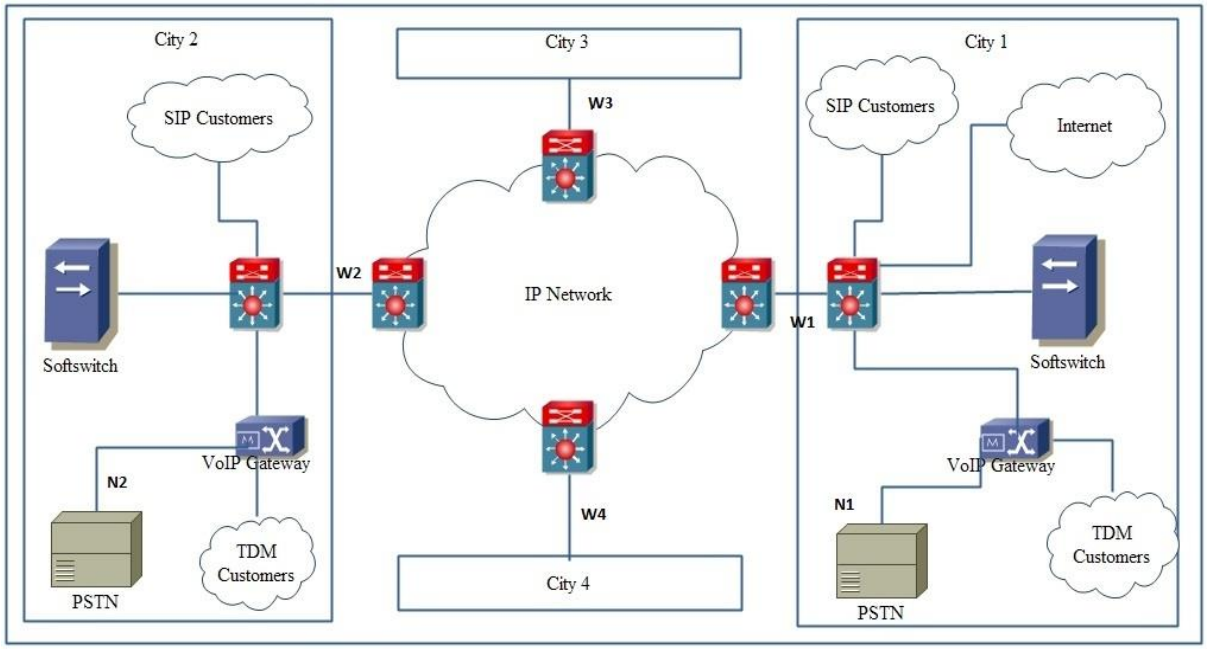

Figure 4. VoIP carrier network

Large-scale carrier networks have more than one limiting resource: (i) the WAN connections between the central offices (W1, W2, W3 and W4), (ii) the number of trunks to the PSTN, and (iii) the capacity of the softswitch and VoIP Gateways. In this research, we focus on the WAN connections described in (i) which is considered the most important limiting resource of such VoIP network.

5. Call Centers: call centers are a special case of business users category with three major differences: (i) unlike regular business phone systems, call centers have automated attendant systems and waiting queues where calling customers can interact with the automated system and/or wait for the next available agent. (ii) calls to call centers tend to be relatively longer. (iii) call centers usually operate even outside the business hours or days. The limiting resource on a call center system is the number of agents servicing customer calls.

2.3 Call Admission Control (CAC)

The purpose of Call Admission Control (CAC) is to determine whether or not the network has sufficient resource to route an incoming call. In the circuit-switched networks the Call Admission Control algorithm is simply to check if there are circuits (or trunks) available between the origination switch and the termination switch. VoIP traffic is carried over packet-switched networks, and the concept of circuits (trunks) is not applicable. However, the need for Call Admission Control (CAC) for VoIP calls is the same. Packet switched networks, by nature, accept any packet regardless of voice or data packets. When the incoming traffic exceeds the network capacity, congestion occurs. Control mechanism is needed to address the issue of congestion by traffic shaping, queuing, buffering, and packet dropping. As a result of this procedure, packets could be delayed or dropped. Delay is usually not an issue for data-only applications. Packet loss can also be recovered by retransmission which is supported by many protocols such as TCP or TFTP. However, retransmission would cause longer delay which is not acceptable to time-sensitive applications such as VoIP. For voice traffic, delay and packet loss

would degrade the voice quality, which is not acceptable to end-users. ITU-T standards provide the following guideline for the voice quality measurement [6]. Table 3 shows a summary of ITU-T guidelines for VoIP:

Table 3. VoIP Quality Measurement

Network Parameter	Good	Acceptable	Poor
Delay (ms)	0-150	150-300	> 300
Jitter (ms)	0-20	20-50	> 50
Packet Loss	0-0.5 %	0.5-1.5%	> 1.5%

The standard voice quality measurement is the Mean Opinion Score (MOS) where different voice samples are collected and played back to a group of people who rank the voice quality between 1 and 5 (1 is the worst and 5 is the best). An MOS of 4 or better is considered toll quality. The objective of Call Admission Control is to prevent network congestion so that all calls could achieve toll quality or better.

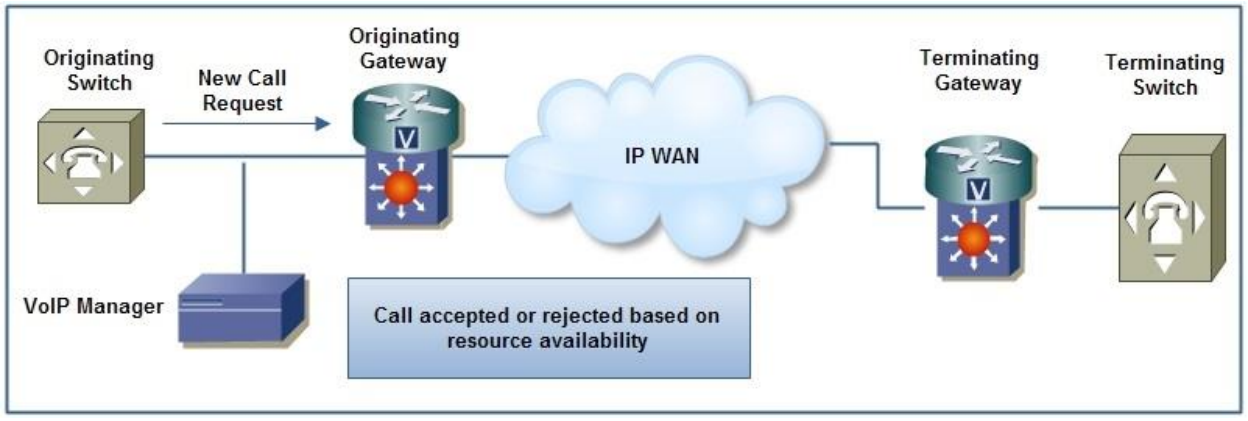

Figure 5. Call Admission control for VoIP system

Figure 5 shows a simple CAC for a VoIP system. It should be noted that that CAC is different from Quality of Service (QoS) as frequently referenced in the literature. The main difference is that QoS is a priority scheme to differentiate the traffic that is already on the network, while CAC is to police the traffic from coming to the network when the network is congested [7]. CAC for circuit-switched network is implemented in the Q.931 and SS7 signaling protocols. Q.931 is used to determine if there is a free B channel in the ISDN trunk and reserve that channel for an incoming call. SS7 signaling is to identify a free DS0 channel between central office switches and reserve that DS0 channel for an incoming call. Although VoIP is on a packet-switch network; however, voice communications still require end-to-end connections to guarantee the voice quality. There are many publications about ensuing voice quality over IP networks, and the general approach of Call Admission Control is to reject a VoIP call request if the network cannot ensure the voice quality. CAC mechanisms are classified as measurement-based control and resource-based control.

Measurement-based Control: For measurement-based control, monitoring and probing tools are required to gauge the network conditions and load status in order to determine whether to accept new calls or not [66]. A protocol, such as RSVP, is required to reserve the required bandwidth before a call is admitted into the network.

Resource-based Control: In the case of resource-based control, resources are provisioned and dedicated for VoIP traffic.

Those two mechanisms are also referenced as link-utilization-based CAC and site-utilization-based CAC [9]. Another reference of these two methods is measurement-based CAC

and parameter-based CAC [10]. In both CAC methods, voice quality of a new call and existing calls shall be assured after a call admission is granted.

Traditional CAC approaches make their decision without taking into consideration of the priority of the requested service. It is difficult to use such CAC systems for networks that provide differentiated services with different priorities. For example a network may have voice, video, data, interactive data, and signaling traffic. Therefore, some recent researches focus on providing priority-based CAC algorithms in which service priority as well as requirements is taken into consideration. Bae et al (2009) [11] proposed an adoptive resource-based CAC algorithm for packet-switched IP-based mobile network. Whenever resources are needed to satisfy a service request, the CAC algorithm estimates the required number of Physical Resource Blocks (PRBs). Other factors are considered in determining the number of PRBs such as the service type, and modulation and coding scheme (MCS) level. The goal of this CAC is to guarantee QoS requirements for packet delay in the packet-switched network. Dandan et al (2007) [12] proposed another adoptive CAC algorithm for CDMA networks. The algorithm determines the required resources based on the service requirements and the priority of the traffic.

2.4 VoIP Call Resources and Admission Control

It is common to calculate the resources needed for VoIP calls based on bandwidth requirements alone [9]. In this section we present a study for the VoIP call resource requirements in which we show that there are other resources that must be taken into consideration. Resource requirements studied in this research are model-independent. This

means these resources are the same whether traffic engineering is based on Erlang models or any other models. For example, the Erlang-B model uses traffic intensity and Grade of Service (GoS) to determine the number of trunks in circuit-switched networks. VoIP, however, is carried over packet-switched networks, and network capacity is measured in bits per second instead of the number of trunks. We studied different network designs for VoIP, and proposed a Call Admission Control (CAC) scheme based on network capacity. We then proposed a new measurement scheme to translate network bandwidth into the maximum call load. With this new metric, resource requirements in traffic engineering models such as number of trunks in Erlang-B model become applicable to VoIP. We conducted experiments to measure the maximum call loads based on various voice codec schemes including G.711, G.729A, and G.723.1. Our results show that call capacity is most likely constrained by network devices rather than physical connections. Therefore, we recommend considering both packet throughput (pps) and bit throughput (bps) in determining the max call load. If network capacity is constrained by packet throughput, then codec schemes would have almost no effect on the maximum call load.

2.4.1 Empirical Results and Analysis

We emulated VoIP in the lab over different links. The expected results (theoretical limit) are calculated based on the overall bandwidth requirements for each codec shown in Table 2. Table 4 shows a summary of the theoretical maximum call load for different codec schemes on different links.

Table 4. Theoretical limit of VoIP call capacity (Max Call Load)

Links	**G.711** (20ms)	**G.711** (10ms)	**G.729A** (20ms)	**G.723.1** (30ms)
FD FT1 (768k)	9.3	7.6	28.7	43
FD E1 (2.0M)	24.2	19.7	74.6	111.9
FD 2×E1 (4.0M)	48.3	39.4	149.3 [1]	223.9 [1]
10BaseT (HD)	52.5	39.6	127.6 [1]	191.3[1]
10BaseT (FD)	105		255.1	382.7
100BaseTX (FD)	1,050	791.1	2,551	3,827

We compare the experimental results with the theoretical limits presented in Table 2 using the following metric:

Utilization = experimental result ÷ theoretical limit

This new metric is to measure the efficiency of a link for voice calls, and it is different from the traditional measure of data throughput and link utilization.

The first experiment is a VoIP traffic test over a full duplex 10/100BaseTX link. The key measurement is the maximum number of simultaneous calls with toll quality (max call load). When we tried to run this experiment over the 100BaseTX link, the CPU utilization of the Linux machine reached 98%. Therefore, the experiment of 100M is considered not applicable for measuring the max call load. The second experiment is to test the VoIP traffic over a serial link with two routers; we configured the link speeds to 768Kbps, 2Mbps, and 4Mbps. The third

[1] Note that a Full Duplex Serial link of 4.0M carries more calls than a half-duplex 10BaseT link because PPP has less overhead than Ethernet.

experiment is to emulate VoIP over three routers with 10BaseT link (half duplex). During the experiment run, we also monitor the CPU utilization of traffic transmitter and receiver. The CPU utilization on the transmission side is 40% for G.723.1 and G.729A and 20% for G.711. The utilization is much lower on the receiver side, less than 10% in all cases. The fourth experiment is to emulate VoIP over a routed full duplex 100BaseTX link. In this experiment, we used a Linux-Based router on a Pentium 4 machine, and the CPU utilization for sender and receiver is less than 40% in all cases. A summary of the observed maximum call loads versus expected (theoretical) maximum call loads is shown in Figure 6.

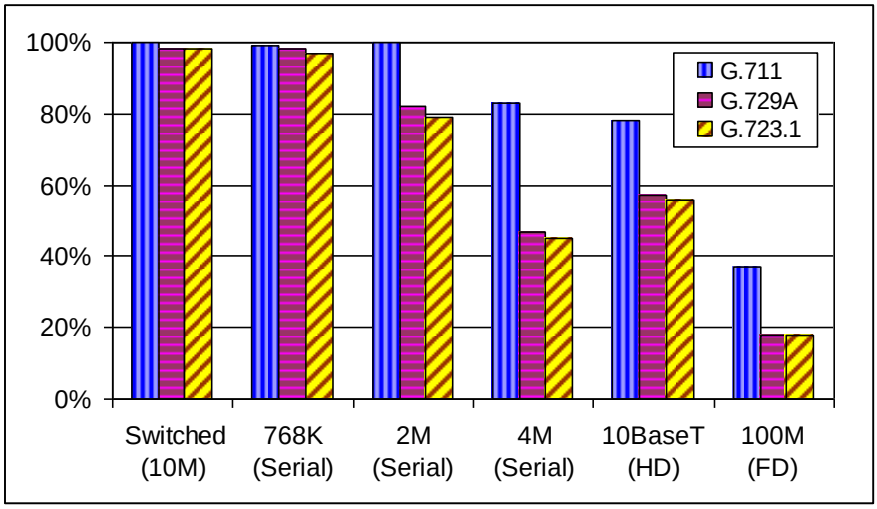

Figure 6. Call utilization for various links

The observations from these experiments are summarized as follows:

- We are able to achieve line speed performance (96% or better) using the max message size in all experiments. This result confirms the validity of the measurement tool and the experiment process.

- The data shows close to 100% utilization on 10BaseT switched Ethernet. It shows that we could achieve the max call load as calculated from the available bandwidth.
- In the cases of routed networks, we observed close to 100% utilization only on low speed links, but poor utilization on high speed links. It shows that the max call load cannot be achieved on the high speed links.
- G.711 always yields better utilization than G.729A which is comparable to G.723.1. It shows that the smaller size for a codec scheme would yield less utilization on the link. This is an interesting result, and we will investigate further later.
- Although G.729A and G.723.1 compress the voice payload by a factor of 8-10, their improvement to the max call load is less than 10% on high speed links.
- When using larger packet sampling rates (from 10ms to 20ms), we notice significant increase in the Max Call Load.

In summary, the experimental results raise a question about how to measure max call loads for VoIP. Many other studies calculate the call load based on the bit throughput (bps), and our experiment shows that bps alone could not explain the results observed in the experiment as there is a large discrepancy between observed data and calculated data.

2.4.2 Packet Throughput and Maximum Call Load

Our lab experiments show that in the case of low utilization, it always involves routers. This observation leads to the study of packet throughput (number of packets processed per second) of network devices. The routers used in this experiment are Cisco 2610 and Cisco 2620. According to the product specifications, these routers are able to carry 1,500 packets per second

(pps). If Cisco Express Forwarding (CEF) is enabled and the traffic pattern is applicable, the router could achieve 15,000 pps. Each VoIP call requires two connections (one in each direction) and this is the symmetric characteristic of VoIP traffic. The way pps is calculated for router is that each packet is counted twice as it goes through the incoming port and the outgoing port. If we use 20ms sampling interval and 64-byte frames, the calculated max call load of a router would be:

$$15{,}000 \text{ pps} \div (1000 \text{ sec} \div 20 \text{ ms}) \div 4 = 75 \text{ calls/sec}$$

And for 30ms sampling interval (G723.1) we have:

$$15{,}000 \div (1000 \div 30) \div 4 = 112 \text{ calls/sec}$$

These numbers are consistent with all the experimental results of the routers. In other words, the max call load is bounded by the router "capacity" rather than the link capacity. We also noticed that we were able to achieve maximum utilization on the physical links for the baseline tests (using MTU as the packet size). The inconsistency in utilization leads to the question about the root cause of difference between the baseline tests and emulated VoIP tests. To answer this question, we need to study the VoIP traffic characteristics explained in Section 2.1 and compare with the processing of packets by network devices. We find that VoIP uses small packet size to transfer calls. In order to achieve higher link utilization using small packet size, we need to send more packets per second. Pushing more small packets into the network would not cause congestion on the link itself; instead, the routers may not be able to process the demand and become the congesting point.

As an example, the frame size of G.729A is 98 bytes (or 784 bits, see Table 2). If we want to achieve full link utilization (10 Mbps) using G.729 codec, we need packet throughput of:

10,000,000 bps ÷ 2 ÷ 784 bit/packet = 6,377pps

Since VoIP traffic is symmetric in both directions, we need the network to handle twice this amount. According to the product specification, each packet is counted twice as it goes through the router (coming and leaving). Therefore, the required packet throughput for the router is:

$$6{,}377 \times 2 \times 2 = 25{,}508 \text{ pps}$$

Given that our router (Cisco-2600) is capable of processing 15,000 pps. Because of this constraint, we observe a lower link utilization which is:

$$15{,}000 \div 25{,}508 = 58.8\%$$

This calculated utilization is almost identical to our experimental results of 57% as presented in Figure 6 . This example of calculation is applicable to all the results we obtained in this research. It proves our point that the limiting factor (bottleneck) is on the router's capability to process packets rather than the network itself. Therefore, to provide sound traffic engineering for VoIP we need to consider pps as well as bps.

2.5 Summary

We propose to use the max call load for VoIP networks as a comparable measure to network trunks. With this modification, traffic engineering models can be used to determine the call capacity of VoIP networks. Packet-switched networks, by nature, do not have the concept of blocking, and all incoming packets are accepted even if the new packets will cause congestion on the network which could result in delay and packet loss. In the case of VoIP, this will cause

quality degradation to the new calls as well as to the existing ones. The solution to this problem is to use a Call Admission Control (CAC) where call manager or softswitch can apply a traffic engineering model to implement a CAC algorithm to accept or reject an incoming call request.

The traditional calculation of the maximum number of calls is based on network bandwidth, and our experiments show that this approach fails to work on some routed networks with high speed links. Our experiments show that packet throughput of network devices (pps) could be the constraint for VoIP traffic. When doing traffic engineering for VoIP networks, network engineers should calculate not only the physical bandwidth of network interfaces but also the capacity (measured in pps) of network devices. If the device capacity is the limiting factor, codec schemes would have no effect on the call capacity; instead, packet sampling interval could significantly change the maximum call load.

CHAPTER 3

3 Literature Review of Traffic Models

In this chapter we provide a review for the work that has been done in the traffic theory and traffic engineering for VoIP as well as PSTN.

3.1 Telecommunication System Modeling

A telecom system can be viewed as a service center in which customers arrive at a service point, get serviced and then depart the system. Some systems have a single server others have multiple servers. Some systems allow customers to wait if they arrive while the server(s) is busy such as call centers. Other systems don't provide any waiting space or mechanism (block customers) such as regular residential or business phone systems. For the purpose of this study we assume the telecom system has multiple servers and no waiting queue is provided (blocking system).

According to this analogy, the system can be divided into three major components: the arrival process, the service time process, and the server(s). Analyzing such system involves

studying statistical queuing techniques based on theoretical distributions and simulations. In a basic telecommunication queuing system, theoretical distributions are used to describe the call arrival rate and the call holding time. Figure 7 shows a basic telecommunication system model.

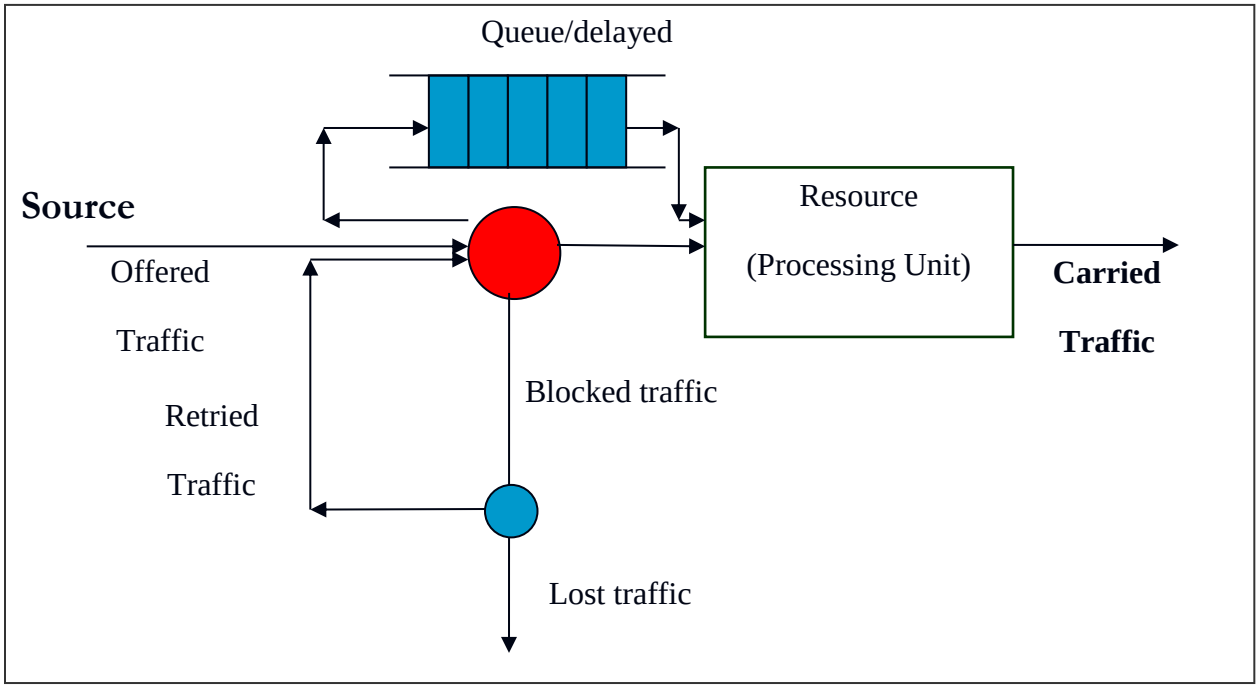

Figure 7. Telecommunication system model

Call arrival rate and call holding time are not deterministic values. Therefore, the first step in analyzing telecom systems is to find the probabilistic models that best approximate these processes.

If simple models and relations are chosen to approximate the call arrival and call holing time distributions, we might be able to use mathematical methods to obtain equations that can be used to estimate the system parameters and this is called an analytical model or solution for the system. However, if complex models and relations are chosen, an analytical model for the

queuing system might not be feasible and in this case simulations are used to estimate the required parameters. Simulation systems utilize computers to evaluate a model and estimate parameters numerically.

Both analytical modeling and simulations have their own advantages and disadvantages. Analytical modeling yields mathematical formulas that can be used to estimate system parameters directly. However, it is not always possible to perform analytical modeling in order to obtain such formulas. Even if mathematical formulas are obtained, it is also important that these formulas can be computed efficiently. It should be noticed that the modeling process that yields simple analytical models usually involves a lot of approximations which might yield inaccurate results especially in systems with complex traffic patterns. Simulations can be easily modified and adapted to any system. Also simulation models require less approximation and could yield more accurate results in many cases. Furthermore, simulations can be tuned to achieve arbitrary accuracy for the estimated parameters. On the other hand, simulations might take a long time depending on the complexity of the system and the required accuracy [13]. Also simulation might suffer insufficient level of abstraction which limits the ability of a simulation model to explore only a limited portion of the system behavior and characteristics. It is desirable to study systems using analytical models whenever such models are available.

3.1.1 Call Arrival Process

The goal of studying call arrival process is to determine the behavior and rate of calls arriving at the system. In other words we need to know that during the next t seconds k calls will arrive at the system with a probability of $p(k,t)$. It is very common to model the call arrival rate using a *Poisson distribution,*

$$p(k,t) = \frac{(\lambda t)^k}{k!} e^{-\lambda t} \quad \textit{for } k = 0,1,2,\ldots \tag{1}$$

where λ is the key parameter for the distribution, and it determines the shape and indicates the average number of events in the given time interval (rate). Equation (1) is known as *Poisson Distribution* and it is used to model the number of events within a given time interval. Any process that can be described by this distribution is called Poisson Process.

A Poisson Process has some interesting properties that make it attractive for modeling traffic arrival rate in telecommunication systems. Below is a summary of the most important Poisson properties [15]:

- If X_1, X_2, … are random variables representing the inter-occurrence times of a Poisson Process, then $\{X_n, n=1,2,\ldots\}$ are i.i.d (independent and identical distributed) and have exponential distribution.
- Poisson distributions are additive: If two types of Poisson events occur independently of each other (say X_1 and X_2 having the parameter of λ_1 and λ_2 respectively), then the probability that X_1 occur before X_2 is given by $\frac{\lambda_1}{\lambda_1+\lambda_2}$
- Poisson Processes are additive as well: let $Y_1(t)$ and $Y_2(t)$ be two Poisson processes with parameters μ_1 and μ_2 respectively and $Y(t) = Y_1(t) + Y_2(t)$ for $t \geq 0$. Hence, $Y(t)$ is also a Poisson process and its probability P is given as:

$$P[Y(t) = n] = e^{-(\mu_1+\mu_2)t} \frac{[(\mu_1+\mu_2)t]^n}{n!}$$

- Relation between Poisson process and uniform distribution: let *n* Poisson events occur at times $t_1 < t_2 < t_3 < \ldots < t_n$ in the interval *[0, T]*, then the random variables $t_1, t_2, \ldots, t_n$ have the same distribution as the *nth-order* statistics corresponding to the independent random variables $U_1, U_2, \ldots, U_n$, and they are uniformly distributed in the interval *[0, T]*.

3.1.2 Call Holding Time

The second process that we must study is the service time (call holding time). Likewise, we need to be able to estimate the probability of an ongoing call leaving the system (call ends) during the next *t* seconds *H(t).* In traditional telecommunication systems call holding time is usually modeled using a negative exponential distribution:

$$H(t) = 1 - e^{-\mu t} = 1 - e^{-t/\tau} \quad (2)$$

where μ is the call departure rate, and is computed as the reciprocal of the average call hold time ($\mu = 1/\tau$). Equation (2) is known as the negative exponential distribution. This distribution describes the probability of the call remaining time rather than holding time. This is possible because both call holding time and call remaining time follow the same exponential distribution. This property of the exponential distribution is called "*memory-less* property" [16] and it states that at any time (t), the remaining time for the ongoing calls follow the same negative exponential distribution as the original one (with the same parameter μ) regardless of the amount of time each has spent on the system. This property can be proven mathematically as follows: Let $P(Z_n)$ be a negative exponential process such that:

$$P(Z_n \le x) = 1 - e^{-\mu x} \quad (\mu > 0)$$

The probability of call termination after time *t* is given as:

$$P(Z_n \le t + x \mid Z_n > t) = \frac{P(t < Z_n < t + x)}{P(Z_n > t)} = \frac{[1 - e^{-\mu(t+x)}] - [1 - e^{-\mu t}]}{e^{-\mu t}}$$

$$= 1 - e^{-\mu x}$$ (This is a negative exponential distribution)

3.2 Traffic Engineering Models

The traditional approach for telecommunication systems traffic engineering is based on the assumption that the call arrival rate conforms to a Poisson process and call holding time follows the negative exponential distribution. Conformance to a Poisson process means that call inter-arrival times are described by a negative exponential distribution. Traffic that follows these assumptions is said to be random. Traffic randomness means:

1. **Random call arrival**: The arrival time of each call is independent of the arrival times of other calls. The inter arrival time follows a negative exponential distribution.
2. **Random call holding time**: The call holding time of each call is independent from the holding times of other calls and follows a negative exponential distribution.

3.2.1 Traffic Measurement

In circuit-switched networks, the limiting resource is the number of circuits which is also known as trunks (N). Traffic load on the network is measured by Traffic Intensity which is defined as:

Traffic Intensity (A) = Call Rate × Call Holding Time

where call rate is the number of incoming calls during a certain period of time. Call Rate is randomly distributed and assumed to follow the Poisson distribution in the traditional models. Call Holding Time is the summation of (a) call duration which is the conversation time, (b) call initialization and setup (c) ringing time. The measurement unit of Traffic Intensity is Erlang which is the traffic load of one circuit over one hour. For example if a circuit is observed for 30-minute of use in a 60-minute interval, the traffic intensity is 30÷60=0.5 Erlang.

3.2.2 Erlang-B model

The Erlang-B model [17] is the standard to model the network traffic of circuit-switched networks. It is known as the blocked-calls-cleared model, where a blocked call is removed from the system (no waiting queue). In this case, the user will receive an announcement of circuit busy. Notice that a busy announcement is not the same as busy signal, which is the case when the callee is already on the phone. From the perspective of the Erlang-B model, not-answered-calls and busy calls are all considered successful calls. Traffic randomness is the primary assumption in the Erlang-B. In addition to the traffic randomness, the Erlang B model has the following two assumptions:

1. **Infinite number of sources (users)**: The model implies that a large number of users who could make a call through the network. In practice, if the number of potential users is much larger than the number of trunks, this assumption is considered valid

2. **Blocked calls are cleared**: When a call is blocked due to insufficient resources (trunks), the user will get a recording or a fast busy tone. The call request is discarded (cleared) by the network and the user must hang up.

A mathematical formula for the Erlang-B model is derived as follows:

Let *A:* be the random traffic load, *N:* the number of servers (trunks), *k*: index of the number of arriving calls (rate), and P_j: the probability that an arriving call finds *j* ongoing calls in the system ($j \leq N$). P_j: can be expressed as:

$$P_j = \frac{\frac{A^j}{j!}}{\sum_{k=0}^{N} \frac{A^k}{k!}} \quad for\ j = 0,1,2,\ldots,N \tag{1}$$

The blocking probability is defined as the probability that an arriving call is blocked because all (N) trunks are busy. When all trunks are busy no further traffic can be carried by the system and the arriving traffic is blocked and cleared off the system. Blocking probability *B(N,A)* is given by:

$$B(N,A) = \frac{\frac{A^N}{N!}}{\sum_{k=0}^{N} \frac{A^k}{k!}} \tag{2}$$

Equation (2) is known as Erlang's Loss Formula or Erlang-B formula. This formula can be used even if the offered load is larger than the available servers basically because blocked calls will be

cleared off the system. Therefore, it is clear that the telecommunication systems might not carry the entire offered load. Recall that *A* is the offered Load, so the carried load is given as:

$$A_o = A * [1 - B(N, A)]$$

And the lost load (blocked and cleared) is the difference between offered load and carried load and is given as:

$$A - A_o = A * B(N, A)$$

Therefore, Grade of Service (GoS) can be written as:

$$GoS = \frac{A - A_o}{A}$$

Under Erlang-B assumptions, the blocking probability and the Grade of Service (GoS) are equal.

The reason for a call being blocked on a typical circuit-switched network is that all trunks are busy. A GoS of 0.01 shows that there is 1% probability of getting a busy announcement. GoS is a critical factor for calculating the required number of trunks since it represents the trade-off between service and cost. For a local telephone switch, if we set the number of trunks (to the tandem office) equal to the number of subscriber lines, then the switch would have GoS=0 (100% non-blocking) regardless of the traffic load. Of course, this is a hypothetical example as no carriers would have this engineering practice. Different subsystems might have different GoS values on the same telecommunication network. The overall GoS for the whole network is the highest GoS value of the subsystems.

In Erlang-B formula shown in (2) above, if traffic intensity *A* is small compared to the number of trunks *N*, then *A/N* is very small and the denominator in (2) reduces to e^A Therefore, equation (2) can be rewritten as:

$$P_k = \frac{A^k}{k!} e^{-A}$$

Recall that that *Traffic Intensity A = Call Rate × Call Holding Time*

Therefore, the above equation is the same as Poisson equation. In this case the traffic conforms to a Poisson distribution. If we take the limit when the number of trunks (*N*) approaches infinity, we get:

$$lim_{N\to\infty} \frac{A^N}{N!} e^{-A} \ \to 0$$

Thus, if we have Poisson traffic and infinite number of trunks, the blocking probability tends to zero.

It should be noted that the assumptions of the Erlang-B model are transparent to the underlying networks regardless of whether it is a circuit-switched network carrying traditional phone calls, or a packet-switched network carrying VoIP calls. The standard practice is to take a conservative approach in measuring traffic intensity on the Busiest Hour of the Busiest Week/Season (BSBH) in a year. In other words, one should never engineer the network based on the average demand. Instead, it should be based on quasi-peak demand.

3.2.3 Erlang-B model Extensions

Extended Erlang-B model: Erlang-B model is based on the assumption that blocked calls are cleared from the system and it does not take retries into account. Extended Erlang-B model, however, takes into consideration the probability that a blocked user will try again immediately. This probability depends on the *Recall Factor (R_f)* which is a new parameter that has been introduces to traffic model in the Extended Erlang-B model. The mathematical representation for

the Extended Erlang-B is based on using the original Erlang-B formula in an iterative manner. In each iteration, the number of retried calls is calculated based on the *Recall Factor* and the resulting number is added to the initial call load. This process is repeated until all call attempts are satisfied. For example, if the initial level of traffic is represented by A_0 then we use Erlang-B model in the following manner:

Find $P_j = B(N, A)$ and then calculate the probable number of blocked callsB_e:

$B_e = A\,P_j$ and then we calculate the number of recalls *R*:

$R = B_e R_f$ the new offered load is $A_{i+1} = A_0 + R$

Now we return to the first step and keep iterating until we reach a stable value for A.

Engest Traffic: Erlang-B formula was developed based on the assumption that the call arrival rate is independent of the number of calls in the system. Such assumption can be justified only if the number of users (subscribers) is much larger than the number of trunks (infinite number of sources (users)). In practice, this assumption might not hold all the time. There are cases when the number of subscribers is comparable to the number of trunks. In such cases the arrival rate depends on the number of calls in the system. This observation is explained as following: a user can be involved in one call at a time only, hence users who are already involved in calls cannot initiate new calls, and this means that the expected call arrival rate depends on the number of free users who might initiate new calls. Expected call arrival rate is inversely proportional to the number of busy users/trunks. In this case the traffic is known as Engest Traffic and the telecom model is based on finite population assumption.

The blocking probability of a finite source system is less than that of an infinite source system, and the reason is because the arrival rate decreases as the number of busy users/trunks increases.

Blocking probability P_B for Engast traffic is given as:

Let λ_s = call arrival rate per subscriber

k = number of busy users in the system

N = total number of subscribers

R = number of trunks

t_h= mean call holding time

μ = mean call termination rate ($\frac{1}{t_h}$)

$\rho = \frac{\lambda_s}{\mu}$

$$P_B = P_R = \frac{\rho^R \binom{N}{R}}{\sum_{k=0}^{R} \rho^k \binom{N}{k}}$$

Where $\binom{N}{k}$ is the binomial coefficient and is given as:

$$\binom{N}{k} = \frac{N!}{k!\,(N-k)!}$$

Notice that the offered traffic (arrival rate) is a function of the number of busy users in the system. When we have *k* busy users only *(N – k)* users can generate calls at a rate of λ_s per user. Therefore, the offered arrival rate C_k in case of *k* busy users can be expressed as:

$$C_k = (N-k)\lambda_s for \quad 0 \le k \le R$$

The mean offered traffic *C* can be given as:

$$C = (N - A_o)\lambda_s$$

where $A_o$is the average number of busy trunks. The offered traffic intensity *A* is:

$$A = Ct_h = \lambda_s t_h (N - A_o)$$

The system will be in blocking state when all trunks (*R*) are busy, in other words when $A_o = R$

In this case the offered call rate is $(N - R)\lambda_s$ and all arrival calls are lost (blocked). The lost traffic can be expressed as:

$$A - A_o = (N - R)\lambda_s t_h P_R$$

Unlike Erlang-B model, under Engest traffic assumptions, GoS is not equal to the blocking probability. GoS is given as:

$$GoS = \frac{N - R}{N - A_o} P_R$$

3.3 Other Research on Traffic Models

As indicated in Section 3.2.2, the Erlang-B model is based on the assumptions that call holding time follows a negative exponential distribution and call arrival rate follow a Poisson Process with a constant rate over a certain block of time. A separate queuing model will be provided for each of those time blocks. The exponential approximations are made in order to achieve relative simplicity in the corresponding mathematical and analytical models. Under exponential call inter-arrival assumption, the observed call arrival process consists of the sum of a large number of independent call arrivals. Therefore, we are dealing with memory-less

exponentially distributed events for call inter-arrival time and call holding time. The memory-less property of the process is also referenced as Markov property which is essential for providing analytical solution to the queuing model. Hence, the telephone network is easily modeled as M/M/c/c queuing system and Erlang models can be used to study the performance of the telecommunication system and calculate the required resources. Exponential distribution is used for one-parameter approximation of the data. In the case of call holding time, the rate parameter for the exponential distribution is based on the mean of call holding times, and in case of call inter-arrival time the rate parameter is based on the mean inter-arrival times. On the other hand, the lognormal and Erlang distributions can be used for two-parameter approximations.

Recently, there has been a growing interest in modeling more complex call arrival flows and holding times. This interest is driven by the attempts to solve problems associated with the inadequacy of the exponential assumptions. Such problems affect the design and performance of the system. As we mentioned in the previous section, finding explicit equations for systems with complex arrival flows might be very difficult. When the models used to capture call arrival process or call holding time lack the Markovian property, the analytical approach for performance evaluation is not feasible. Research in this field either tends towards simulations or towards analyzing the system under the condition of heavy traffic (many calls in the system) [23] and low traffic (the system is mostly idle) [24].

3.3.1 Modeling Call Arrival Process

Erlang model assumes that call arrival occurs as a Poisson process, and it implies calls are generated independently by a large number of users [infinite number of user assumption]. In

addition, it is assumed that each user generates at most one call in a given period of time. Therefore, the arrival time of calls is uniformly distributed over that period and hence the call arrival rate is represented by a stationary Poisson process. In practice, the stationary Poisson assumption is often violated since call arrival rate is a function of time and not uniformly distributed over a long period of time. In addition, some users generate more than one call during a certain period of time. Therefore, the results obtained by stationary Poisson-based call arrival models are not accurate. In this section we will study the main approaches that have been proposed to replace the stationary Poisson call arrivals.

3.3.1.1 Batch (Session) Based Call Arrivals

It is common that calls arrive in bursts (batches) in which each call starts after the previous call ends. This case might be handled by assuming that bursts are of fixed size (x) and burst arrivals follow a Poisson process. In this case each burst of size x is treated by Erlang formula as a single call that occupies x lines for the period of the call holding time. Also one of the factors that violate Erlang's stationary Poisson assumption is the fact that many users generate multiple calls rather than one call during a given period of time. This effect can be minimized by introducing the concept of sessions. A session is defined as the sequence of calls generated by a certain user. The batch-based approach is similar to the session-based; however, calls arrive in batches regardless of the user generating the call. Using session arrivals instead of call arrivals enables us to use Erlang models to engineer the network.

In [18] Bonald proposed to model call arrival rate by using the concept of Poisson-based sessions rather than Poisson-based calls. Sessions are assumed to be independent from one another and each session contains a random finite number of calls and idle periods. Bonald

based his work on the observation that although call arrival rate within each session follows a non-Poisson process, session arrival rate follows a Poisson process. As a result, Erlang traffic formula can be used, or follows a permanent nature and hence Engest Formula can be used.

In [68] and [70] the authors studied call arrivals on an IP based network with resource reservation capabilities. They suggest a Batch Poisson call arrival process in which calls arrive in batches, and batches arrive randomly following a Poisson distribution. Batches may have different sizes. They proposed an application for the Bandwidth Reservation (BR) policy of the Erlang Multirate Loss Model (EMLM). The proposed model is named Batched Poisson EMLM under Bandwidth Reservation (BR) policy (BP-EMLM/BR). The model is based on partial batch blocking, i.e. a part of an arriving batch can be accepted while the rest of it is rejected depending on the available link bandwidth [69]. The authors proposed a recursive method to approximate the link utilization with two probability functions: Time Congestion (TC) probability and Call Congestion probability (CC). The authors also considered the case of finite population and Quasi-random traffic. In such case the Engset Multirate Loss Model (EnMLM) can be used instead of the EMLM model.

Session-based model have been used on systems other than IP networks. For example, Hess and Cohn [19] studied the voice traffic behavior in mobile radio systems. They followed a session-based approach and concluded that session inter-arrival time follows an exponential distribution. They suggested a model for peak load estimation and used Erlang-C formula to calculate the required resources for the network.

The advantage of the session approach is that call arrivals within each session can have any arbitrary distribution. In addition, it provides more accurate results than the traditional Poisson call arrival approach. On the other hand, we still have to assume that sessions are

independent from each other and follow a Poisson process. Basically, if the number of users is relatively low, then the session arrival process depends on the number of on-going calls and therefore the Poisson session arrival assumption is violated.

3.3.1.2 Traditional Stationary Poisson Arrival Rate

This is the traditional approach used by Erlang-B model. Call arrivals are assumed to follow a Poisson distribution with fixed rate. We have covered this approach in more details in Section 3.1.1. However, we mention it here since we still find some modern research that depends on this approach. Zvezdan et al. [20] provided a bandwidth calculation method for VoIP networks based on Poisson call arrival rate. Erlang-B and Extended Erlang-B models were used to calculate the required network resources. The authors are aware of the well-known limitations of Poisson approximation. Therefore, they based their calculations on the assumption that the Busy Hour Traffic (BHT) is approximately 17% of the whole traffic for that day. The calculation method takes into consideration factors such as Voice Activity Detection, RTP header compression, and the used codec. The results of the proposed method are validated my Matlab simulation.

Duncan et al. [28] investigated busy period voice traffic for a trunked mobile radio system. They used data aggregated of multiple talk group traffic. The analysis indicated that call inter-arrival time follows an exponential distribution and exhibit certain degree of long-range dependency.

3.3.1.3 Erlang-jk

Erlang-jk is sometimes considered to model call inter-arrival time distributions. It is composed of a mixture of two Erlang distributions with different proportions. In [25], Barceló and Bueno

studied call inter-arrival time for mobile telecommunication networks, and they used two different methods to estimate call inter arrivals. The first applies a filtering process and assumes that in the filtering process all and only calls that have been rejected are eliminated. Therefore, analyzing the inter-arrival time of the samples remaining after filtering, we get the coefficient of variation of the inter-arrival time to the system. The second proposed method for inter-arrival estimation is based on the delay probability estimation which is sensitive to call inter-arrival time [26]. Barceló and Bueno concluded that channel idle time is best modeled by an Erlang-jk distribution. The same model can be used to represent call inter-arrival time after filtering unsuccessful call attempts. This approach has the disadvantage that it does not accommodate for attempt calls and it ignores calls with short duration (bound of the filtering process).

Call inter-arrival time for cellular networks was studied by Sánchez et. al [29]. They based their research on real traffic samples and concluded that call inter-arrival time is far from being exponential Also they concluded that call arrival rate cannot be represented by a Poisson process. Multiple models were examined, and the Kolmogorov-Smirnov (K-S) goodness of fit results indicate that call inter-arrival time is best modeled as Erlang-j-k with j=3.

3.3.1.4 BCMP

A BCMP network (named after the authors of the paper who first described this network) is a heterogeneous queuing network with multiple classes of customers having different distributions. A product form of equilibrium distribution exists for the BCMP network. It is considered an extension to a Jackson network allowing several customer classes and service time distributions. In a recent study for call processing in Intelligent Networks (IN), Irina et al [21] modeled the SS7 signaling traffic as exponential BCMP queuing network. The study provides a method of analyzing the post-

selection delay in the SS7 channel and IN nodes as part of the call setup process. Analytical as well as empirical studies are provided. Signaling arrival rate is modeled as a Poisson Process with fixed rate (λ). The study proposes a mathematical model that can be used to calculate the post-selection delay assuming an exponential service time. However, In the case of general service time distribution in BMCP network nodes, calculating the delay can be achieved using simulations only.

3.3.1.5 Non-homogeneous Poisson Process

Unlike the stationary Poisson process, the non-homogeneous (non-stationary) Poisson process has a rate that is a function of time. In a study of a telephone call center, Lawrence Brown et al. [27] modeled the call arrivals as a time-inhomogeneous Poisson process with piecewise constant rates. Under the proposed model, the duration of the day is divided into short time intervals and during each interval the arrival rate is assumed to be constant. It is not necessary to make all the intervals with equal time lengths. Brown et al studied different types of traffic, and they used a different fixed-length time block for each traffic type. Their decision on the block length was based on the arrival rate. They used smaller blocks (6 minutes) for traffic with high arrival rate and larger blocks (up to 60 minutes) for traffic with low arrival rates. The general rule is to have intervals short enough so that the arrival rate can be assumed constant within each interval. Lawrence Brown used Kolmogorov-Smirnov statistic test to accept the hypothesis that call arrival rate is a non-homogeneous Poisson process. Furthermore, Brown provided empirical evidence that the call holding time of a call center follows a distribution close to lognormal rather than exponential. Although the lognormality hypothesis was rejected by Kolmogorov-

Smirnov test, Brown adopted this distribution based on empirical observations and the used the large sample size for justification.

3.3.1.6 Packet-level Arrival Modeling

Another approach for VoIP traffic engineering is based on the packet level rather than the call level. In this approach mathematical and statistical models are provided for packet arrivals. Bowei et al. [67] provided a packet-level VoIP modeling study in which they collected 332,018 call records from a live network and then they used the empirical data to develop and validate the traffic models. They used the developed models to build a simulator for QoS studies on the IP network. The authors proposed two models for packet inter-arrival times:

(i) Semi-empirical model in which the empirical data is used to construct the model.

(ii) Mathematical model that consists of parametric statistical model. For each call, a call duration is generated (a random variable obtained from a proposed piece-wise Weibull distribution), then periods of transmission and silence are generated. Packet arrivals are inserted every 20ms during the transmission periods, and every 2 seconds during the silence periods.

The authors provide a parametric model for the periods of transmission and silence for systems with silence suppression capabilities. They found that the square-root gamma distribution provides a good and flexible fit for the data. Although Bowei et al. focus on modeling packet arrival process; however, this process is directly related to call arrival process. They consider the call arrival process as a non-homogenous Poisson process for the tow proposed models.

Another result of this research confirms our previous observation of the symmetrical nature of VoIP traffic on the packet level. The authors observed similar traffic patterns from

caller to callee as from callee to caller. The only observed differences are for FAX calls where the transmission is mainly from the sender to the receiver and during the ringing period of regular calls where more packets are sent from callee to caller than in the opposite direction. These effects are minor and can be ignored especially in the large-scale networks.

In another packet-level study, Jiang and Schulzrinne provided analysis for the talk-spurt and silence gap distributions produced by some modern silence detectors. They concluded that the inter-arrival times of talk spurts and the gaps do not follow the traditionally-assumed exponential distribution. Instead, the study suggests heavier tails for both talk and gap distributions. The authors propose a simulation system based on using the real Cumulative Distribution Function for the talk-spurts or gap arrivals. Using CDF is a completely empirical approach, where no model is assumed. Instead, real traffic data is fed to the simulator which in turn computes the CDF and uses it as the call arrival distribution.

3.3.2 Modeling Call Holding Time

It has been recognized that the exponential approximation for call hold time seriously underestimates the long calls [22], and the reason is because the exponential distribution lacks a heavy tail that can accommodate for long-duration calls. The need to fit call hold time into a heavy-tailed distribution is mainly to capture calls with long hold times. It is possible to achieve good fit using a longnormal mixture basically when we truncate the very long hold times "statistical outliners". Although this truncation permits a data fit, it might lead to loss of significant fraction of calls. On the other hand, leaving all the calls including the few extremely

long duration ones leads to infinite variance/mean distribution [16]. In this section we summarize the major approaches for modeling call holding time.

3.3.2.1 Traditional Exponential Call Holding Time

This approach is covered in details in Section 3.1.2, and it is the model assumed by Erlang formulas. We briefly look at some of the recent traffic modeling approaches that adapted this model. The finite population with Quasi-random traffic pattern was also considered in [71]. The authors provided analytical and as well as a simulation for calculating Call Blocking Probability (CBP) under this condition. They considered exponential holding time for both the simulation and the analytical model. They concluded that the accuracy of the calculation was satisfactory compared to the simulation.

In [82], Pareto, exponential, multimodal, and deterministic distributions were compared for call holding times. The study provided simulation comparison for the effect of these distributions on call routing and QoS. The authors concluded that the choice of call holding time model has only a slight impact on the efficiency of QoS routing. The study states that the traditional exponential holding time can be considered as a reasonable approximation because the QoS and call losses are insensitive to the used distribution. As a result, the call holding time is determined by the mean value of the call durations.

3.3.2.2 Lognormal Model

Lognormal distribution provides a heavier tail than the exponential. It is a 2-parameter distribution: location and scale (or geometric mean and geometric standard deviation). Therefore, this distribution attracted researchers to use it for call holding times. In their study of traffic for a

trunked mobile radio system, Duncan et al. [28] concluded that call holding time follows a lognormal distribution and hence suggested that using Erlang models for such traffic may not lead to reliable results.

Jedrzychy and Leung [33] provided another research in which they confirmed that the negative exponential assumption for channel holding time is not correct and that a lognormal model approximation provides a better fit for data. The study was based on real traffic data, and maximum likelihood estimation method was used for model estimation. After model estimations, the authors used chi-square to test the goodness of fit. Duncan et al. [28] also concluded in their research that the call holding time has a lognormal distribution and exhibit no significant correlation structure. They used Kolmogorov–Smirnov test to examine several distributions: exponential, lognormal, gamma, and Erlang, and found that lognormal yield the best fit.

3.3.2.3 Mixture of Lognormals

A mixture of lognormals allows more flexibility to fit calls with more variability in the duration. V. Bolotin (1994) [30], provided an empirical study in which he concluded that call hold time can be best modeled as a lognormal or a mixture of lognormals. The author used Kolmogorov-Smirnov goodness-of-fit test to fit his empirically-obtained sample. In a later work, Chlebus [31] used the more reliable Anderson-Darling test to prove that call holding time for mobile telephony follows the same lognormal patterns obtained by Bolotin for fixed telephony.

Barcelo and Jordan [34] have studded channel holding time for a public cellular telephony network. They made a series of experiments and concluding that the negative

exponential distribution is not a good approximation of the channel holding time. They suggested that the probability distribution that better fits the empirical data was a sum (mixture) of lognormal distributions. This suggestion was supported by Kolmogorov-Smirnov goodness-of-fit test results. It is worth mentioning that channel holding time equals call holding time if the user remains within the same cell.

Barceló a (1999) [34] provided a field study for the channel occupancy in cellular networks. He concluded that the exponential distribution is far from capturing the empirical data. Finally, the author landed on using a mixture of lognormal distributions similar to those found by Bolotin and Chlebus for call holding time.

3.3.2.4 Phase Type Distributions

Phase-type distribution is composed of one or more Poisson processes. These processes are related and occur in a certain sequence (phases). In general, if a system is modeled using exponential distribution and an explicit mathematical solution is found, we can replace the exponential with a phase-type distribution (in order to accommodate for the variability in the data) and still be able to derive mathematical solutions [73].

V. Ramaswami, et al (2003) [35] studied the effect of long holding time for dial-up connections on the call holding time distribution. The study is based on a sample of 4.5 million calls. The collected data showed that the median was only 48 seconds while the mean was 297 seconds. This data failed to fit into an exponential distribution. The authors used the Expectation-Maximization (EM) algorithm to fit the data into a phase type distribution. They concluded that the call holding time is best modeled as a 4-component phase type distribution.

Hyper-Exponential and Hyper-Erlang distributions belong to the phase-type distribution family. Hyper-Exponential distribution provides a mixture density so that it can accommodate for more than one type of calls, such as calls with long duration and calls with short durations. Also Hyper-Erlang distribution is used for mixed type traffic data with heavy tails. Both distributions preserve the Markovian property of the queuing system and hence analytical solutions can be derived [72].

Fang et al (1998) [36] studied call holding time for complete and incomplete calls in PCS networks. They used a general distribution to model and derive general formulas for call holding time for both complete and incomplete calls. They provided analytical study for each of the following distributions: Gamma, (staged) Erlang, hyperexponential and hyper-Erlang.

In another study [16], Fang applied two new distributions to model call holding time. The first is called Sum of Hyper-exponential (SOHYP) model which was previously used to model channel holding time for cellular networks, and the second model is called the Hyper-Erlang model (AKA mixed-Erlang) which was previously used to model the cell residence time for PCS networks. The interesting feature of the Hyper-Erlang and SOHYP models is that they preserve the Markov property which is required for performing theoretical queue analysis. Fang provided a unifying analytical approach to analyze the performance of the resulting queuing system under the assumption that cell residence times are independent and identically distributed (i.i.d.). He provided analytical formulas for handoff probability, handoff rate, call dropping probability, and the actual call holding times for both complete and incomplete calls.

In their study of channel holding time in cellular communication networks, Thomas et. al. [74] based their study on the assumption that call holding times and cell residence times follow

phased-type distributions. The authors derive channel holding time from the phase-type call holding time and cell resident time distributions, and the result is another phase-type distribution for channel holding time. They considered both hyper-Erlang and SOHYP distributions for call holding time. In a similar study of channel holding time, Orlik and Rappaport [75] reached similar conclusion, and they derived channel holding time distribution from the assumed SOHYP call holding time.

3.3.2.5 Weibull and Piecewise Weibull Distributions

Weibull distribution exhibits a heavy tail property when the positive shape parameter is less than one. When the shape parameter equals to one the Weibull distribution becomes exponential. Therefore, Weibull has multiple applications in telecommunication modeling. For example, Weibull distribution has been used to model caller patience factor in call centers [78], and call holding time for internet dial-up connections and World Wide Web sessions [77]. Piecewise Weibull distribution has been used to model a mixture data (data set belonging to multiple categories or classes). Each piece of the distribution corresponds to a category of the data set.

In their VoIP traffic modeling study, Bowei et al. modeled call durations as a mixture of piecewise Weibull distributions. The work is based on 138,770 call information. They fitted the empirical data to a 6-piece Weibull distribution. These different pieces result from the mixture of different types of calls (for example: machine-to-machine, voice call, fax … etc). An interesting point of this study is that it investigates the validity of i.i.d (independent and identical distributed) assumption for call durations. i.i.d assumption is sometimes violated since users tend to use some applications more often during a certain time of the day. The authors provide analysis for the relation between call duration and the time of the day. They found minor effect

of the time of the day on the call duration, and this effect is mainly due to the decrease in the frequency of short calls between 10 PM and 6 AM. They concluded that this effect is minor and can be ignored, and hence the iid assumption holds.

In addition, Weibull distribution has been used to model call holding time in wireless cellular networks [76]. The authors derive a call completion probability function based on Weibull call holding time and cell dwell time. In [79] the authors provide a review of several research papers that employed Weibull distributions to model the call holding time of integrated voice and multimedia packet data services.

3.3.2.6 Pareto Distribution

The Pareto Distribution was first proposed as a model for the distribution of wealth in society. It is a 2-parameter skewed and heavy-tailed distribution. These properties of Pareto distribution attracted some telecommunication traffic researchers to use it for service time modeling. In [80], the authors used Pareto distribution to model cell dwell time. The choice of Pareto was made because of its heavy tail feature. They provided channel holding time statistics based on the Pareto assumption. In the same work, the authors also considered Weibull, and Lognormal distributions for cell dwell time. However, mathematical models were provided for the case of Pareto only.

In [81], the authors considered Pareto distribution for call holding times in their performance analysis for wireless cellular networks. Similarly, in [83] Pareto distribution was considered for call holding times. The authors presented a formula for the probability mass function (pmf) of the number of handovers based on renewal theory arguments and a Pareto distributed call holding time (CHT). The study provides comparison of the system performance

between the Pareto CHT case with cases of Erlang_K and hyperexponential distributions. The study concludes that call holding time is best modeled as Pareto distribution. A major disadvantage of using Pareto distribution is that it is known for being prone to errors and statistical inaccuracies in simulations [81].

3.4 Summary

Many researchers have demonstrated that the traditional traffic engineering models are inadequate for modern telecommunication systems such as wireless, and VoIP. The Poisson call arrival rate and the negative exponential call holding time involve high degree of approximations which will result in systems that are not properly engineered.

The majority of the recent research in this field focuses on modeling either the call holding time or the arrival rate without providing a complete traffic engineering model. An example model is found in [44] where Baynat et al. derived an Erlang-like formula for dimensioning radio resources in GSM/GPRS/EDGE networks. The proposed formula takes into account both the voice and data traffic. According to the authors, the advantage of the proposed Erlang-like model is that it has an analytical solution for the formula, and therefore avoiding the computational complications of the simulations. The model was validated against a simulator and results show close match with the advantage of short computing time for the formula compared to longer time for the simulation.

Table 5 below provides a summary of the major approaches for modeling call arrival process:

Table 5. Call arrival process modeling approaches

Approach	Advantages	Disadvantages
Poisson call arrivals (exponential inter-arrivals)	• Simplicity • Analytical solution	• Inaccurate • Inadequate for modern systems
Session (batched) based arrivals	• Poisson-based session • analytical solutions • Erlang Formulas can be applied	• i.i.d assumption might be difficult to hold • not accurate
Erlang-jk	• More flexible call inter-arrival time	• two Erlangs means more parameters • Added complexity
BCMP	• Mixed queuing network • Poisson arrivals	• Complex • suffers Poisson limitations
Non-homogeneous Poisson	• Flexible • Accurate	• Accuracy of results depends on the time function • No exact analytical solution
Packet-level arrivals	• Takes into consideration VoIP features such as silent detection and codec. • Could provide accurate results, depending on the selected models	• More complex • Obtaining packet information is more difficult than obtaining call information • In case of using one codec and if silent detection is not used, then the relation between call arrival and packet arrival might be too simple so that the added complexity is not justified. • Difficult what-if analysis

Table 6 summarizes the major approaches used for modeling call holding times

Table 6. Call holding time modeling approaches

Approach	Advantages	Disadvantages
Traditional Exponential	• Easy analytical solution • Simple calculations (one parameter distribution)	• Not accurate • Ignores heavy tailed data • One call type (not mixed)
lognormal	• Accommodates for some heavy-tailed data	• The tail is not heavy enough • The tail decays exponentially
Mixture of lognormals	• flexible	• Not general enough to capture wide mixture • Complexity of having a distribution of multiple pieces • More parameters • The tail decays exponentially
Phase-type distribution	• General family that contains multiple distribution for different cases • Preserve Markovian property for analytical solution	• Might be complex • Multiple parameters for multiple phases.
Weibull and piecewise Wibull	• Heavy-tail • Piecewise Weibull provide flexibility to model a mixture of multiple types of calls • Wiebull converges to exponential under special case	• For Piecewise Wiebull, we need to have 2 parameters per piece • Complexity of having multiple pieces
Pareto distribution	• Heavy-tailed distribution that can capture long calls. • Only 2 parameters	• Statistical inaccuracy for simulations

In addition, we do not find studies targeting large-scale VoIP networks, such as Tandem networks. The main characteristics of such network are that it carries huge amount of traffic, and the traffic usually is composed of a mixture of residential, wireless, and business calls. In this research we plan to bridge the gap by providing a deep study for traffic patterns obtained from actual large-scale tandem network. We will provide a frame work for modeling call arrival rate and call holding time, and then we will use the provided frame work to find distribution functions that capture some sample traffic data. A complete traffic engineering simulation model will be provided in order to optimize resource usage on VoIP networks.

CHAPTER 4

4 Research Methodology

In this chapter we discuss the methodology we followed throughout this research. We explain the data collection process and environment, and then we describe the simulation model. Also we cover the mathematical and statistical methods used in to develop the models in this research.

4.1 VoIP Traffic Data from IP Tandem Network

One of the unique features of this study is the quality and quantity of call information from which the proposed models have been developed and validated. This study has been sponsored by one of the major VoIP Tandem carriers in the United States. Therefore, we were given access to billions of call information records in order to develop and validate our models. Traffic carried on tandem networks is composed of wide mixtures of residential, business, and wireless traffic. Using large mixture of traffic enhances the robustness, correctness and usability of the models.

4.1.1 IP Tandem Network

Tandem networks play the backbone roles in the telecommunications hierarchy. They interconnect different central offices together by means of tandem switches. Central offices might belong to the same carrier or to different carriers. In the later case the tandem service provides interconnectivity and switching between different carriers (inter-carrier switching). Therefore, tandem networks are expected to carry large amount of traffic and should be designed for high capacity, high availability, high scalability, and cost efficiency. An IP-Based Tandem service utilizes IP core network instead of the legacy TDM as a transport for the voice traffic. The IP core network could be dedicated for voice only or could be shared between voice and data. Using a converged IP network for data and voice provides substantial cost saving for network design and management. Figure 8 illustrates a typical IP tandem network.

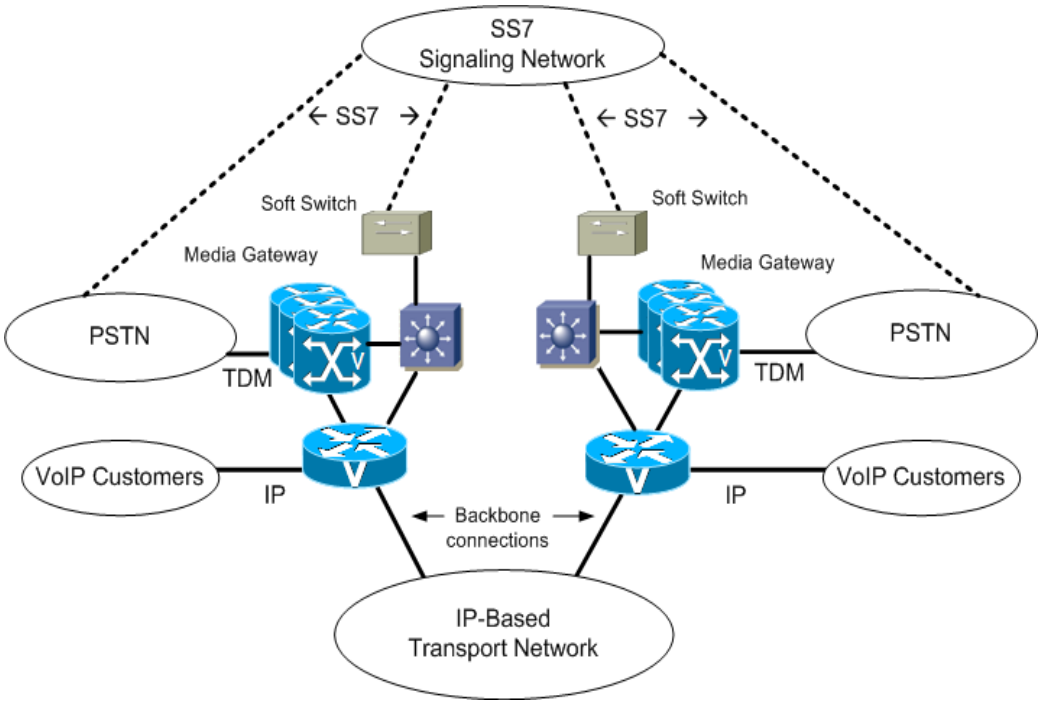

Figure 8. Typical IP-based tandem network

The legacy PSTN is connected through TDM trunks. VoIP customers are connected via IP links. The network has an IP core which is used to interconnect different sites. The limiting resources on the network can be the IP backbone connections between different tandem offices, the IP connections to the VoIP customers, or the TDM connection to the legacy PSTN. The scope of our research is to optimize the first two IP-based resources.

4.1.2 Data Collection and Processing

During this study we have collected several billions of call detail records (CDR's) from the IP tandem network under study. We developed a library of scripts and tools in order to collect the raw data from the different sources and then filter, aggregate, process and visualize the data according to the study needs. Figure 9 shows the number of CDR's collected over three years.

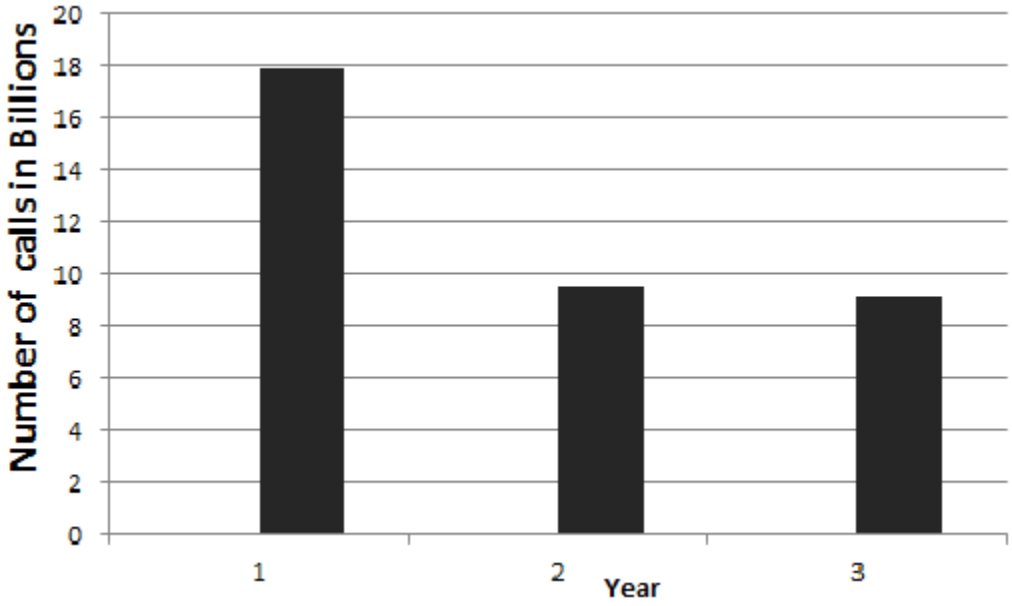

Figure 9. Number of collected Call Detail Records

During the first year of the study, we collected more data because our collection criteria were wide open in order to explore more traffic samples covering more markets and customers. During the second and third years, we narrowed our collection to cover samples of markets with different sizes and different customer base (Business, residential, wireless, VoIP and landline).

A Call Detail Record (CDR) is kept for every call on a local Billing Server (Network File System) located at each tandem office. We installed a CDR Extraction Script (CDR ES) on each of the remote NFS servers. The purpose of the CDR ES is to access the local NFS and extract the CDR fields that we are interested in. Our centralized data collection server executes the CDR ES every day after midnight via SSH and stores the collected data onto a centralized attached storage. Figure 10 illustrates the data collection process.

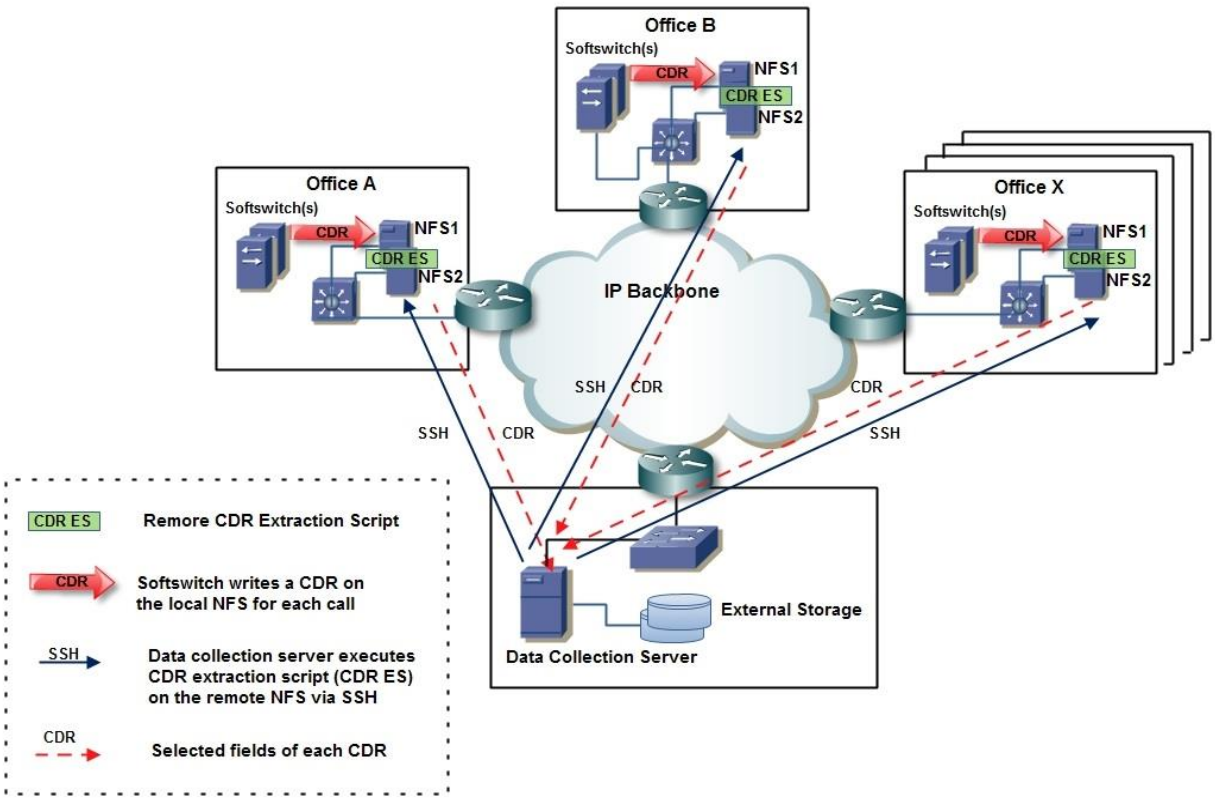

Figure 10. Data collection process

Once we get our CDR copy we divide the traffic into three categories:

- Wireless traffic
- landline traffic
- VoIP traffic.

This categorization is based on the origin of the call. It should be noted that all calls leaving the tandem office to another tandem office are converted into VoIP so they can be transported on the IP backbone. Calls leaving the office to carrier networks (customers) will be converted to VoIP only if the carrier is connected to the office by means of IP circuits. The wireless traffic is usually delivered over TDM links. Figure 11 shows a comparison between the three traffic categories in a typical tandem office. The figure shows that traffic coming to the tandem office from the carrier networks over IP links is only 15% of the overall traffic; however, it is important to notice that all other traffic (wireless and landline) will be converted into VoIP to be transferred to other offices. In addition, all the traffic coming over the backbone from other offices is VoIP. In other words, all the incoming traffic [over the backbone as well as over the carrier links] is either VoIP or "potential" VoIP. The remaining TDM connections to wireless and landline carriers are being converted into IP connections. It is expected that within the next few years all TDM links between the carriers will be replaced by IP connections. After traffic is categorized, we extract traffic information of interest. We keep the raw Time of Arrival (ToA) for each call. We also generate aggregated forms of the call data by dividing the day into time blocks and finding the mean of the call arrivals over each time block. We generate 1, 10, 100, 1200, 3600 seconds aggregated data files. For call holding time study, each data point consists of the call time of arrival (ToA) and the call duration.

We make sure to select different samples of data so that each sample is taken from a different city. Some of these samples are collected from big cities with more than 10 million calls per day, and other samples are collected from small cities with less than 1 million calls per day. This variation in the samples helps finding robust models that can fit wide range of call patterns.

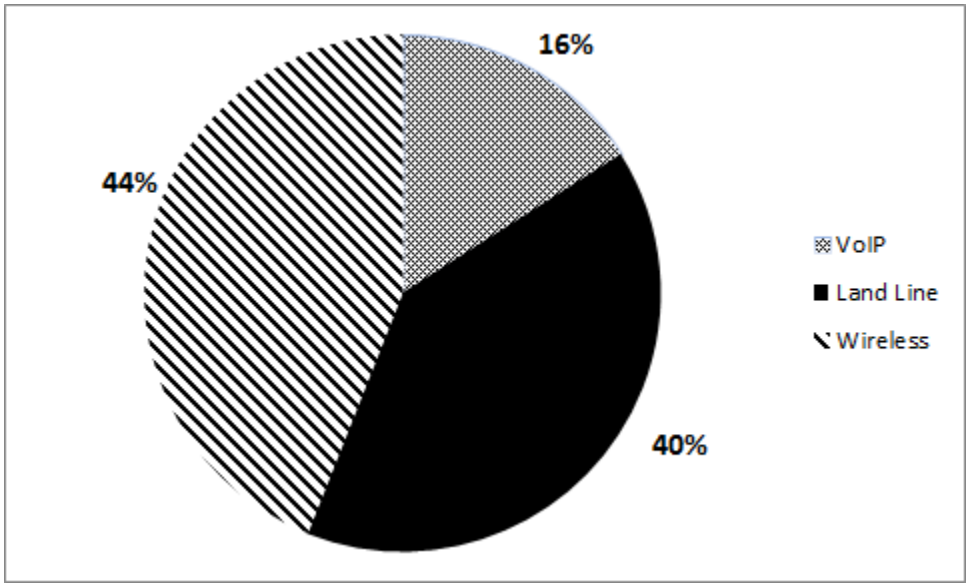

Figure 11. Tandem traffic categories

Our modeling results are identical for all traffic aggregations which indicate the goodness and significance of the proposed models and hence the correctness and robustness of the proposed engineering framework.

4.2 Mathematical and Statistical Modeling and Analysis

We used various mathematical and statistical techniques to fit identify and distributions to empirical data and then to validate the fitted models and estimate the parameters. In this section we briefly describe the major mathematical and statistical methods we used.

Maximum Likelihood estimation (MLE): we used MLE to fit the proposed model function to the actual call arrival data and to estimate the model parameters. MLE is a well-known estimation method that involves a systematic search over different population values. Eventually, MLE selects the estimates that most likely to be true based on the given empirical data sample [45]. MLE is widely used for linear and generalized linear models which we use in our research. The ML estimators are obtained by taking the partial derivatives of the log-likelihood function of the model with respect to each of the model parameters.

Fisher scoring method: It is a mathematical estimation method that is specialized in maximizing the log-likelihood function [46]. We used this method to solve the maximum likelihood equations numerically and hence estimate the parameter values of the generalized linear model fitted to call arrival rate under the non-homogeneous Poisson process.

Wald's significant test: we used this test to test the significance of each parameter in the proposed call arrival function. Wald's test is a well-known hypothesis test and it requires estimation of the unrestricted model (the model without the imposition of null hypothesis restrictions) [47] [48].

Likelihood ratio test: we used this test in modeling call arrival rate as non-homogeneous Poisson process. The test was used to confirm the model and parameter significance results

obtained by Wald's test. It is based on evaluating the difference between the likelihood statistics of two models given that one model is a special case of the other [50]. The Poisson process is a special case of the non-homogeneous Poisson process. Therefore, we used this method to test whether the call arrival rate function is generated by a Poisson process (special case) or by a non-homogeneous Poisson process (general case). The null distribution of the resulting test statistic is a Chi square whose number of degrees of freedom equals the number of model parameters minus one [51].

Survival analysis: Survival analysis techniques are used to analyze time to event problems. They are widely used in medical, biological, engineering, economics, demography, public health, and epidemiological studies [52]. A common feature of the data sets that motivates this approach is that it represents a set of random event durations that can be looked at as time-to-event durations. In our case, the event of interest is call termination and we represent call duration by looking at the time needed for the call to end. Therefore, we used this approach to model call holding time.

Statistical graphical methods: we used graphical methods in the call holding time modeling process. The tools were used based on the empirical estimators and their log transformations. The purpose is to identify the underlying true distribution that fits the empirical data. For example, let $F_n(t)$ be the empirical estimator of the distribution function which is defined as $F_n(t) = n^{-1} \sum_{i=1}^{n} 1(t_i \leq t)$, where $t_1, \ldots, t_n$ are the observed call durations and 1(A)=1 when A is true and 0 otherwise. Then, the plots of $\log(1-F_n(t))$ and $\log(-\log(1-F_n(t)))$ versus t should both yield straight lines when the call duration have an exponential or an extreme value distribution. While the plots of $\log(-\log(1-F_n(t)))$, $\Phi^{-1}(F_n(t))$ and $\log(F_n(t)/(1-F_n(t)))$ versus log(t) would indicate, respectively a Weibull, Log Normal, and Log-logistic random variables, when the

curves are linear. We used SAS software in order to plot the functions and their transformation mentioned above.

Gauss-Newton numerical method: Gauss-Newton optimization method is a non-linear least square modification of Newton method. It is used to minimize the sum of squared values of the function [55]. It is a fast method and it is recommended whenever the problem can be expressed as a non-linear least square format [54]. We used this numerical method to compute the non-linear maximum likelihood estimator for the call cease rate function fitted to call holding times.

Cox-Snell residuals analysis: It is an efficient technique used to compute the departure of the data from the proposed model [58]. We used this technique as a goodness of fit assessment method in order to detect the deviation of the empirical data from the proposed call holding time model. If empirical data has been fit to the correct model, then Cox-Snell residuals will have a unit exponential distribution with a hazard ratio of one. Cox-Snell residuals are given as [57]:

$$e_i = \log\left(1 - \hat{F}(t_i)\right)$$

where is $\hat{F}(t)$is the estimated probability distribution function based on the fitted model

If the Cox-Snell residuals follow a unit exponential distribution with a hazard ratio of one, then the plot of estimate of the integrated hazard rate of Cox-Snell residuals against Cox-Snell residuals themselves is a straight line with a slope of one [56].

Akaike Information Criteria (AIC) and Bayes Information Criteria (BIC) test: AIC is a variant of the likelihood ratio test and is used to measure model fitting accuracy [59]. BIC is useful for model selection by comparing different models. The likelihood can be increased by adding more parameters to the model. BIC adds a penalty term with each model parameter in

order to prevent over-fitting by introducing many parameters [60]. We used AIC and BIC tests in order to compare between a set of candidate models for Call Cease Rate function and then select the best model that fits the sample data.

Least-Squares (LS) method: we used least-square model to estimate the parameters of the model fitted to call arrival rate under the Gaussian approximation condition. LS method is basically about minimizing the sum of the squares of errors between the sample data and the fitted model or estimated parameter [62]. The major advantage of LS estimation method is that it does not require knowledge of the underlying distribution of the error component (t). When the model is linear, it delivers explicit expressions for the parameters' estimates. In addition when ϵ(t) is assumed to have a Gaussian distribution, it is possible to make inferences about the parameters' significance and about the model's validity and usefulness. Therefore, we can use the model to make predictions about future observations [61].

Normality tests: we used normality tests to prove that call arrival rate can be approximated as a normal (Gaussian) distribution under heavy traffic condition. The assumption of normality is a statistical procedure that requires some robust testing in order to confirm whether or not the assumption holds [63]. In order to verify the validation of our normality assumption, we used three different tests: Anderson-Darling test, (ii) Kolmogorov-Smirnov, and (iii) Shapiro-Wilks

R-Language: we used R-Language for the majority of data fitting, modeling and validation throughout this research. R is a language for statistical data analysis and graphics. It is open source and runs on Linux, windows and Macintosh. It has good graphical capabilities and excellent online-help support. The R-Language has powerful syntax with many built-in functions. It also supports used defined function for further flexibility and extendibility [64].

4.3 Simulation

The call holding time and call arrival rate models proposed in this study are non-Markovian and hence providing an analytical performance study for the traffic engineering model is not feasible. Therefore, we built a simulation-based system in order to study the performance of the VoIP networks based on the proposed models (VSIM).

VSIM simulation consists of a parametric G/G/c/c simulation based on the NHPP model, and a non-parametric simulation based on the captured call information. According to Kendall's notation the G/G/c/c is a queuing system where calls are assumed to arrive according to a general distribution (G) and have a service time that follows another general distribution (G), the system has a limited number of servers/channels (c) and no waiting queue (maximum number of calls in the system equals the number of servers c). Therefore, VSIM simulator engine can be used to simulate and complex queuing system.

VSIM simulation is built using Java programming language and based on the CSIM for Java API [105]. CSIM API is an advanced simulation kit for building large-scale and complex simulation models. It provides a library of routines for building process-oriented discrete-event simulations. Below are the major highlights of the CSIM program structure.

Process: CSIM API models a customer or call entering a queue as a process that starts by creating an active entity. CSIM processes run under the control of CSIM execution supervisor which coordinates the execution and timing of the processes. CSIM processes can be in one of the following states:

- Computing: actively computing using the host machine's CPU

- Ready: ready to enter the computing state
- Holding: allowing simulated time to pass
- Waiting: waiting for an event to happen or facility to become available.

A CSIM process can suspend its execution (leave the Computing state) and then resume execution later (enters the Ready state and then reenter the Computing state) for unlimited number of times and in no predictable pattern; the CSIM execution supervisor manages all of these activities. In addition, there can be several simultaneously active instances of the same process (entities). Each of these instances appears to be executing in parallel to each other (in simulated time) even though they are in fact executing sequentially on a single processor on the host machine. The CSIM runtime system guarantees that each instance of every process (entity) has its own runtime environment.

Inter-process communication: CSIM library provides two structures to enable and control communications and interactions between different processes. These structures are:

- events
- mailboxes

A process can wait for a certain event to occur while another process can set an event; causing it to be placed in the OCCURRED state and allowing all of the waiting processes to resume (enter the Ready state). A mailbox is a place where processes can exchange messages. One process can send a message to a mailbox. Another process can attempt to receive a message from the mailbox; if the message is already in the mailbox, the receiving process gets that message and continues computing. However, if there are no messages in the mailbox, all receiving process must wait until a message is sent to that mailbox.

Resources: In queuing simulation, an object/process needs to occupy some simulated system resources for a certain period of time. CSIM offers two kinds of resources: facilities and storages. VSIM was built using facilities to model the system/network capacity.

Facility: a facility is usually used to model system resources. A simple facility consists of a single server and a single queue (for processes waiting to gain access to the single server). Only one process at a time can be using the server. A multi-server server facility contains a single waiting queue and multiple servers. All of the waiting processes are placed in the queue until one of the servers becomes available. Facilities are used to represent simulated system resources where entities (processes) occupy servers in a one-at-a-time fashion. A process can apply one of the following operations on a facility:

- reserve : wait for and then gain access (occupy) to a "free" server
- hold: occupy the facility/server for a certain period of time
- release: release a reserved server
- use: a combination of a reserve, hold, and release operations
- reset: reset statistics and counters associated with a given facility

In addition to the queue and server(s), a facility also has provisions for collecting performance data on the delays associated with gaining access to a server (queue waiting) and on using the servers (hold). This data collection can be provided by CSIM automatically for each facility; a report summarizing the collected performance data can be produced at any time during the execution of the model.

Storage: a storage consists of a queue and a pool of storage units (sometimes called tokens). A process can allocate one or more storage units; if there is not sufficient number of tokens that can satisfy an allocation request, the process is suspended and placed in the waiting queue. When other processes have deallocated their storage tokens, queued processes are given units to satisfy their resource requests. As with facilities, performance data is collected to summarize the queuing delays and holding times for these storage units. The main difference between a storage and facility is that the storage is divided into smaller tokens; therefore it can be partially allocated to a requesting process. Storage resource is not suitable for our research since an incoming call requires a free IP trunk or channel and that is not divisible.

CHAPTER 5

5 Traffic Engineering Modeling Results and Analysis

In this chapter we review the work we have done during this research. Also we highlight the major results, findings, and contributions.

5.1 Modeling Call Holding Time for VoIP Tandem Networks

We present a new approach for modeling call holding time on VoIP tandem networks. This research is based on millions of call information obtained from a tandem network. The collected data is a mixture of residential, wireless as well as business call data. The tandem network topology as well as the research methodology is described in Section 4.1.

Call holding time is a key variable of traffic engineering models. The traditional Erlang-B model uses a negative exponential function to model call holding time. Our study of large

number of telephone calls shows that the exponential assumption is not valid for modern large-scale VoIP networks. We propose to use time-to-event analysis which consists of fitting a parametric model to the call cease rate. Then we study several probability distribution functions and compare their capability to model the VoIP call departure rate. We find that both the log-logistic and the generalized gamma distributions provide a good fit for the data. Our statistical analysis shows that the approach of modeling call cease rate provides more accurate results than the traditional exponential and log-normal holding time models.

5.1.1 Data Exploration

We base our analysis on wide variety of samples. Some samples are collected from big cities (8 to 10 million calls per day) and other samples are collected from smaller cities (0.5 to 1 million calls per day). This variation in the samples helps finding a robust model that fits wide range of call patterns. Each collected data point consists of the call Time of Arrival (ToA) and the call duration. Because of its nature, this type of data can only be modeled using a positive random variable.

The collected data yields a histogram with a very heavy tail. The call service times (in seconds) occur within an extreme range (0.6 – 169,6245) for a small city sample and (0.3 – 235,000) for the big city sample. The mean of each set is much larger than median and the skewness coefficient is 75 and kurtosis is 16,509 (compared to 2 and 9 for the exponential function). It is clear in Figure 12 that many observations occur way beyond the range of values assumed by the exponential and this makes the exponential distribution far from capturing this traffic pattern.

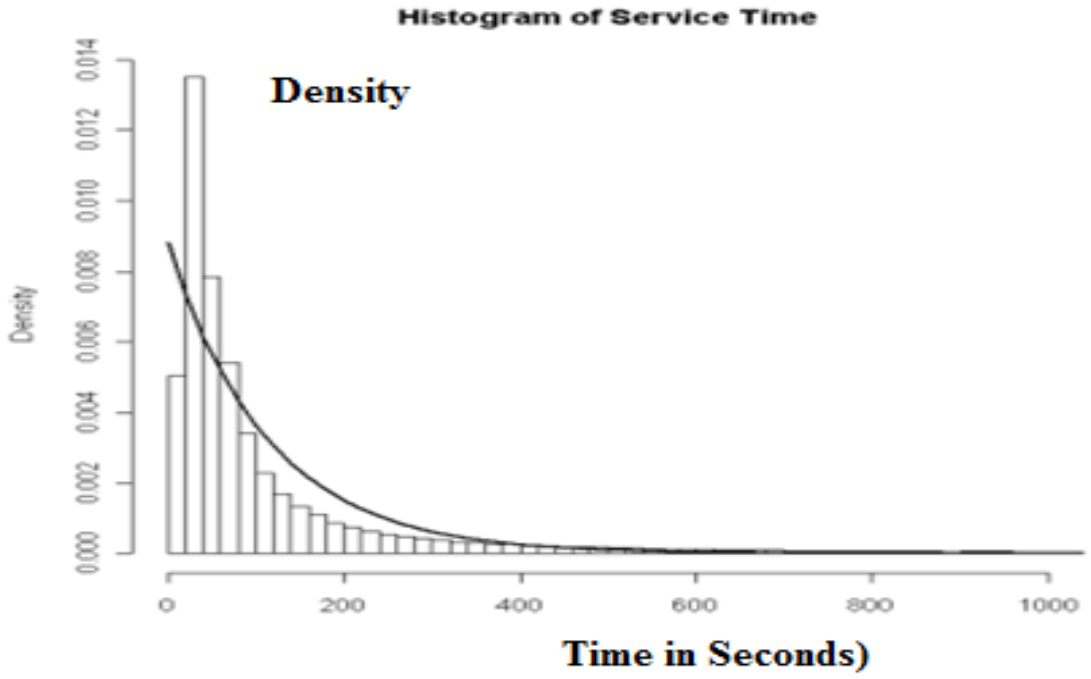

Figure 12. Exponential distribution against truncated data

In cases like this, mixtures of lognormal distributions have been proposed to fit the data whose distributions have tails that are higher than those of the exponential. For this purpose we look at the histogram of the logarithms of the service times and try to fit a mixture of lognormals to the data. Once again, as shown in Figure 13, the histogram's relative frequencies of the tails are much larger than the tail values of the proposed density. Therefore, a mixture of lognormal fails to capture the heavy-tailed data and hence it is not an appropriate model.

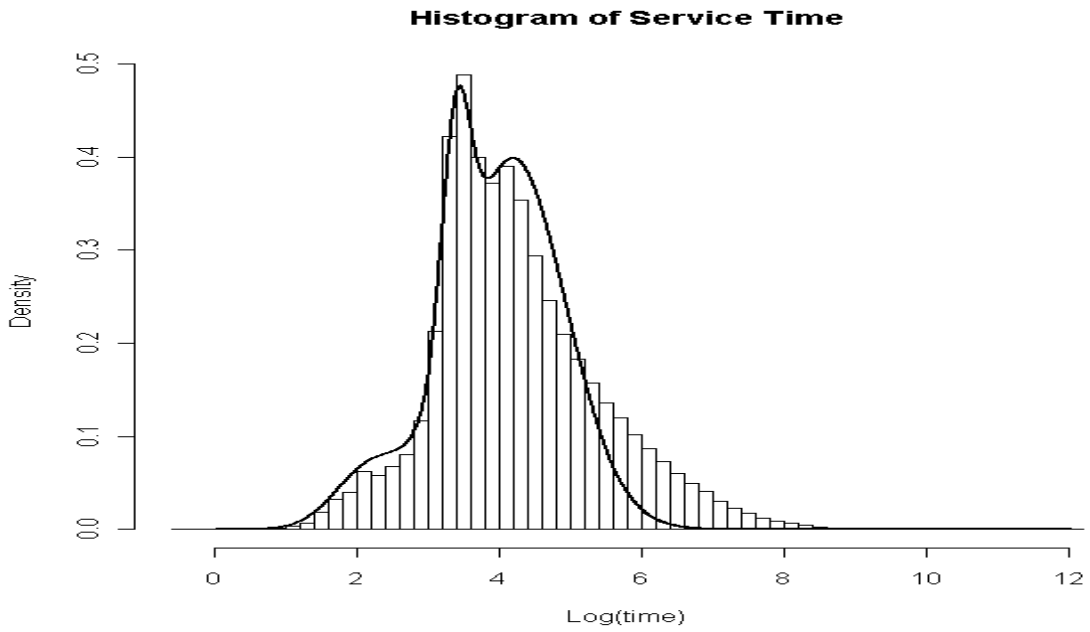

Figure 13. Fitting call duration data to a Mixture of Lognormals

5.1.2 Introducing the Call Cease Rate Function

Based on the conclusions of the previous data exploration, it seems that there is a need to use a distribution with heavier tail than the exponential or lognormal to fit the data. In general, modeling a set of random event durations differs significantly from the classical methodology used when the data is generated by location-scale distributions such as the Gaussian. In the classical approach, the functions of interest to the analyst are the probability distribution and density functions. The data at hand falls in the first category. Therefore, we opted to introduce a function that provides a better interpretation of time-related phenomena such as the one under

investigation. This function represents the instantaneous probability that the call will end at time *t* and is defined as:

$$\lambda(t) = \lim_{\Delta t \to 0} \frac{Pr(a\ call\ that\ lasts\ at\ least\ \ t\ will\ finish\ before\ t + \Delta t)}{\Delta t} \qquad (1)$$

$\lambda(t)$ is called the Hazard Function by survival analysts, Failure Rate function by reliability engineers or Force of Mortality by demographists. Since we are observing the call cease events which does not have a connotation of risk or failure, we chose to call our $\lambda(t)$ as the Call Cease Rate function.

Since time is continuous, the probability that the call will end at exactly time *t* is 0. Hence, we introduce the concept that the call duration is between *t* and *(t+Δt)* and we make this probability implicitly conditional on the call lasting to time *t*. In light of this, the above Call Cease Rate function can be written as:

$$\lambda(t) = \lim_{\Delta t \to 0} \frac{\Pr(t \le T <\ t + \Delta t | T \ge t)}{\Delta t} \qquad (2)$$

where *T* is the call duration.

The relationship between the conditional, the joint, and the marginal distributions of *T* leads to:

$$\lambda(t) = \lim_{\Delta t \to 0} \frac{1}{\Delta t} \frac{\Pr(t \le T <\ t + \Delta t)}{\Pr(T \ge t)} \qquad (3)$$

$$= \lim_{\Delta t \to 0} \frac{1}{1 - \mathrm{F}(t)} \frac{F(t + \Delta t) - F(t)}{\Delta t} = \frac{f(t)}{1 - F(t)} \qquad (4)$$

where *F(t)* and *f(t)* are respectively the probability distribution and density functions.

The relationship between *f(t)* and *F(t)* implies that $\lambda(t)$ can be written as:

$$\lambda(t) = -\frac{\frac{d}{dt}(1-F(t))}{1-F(t)} = -\frac{d}{dt}\log\big(1-F(t)\big) \tag{5}$$

solving the above equation leads to:

$$1 - F(t) = \exp\left(-\int_0^t \lambda(s)ds\right) \tag{6}$$

$$f(t) = \lambda(t)\exp\left(-\int_0^t \lambda(s)ds\right) \tag{7}$$

From 7, we notice that density functions defined through their $\lambda(t)$ form a generalization of the exponential distribution in the sense that, at a given moment t_o, the call duration has an instantaneous exponential distribution with rate $\lambda(t_o)$. There is a one-to-one relationship between $\lambda(t)$ and *F(t)*, so defining a family of distributions can be done through the call cease rate function $\lambda(t)$. Table 7 shows some distributions along with their hazard and density functions.

Table 7. Hazard and Density functions

Distribution	$\lambda(t)$	**F(t)**
Exponential	λ	1-exp(-λt)
Weibull	$\lambda\gamma t^{\gamma-1}$	1-exp($-\lambda t^{\gamma}$)
Extreme Value	$(1/\sigma)\exp((t-\mu)/\sigma)$	1-exp(-exp($(t-\mu)/\sigma$))
Log Normal	$\frac{1}{\sqrt{2\pi}\,\sigma t}\exp\left(-\frac{1}{2}\left($ $1-\Phi(\frac{\log(t$	$F(t) = \Phi\left(\frac{\log(t)-\mu}{\sigma}\right)$ Φ is the standard
Log-logistic	$\frac{\beta}{\alpha}\left(\frac{t}{\alpha}\right)^{\beta-1}\left(1+\left(\frac{t}{\alpha}\right)^{\beta}\right)^{-1}$	$F(t) = \left(1+\left(\frac{t}{\alpha}\right)^{\beta}\right)^{-1}$

Notice that the exponential distribution is a special case of the Weibull distribution (with $\gamma = 1$). The Weibull and log-normal distributions on the other hand are special cases of a larger family of distributions called the generalized gamma, which contains also the classical Gamma distribution.

We can use the collected data to obtain empirical estimate of the call cease rate function $\lambda(t)$ and of the distribution *F(t)*. We used graphical tools based on these empirical estimates and their log transforms in order to identify the underlying true distribution of the data. With this in mind we consider $F_n(t)$, the empirical estimator of the distribution function defined as $F_n(t) = n^{-1}\sum_{i=1}^{n} 1(t_i \leq t)$, where $t_1, \ldots, t_n$ are the observed call durations and 1(A)=1 when A is true and 0 otherwise. Then, the plots of $\log(1-F_n(t))$ and $\log(-\log(1-F_n(t)))$ versus t should yield

respectively, straight lines when the call duration have an exponential or an extreme value distribution (Figure 14). While the plots of $\log(-\log(1-F_n(t)))$, $\Phi^{-1}(F_n(t))$ and $\log(F_n(t)/(1-F_n(t)))$ versus log(t) would indicate, respectively a Weibull (Figure 15), Log Normal (Figure 16) and Log-logistic (Figure 17) random variables, when the curves are linear. Notice that the identification step has to be completely data-based and hence no assumptions are made throughout it.

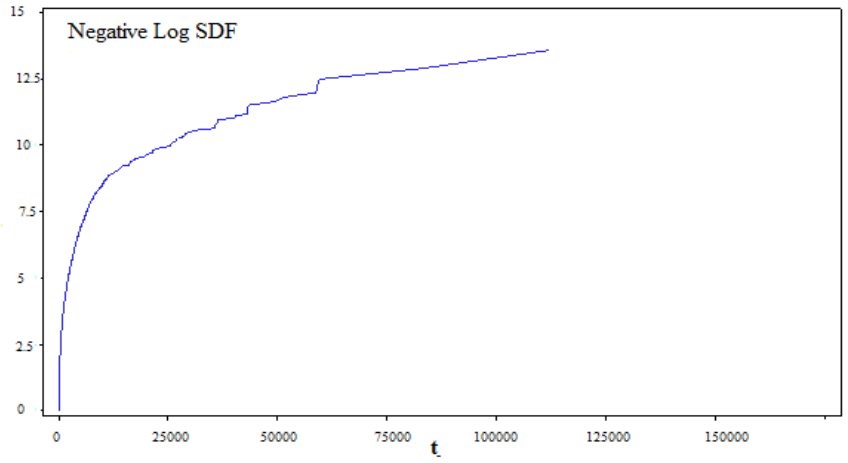

Figure 14. Exponential and extreme value distribution test

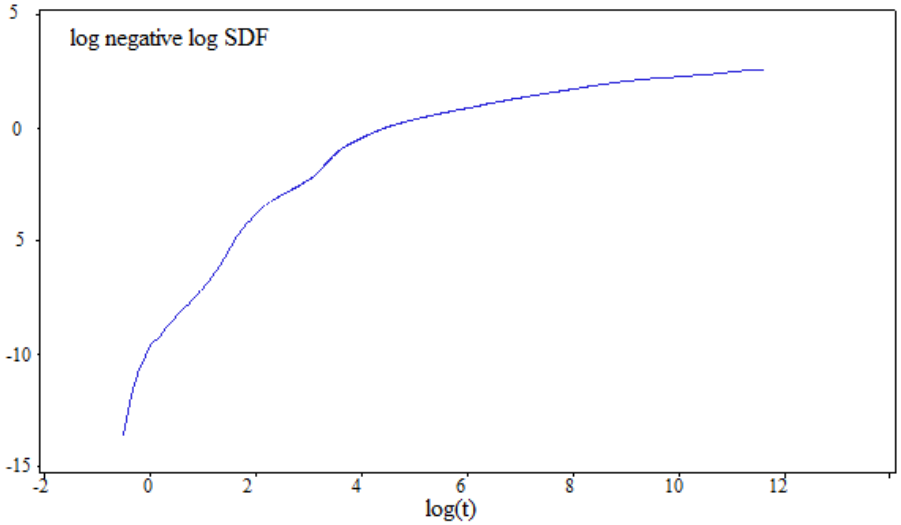

Figure 15. Weibull distribution test

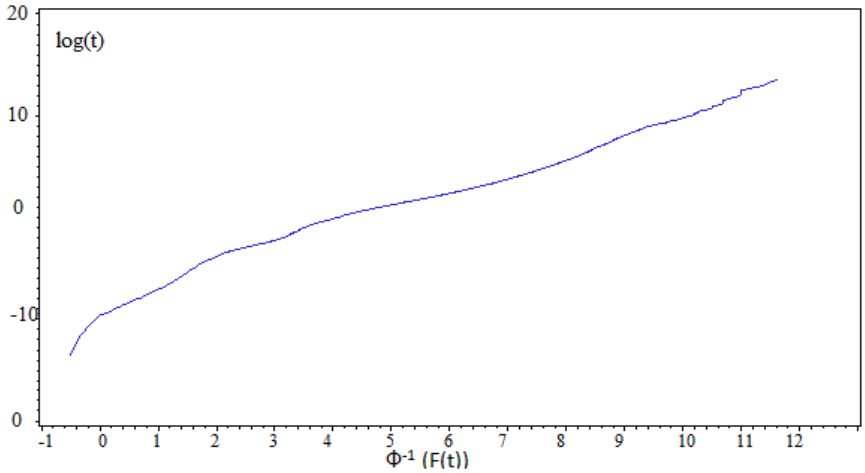

Figure 16. Log Normal distribution test

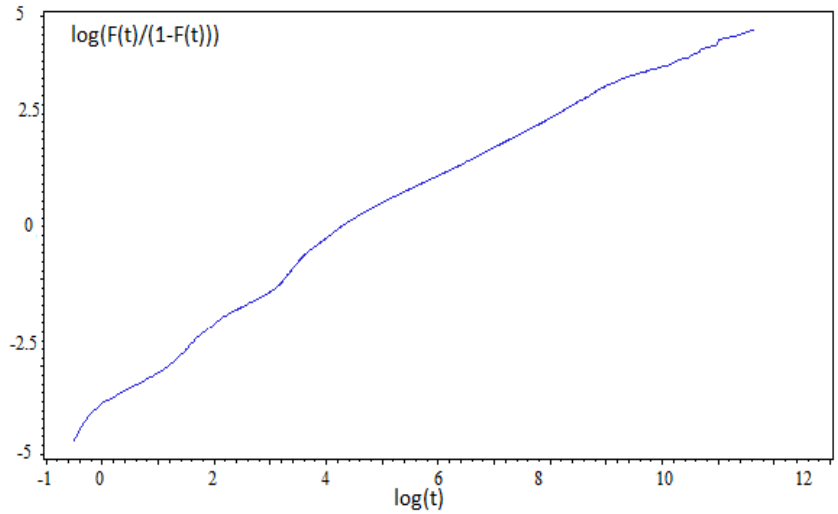

Figure 17. Log-logistic distribution test

Figure 14 and Figure 15 deviate significantly from a straight line; therefore, we can be sure that our call holding data does not follow an exponential or a Weibull distribution. The curves in Figure 16 and Figure 17 seem to be quite close to a straight line, which suggest that the data might follow either a log-normal or a log-logistic distribution. Next, we present the estimation process of the parameters for the call cease rate model.

5.1.3 Model Estimation

We use maximum likelihood estimation (MLE) since it yields parameter estimators that are consistent and whose sampling distributions are known. The sampling distributions are useful in testing the significance of the parameters and hence provide robust model validation. Other advantages of such method of estimation are the information criterion which can be used for model comparison and the Cox-Snell residuals which can be used to check the model's validity.

The likelihood is a function of a model's parameters and is defined to be equal to the joint density of all observations. Since the calls durations do not depend on each other, we can write the likelihood function as:

$$L(\theta) = \prod_{i=1}^{n} \lambda(t_i, \theta) \exp\left[-\int_0^{t_i} \lambda(s,\theta) ds\right] \quad (8)$$

Where $\boldsymbol{\theta}$ is a vector of parameters.

Since the logarithm is a convex increasing function, maximizing the likelihood is equivalent to maximizing the log-likelihood, which can be written as:

$$l(\theta) = \sum_{i=1}^{n} \log\big(\lambda(t_i,\theta)\big) - \sum_{i=1}^{n} \int_0^{t_i} \lambda(s,\theta) ds \quad (9)$$

Now we will apply MLE to estimate parameters for each of Log-logistic, Log-Normal, and Generalized gamma. We will compare the result and decide on the model that best first our Call Cease Rate function.

A. Log-logistic distribution

Since the call cease rate function of a log-logistic is:

$$\lambda(t,\alpha,\beta) = \beta t^{\beta-1}(\alpha^{\beta} + t^{\beta})^{-1} \qquad ,\alpha,\beta > 0 \quad (10)$$

Then its primitive function is:

$$\int_0^t \lambda(s,\alpha,\beta)ds = \log\left(\alpha^\beta + t^\beta\right) - \log\left(\alpha^\beta\right) \tag{11}$$

Therefore, the log-likelihood becomes:

$$l(\alpha,\beta) = n\log(\beta) + n\beta\log(\alpha) + (\beta-1)\sum_{i=1}^{n}\log(t_i) - 2\sum_{i=1}^{n}\log\left(\alpha^\beta + t_i^\beta\right) \tag{12}$$

Taking the derivatives with respect to the parameters leads to the following score functions:

$$\frac{\partial(\alpha,\beta)}{\partial\alpha} = \frac{n\beta}{\alpha} - 2\beta\alpha^{\beta-1}\sum_{i=1}^{n}\frac{1}{\alpha^\beta + t_i^\beta} \tag{13}$$

$$\frac{\partial l(\alpha,\beta)}{\partial\beta} = \frac{n}{\beta} + n\log(\alpha) + \sum_{i=1}^{n}\log(t_i) - 2\sum_{i=1}^{n}\frac{\alpha^\beta\log(\alpha) + t_i^\beta\log(t_i)}{\alpha^\beta + t_i^\beta}$$

At the maximum, the score functions are equal to zero and the maximum likelihood estimators are obtained by solving the equations in (13). The nonlinearity of the equations necessitates the use of numerical optimization methods such as Gauss-Newton to find the MLE of the

parameters. Shown below are the parameters estimates along with their standard errors and confidence intervals:

Parameter:	**α**	**β**
Estimate	63.257	1.546
Std. Error	0.0791	0.0014
95% Conf. limits	63.257±0.155	1.546±0.003

B. Log-Normal distribution

Estimating the parameters of the log normal distribution is much easier, and estimates are obtained by solving a system of linear equations. The likelihood function based on the log-transformed of the call durations is then:

$$L(\mu,\sigma)=\prod_{i=1}^{n}\exp\left(-\frac{1}{2}\left[\frac{\log(t_i)-\mu}{\sigma}\right]^2\right) \tag{14}$$

This means that the log-likelihood function is:

$$l(\mu,\sigma)=\text{constant}-n\log(\sigma)-\frac{1}{2}\sum_{i=1}^{n}\left(\frac{\log(t_i)-\mu}{\sigma}\right)^2 \tag{15}$$

Taking the derivatives with respect to the parameters leads to the following score functions:

$$\begin{aligned}\frac{\partial l(\mu,\sigma)}{\partial\mu}&=\frac{1}{\sigma^2}\sum_{i=1}^{n}\left(\log(t_i)-\mu\right)\\ \frac{\partial l(\mu,\sigma)}{\partial\sigma}&=-\frac{n}{\sigma}+\frac{1}{\sigma^3}\sum_{i=1}^{n}\left(\log(t_i)-\mu\right)^2\end{aligned} \tag{16}$$

When the score functions are equal to zero, we obtain the explicit form of the MLE's:

Parameter:	**μ**	**σ**
Estimate	4.222	1.171
Standard Error	0.0013	0.0009
95% Conf. limits	4.222±0.003	1.171±0.002

C. Generalized gamma distribution

We also fit a generalized gamma distribution to the data. This model has 3 parameters which makes it the most flexible model for a positive random variable like the call duration. As discussed previously, several models are special cases of the generalized gamma. Due to the lack of page space, we skip showing the complex likelihood related calculations. We obtain a table with the parameters estimates, their standard errors and confidence intervals:

Parameter:	**Intercept**	**Scale**	**shape**
Estimate	3.948	1.100	-0.480
Standard Error	0.002	0.0009	0.0024
95% conf. limits	3.95±0.004	1.1±0.002	-0.48±0.005

Notice that the shape and scale parameters are not equal and hence the model is not a standard gamma. Also, the exponential distribution is a special case of the generalized gamma (scale and shape parameters both equal to 1).

5.1.4 Goodness of Fit and Model Validation

The Cox-Snell residuals are considered to be the most efficient at detecting the departure of the data from the proposed models (Allison [57]), and Collett [58]). They are defined as:

$$e_i = \log\left(1 - \hat{F}\left(t_i\right)\right) \tag{17}$$

Where is $\hat{F}(t)$is the estimated probability distribution function based on the fitted model. Unlike the usual residual from a classical linear model, the Cox-Snell residuals are always positive and when the fitted model is correct, they have an approximately exponential distribution with rate equals to 1. Therefore, we can use the plot -log(1-F_n(t)) vs. t described in section 5.1.2 to evaluate the exponentiality of e_i's. Since the rate is 1, the plot should look like a straight line with intercept 0 and slope 1. The graph shown in Figure 18 below shows Cox-Snell residuals plots for each of the models fitted in section 5.1.3; log-logistic, log-normal and generalized gamma.

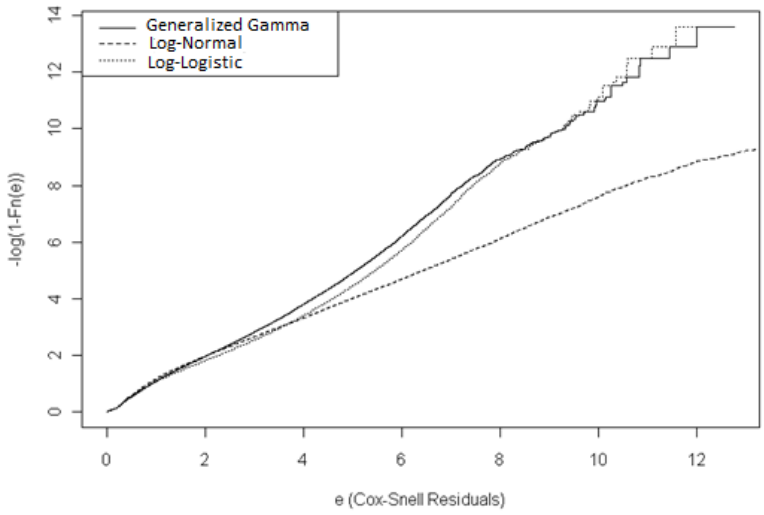

Figure 18. Cox-Snell residuals for Log-logistic, Log-normal, and Generalized gamma

All the graphs seem to display a linear curve running through the origin; however, the log normal graph has a slope that is significantly different from 1. Both log-logistic and generalized gamma provide a slope close to 1 and hence are both valid models for this data, with slight advantage for the generalized gamma. This confirms our early conjecture that the generalized gamma would be a better fit since it is the most flexible.

To compare the 3 different models, the most commonly used criteria (Lindgren,Berger) are the Akaike Information Criteria (AIC) and Bayes Information Criteria (BIC) [38]:

$$\mathrm{AIC} = -2\mathrm{l}(\hat{\theta}) + 2\mathrm{p} \qquad (18)$$

$$\mathrm{BIC} = -2\mathrm{l}(\hat{\theta}) + \mathrm{p}\log(\mathrm{n}) \qquad (19)$$

Where $l(\hat{\theta})$ is the log-likelihood of the estimated model, *p* is the number of parameters in the model and *n* is the number of observations in the model. A smaller information criterion indicates a better model.

Below is a table of the criteria for each of the models considered in section 5.1.3

Distribution	**AIC**	**BIC**
Log-normal	2558481	2558504
Log-logistic	2541143	2541167
Gen. gamma	2519514	2519549

Both criteria agree that the generalized gamma model is noticeably better than its competitors with the log-logistic distribution behaving better than the log-normal. Besides information criteria, the choice between generalized gamma and Log-logistic can be affected by other factors such as model flexibility, which would favor the generalized gamma, or model parsimony, which

would favor the log-logistic distribution. Equations 20 and 21 show the call cease rate functions corresponding to the two generalized gamma and Log-Logistic models respectively. Figure 19 shows their plots truncated at 1000 seconds for clarity.

$$\lambda_{gg}(t) = \frac{3651623\, t^{-2.89} \exp\left(-24.33\, t^{-0.44}\right)}{1-\Gamma\left(4.34, 24.33\, t^{-0.44}\right)} \qquad (20)$$

where Γ is the upper incomplete gamma function.

$$\lambda_{ll}(t) = 1.55\, t^{0.55}\left(619.09 + t^{1.55}\right)^{-1} \qquad (21)$$

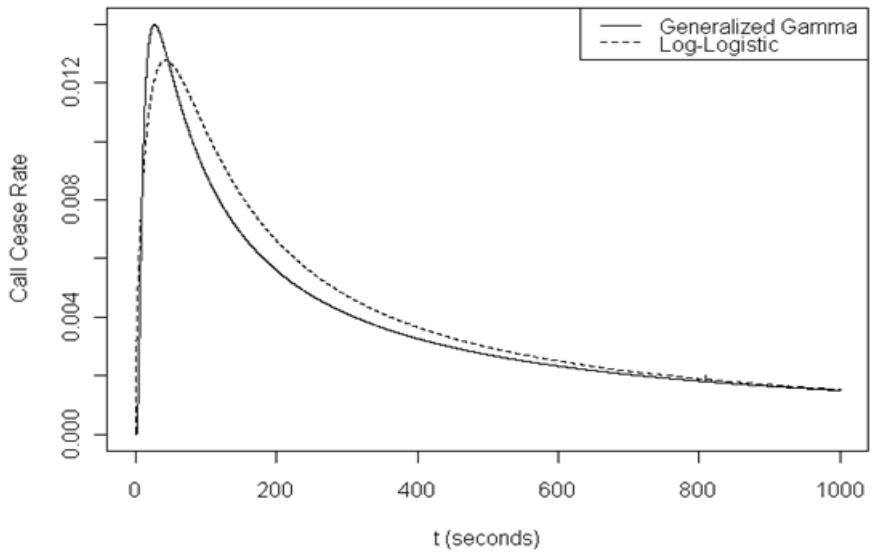

Figure 19. Call Cease Rate function (t <1000)

Notice that both rate functions decay at an algebraic rate. Moreover, we can show through series expansions of the two rate functions, that their tails are assymptotically proportional to t^{-1}. This implies that the tail of the density function is asymptotically proportional to $t^{-(1+a)}$ which

explains the large number of extremely long call durations that cannot be fit with a density that decays exponentially.

5.1.5 Final Model

Based on the model validation and comparison presented in sections 5.1.3 and 5.1.4, we decide that the generalized gamma distribution is the best model. Since our data contains a large number of extreme observations, we opted to zoom the call cease rate function plotted in Figure 20 to the range between 0 and 300 seconds. It is noteworthy that the call durations are not exponentially distributed since $\lambda(t)$ is not a constant. Also notice that short calls of duration less than 28 seconds have an increasing call cease rate which means that at a given time $t<28$ such calls are more likely to end than continue. On the other hand, calls of duration longer than 28 seconds have a decreasing call cease rate which means that at any time $t>28$, such calls are unlikely to end in the next few seconds and will tend to continue for some time. This is an insightful result especially that the durations of more than 20% of the calls are less than 30 seconds. The significant number of short calls is a direct result to the small-business credit card transaction and processing systems as well as the automated voice applications such as voice mail and Interactive Voice Response (IVR) systems where many callers tend to leave very short messages or hang up. These systems and behavior result in generating large number of calls with call duration that does not exceed a few seconds.

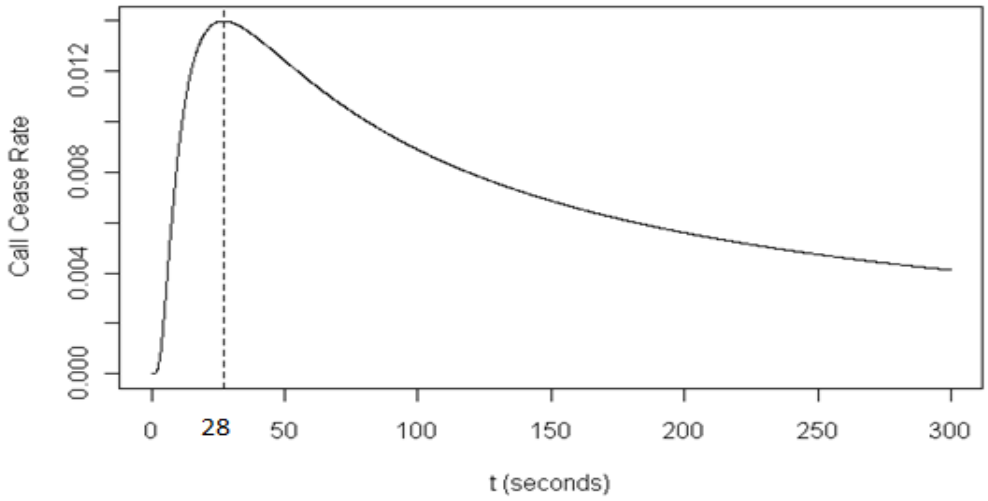

Figure 20. Call Cease Rate function (t <300)

5.1.6 Summary

In this study we used a time-to-event analysis to model call holding time in modern large-scale VoIP networks. This approach consists of estimating the parameters of a hazard rate function which corresponds in our case to the Call Cease Rate function. This methodology is effective in studying phenomena described by random time variables such as call durations.

We were able to obtain mathematical models that can accurately capture important characteristic features of modern telecommunication systems, mainly the skewness and heavy-tailedness of the call duration distribution. We used maximum likelihood estimation for model fitting, Cox-Snell residuals plots for model validation, and Akaike and Bayes information criteria for model comparison. We conclude that the log-logistic and generalized gamma distributions provide good fits for the data with a slight advantage for the generalized gamma.

5.2 Modeling Call Arrival Rate as NHPP

Erlang-B model is traditionally used to estimate the telecom network resource requirements. This model is based on the Poisson arrival distribution where the rate (λ) is constant and is measured based on the Busy Season Busy Hour (BSBH). BSBH is the busiest hour in the busiest week during the year. Networks are designed to handle traffic offered during this hour. Using a constant call arrival rate fails to adapt to the variation of traffic with respect to time such as time of day, day of week, and day of year.

Under the BSBH approach, a considerable portion of network resources will remain idle for the majority of the year which results in poor resource utilization. Such problems can be justified in the PSTN world because of the difficulties associated with allocating and revoking network resources. For example, the typical limiting resource of a PSTN network is the number of trunks connecting central offices. Increasing or decreasing this number is a complicated and expensive process that involves the interaction of multiple parties. In the IP world resource allocation is more flexible. Allocating more or less bandwidth for voice applications is a relatively simple process. Dynamic resource allocation for VoIP traffic can be useful especially for converged networks where voice and data share the same physical facilities. More bandwidth can be allocated to voice traffic during busy days while providing non-used bandwidth for data applications during the remainder of the year.

In this research we propose a new approach to traffic engineering by applying a Non-Homogeneous Poisson Process (NHPP) for call arrival rate. Then we apply a generalized linear function to model call arrivals as a function of time. The proposed model supports dynamic allocation of network bandwidth based on predicted traffic. Modern network management

systems can easily support this dynamic bandwidth allocation procedure. Furthermore, a dynamic resource allocation system can adopt a de-allocation scheme which can significantly minimize call blocking probability and maximize the bandwidth utilization.

5.2.1 Call Arrival Patterns

The models developed in this research are based call information collected from an IP tandem network as described in Section 4.1. During the initial data exploration we noticed that the minimum call load occurs near 4 AM of each day. Based on this finding we redefine the day from a traffic engineering perspective as the period between 4 AM and 4 AM of the next day. Furthermore, we notice that different days have different patterns. For example the difference between the call load on Fridays and that on Sundays is noticeable and should not be ignored. Figure 21 shows call arrival patterns for a typical week. We notice call arrival difference of 70% between Friday and Sunday. Our proposed model takes the daily effect into consideration.

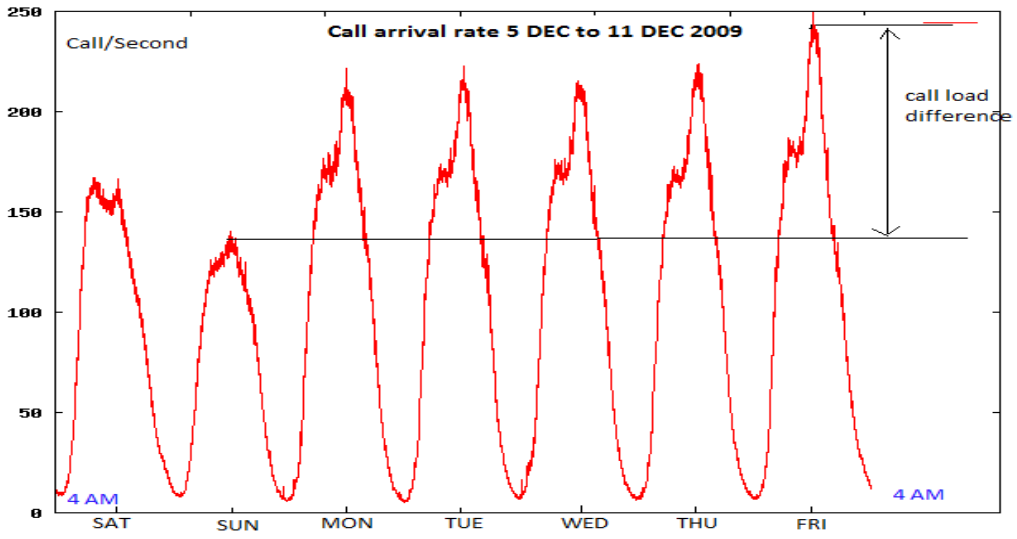

Figure 21. Call arrival pattern for a typical week

5.2.2 Model Formulation and Validation

Given our data, we are inspired to construct a model that describes the variation of call arrival rates during a week. It is common in statistical analysis to model the logarithm of $\lambda(t)$ instead of $\lambda(t)$ itself for count data [39]. Such transformation would guarantee that the estimate of the intensity function is always non-negative. Our model takes into consideration the daily arrival patterns and has the time-dependent intensity function of:

$$log[\lambda(t)] = \mu + \sum_{i=1}^{k_o} [\alpha_i \sin(i\omega_o t) + \beta_i \cos(i\omega_o t)] + \sum_{j=1}^{6} \gamma_j I_j(t) \quad (1)$$

where: $\lambda(t)$ is a function of time (t).

$I_j(t)$ is day Indicator function where j is the day of the week. The value of $I_j(t)$ is 1 if the time $t \in j$ and 0 otherwise. K_o is the number of harmonics in the model. μ represents the model central tendency without daily effects. γ_j is the effect of day j and represents the difference between μ and the mean number of calls for day j. α_i and β_i are the contribution of the *ith* harmonic to the model.

We use Maximum likelihood estimation to fit our proposed $\lambda(t)$ to the actual call arrivals. As explained in section 4.1.2, the processed call arrival data is aggregated into non-overlapping time intervals (δ) of 10, 100, 1200, and 3600 seconds. Thus we will use the total number of calls within time intervals rather than the exact call time of arrival.

Let $n_1, n_2, .., n_{m-1}, n_m$ denote the number of calls arrived at the system in non-overlapping intervals $(a_1, a_2], (a_2, a_3], \dots, (a_{m-2}, a_{m-1}]$, $(a_{m-1}, a_m]$. Therefore, the likelihood function *L* is given as:

$$L = exp\left\{-\sum_{i=1}^{m}\int_{a_i}^{a_{i+1}}\lambda(t)dt\right\}\prod_{i=1}^{m}\frac{\left(\int_{a_i}^{a_{i+1}}\lambda(t)dt\right)^{n_i}}{n_i!} \quad (2)$$

and the log-likelihood, apart from a given constant, is given as:

$$l = \sum_{i=1}^{m} n_i log\int_{a_i}^{a_{i+1}}\lambda(t)dt \quad -\sum_{i=1}^{m}\int_{a_i}^{a_{i+1}}\lambda(t)dt \quad (3)$$

where *m* is the number of intervals within each day.

Given that δ is the aggregation time interval, we can say that:

$$a_{i+1} = a_i + \delta \ \ and \ \ a_m - \ a_0 = m.\delta$$

The value of δ is very small compared to the whole study duration. So practically, the integrals in (2) and (3) can be evaluated using the following approximation:

$$\int_{a_i}^{a_{i+1}}\lambda(t)dt \approx \delta\lambda(t_i)$$

where $t_i = (a_{i+1} + a_i)/2$. The approximation error is of order $o(\delta^2)$. Hence, Equation (3) becomes:

$$l = \sum_{i=1}^{m} n_i log[\delta\lambda(t_i)] \ \ -\sum_{i=1}^{m}\delta\lambda(t_i) \quad (4)$$

Substituting the function $\lambda(t)$ given in (2) into the log-likelihood function and excluding the constants that does not depend on the parameters, equation (4) becomes:

$$l = \sum_{k=1}^{m} n_k \left(\mu + \sum_{i=1}^{k_o} [\alpha_i \sin(i\omega_o t_k) + \beta_i \cos(i\omega_o t_k)] + \sum_{j=1}^{6} \gamma_j I_j(t_k) \right) - \delta \sum_{k=1}^{m} exp\left(\mu + \sum_{i=1}^{k_o} [\alpha_i \sin(i\omega_o t_k) + \beta_i \cos(i\omega_o t_k)] + \sum_{j=1}^{6} \gamma_j I_j(t_k) \right) \quad (5)$$

Equation 5 can be rewritten as:

$$l = n\mu + \sum_{i=1}^{k_o} [\alpha_i S_i + \beta_i C_i] + \sum_{j=1}^{6} \gamma_j F_j - \delta \sum_{k=1}^{m} G_k \quad (6)$$

where:

$$S_i = \sum_{k=1}^{m} n_k sin\,(i\omega_o t_k), \qquad C_i = \sum_{k=1}^{m} n_k cos\,(i\omega_o t_k)$$

$$F_j = \sum_{k=1}^{m} n_k I_j(t_k) \qquad and$$

$$G_k = exp\left(\mu + \sum_{i=1}^{k_o} [\alpha_i \sin(i\omega_o t_k) + \beta_i \cos(i\omega_o t_k)] + \sum_{j=1}^{6} \gamma_j I_j(t_k) \right) \quad (7)$$

Notice that F_j is the total number of calls on day j. The terms S_i and C_i do not depend on the parameters while the terms G_k are exponential terms.

The ML estimators are obtained by taking the partial derivatives of the log-likelihood with respect to the model parameters: $\mu, \alpha_i, \beta_i,$ and each of γ_j. Hence, we obtain the following score equations:

$$\frac{\partial l}{\partial \mu} = n - \delta \sum_{k=1}^{m} G_k \tag{8}$$

$$\frac{\partial l}{\partial \alpha_i} = S_i - \delta \sum_{k=1}^{m} G_k sin(i\omega_o t_k) \tag{9}$$

$$\frac{\partial l}{\partial \beta_i} = C_i - \delta \sum_{k=1}^{m} G_k cos(i\omega_o t_k) \tag{10}$$

$$\frac{\partial l}{\partial \gamma_j} = F_j - \delta \sum_{k=1}^{m} G_k I_j(t_k) \tag{11}$$

The ML estimators are obtained by solving:

$$\frac{\partial l}{\partial \mu} = 0, \frac{\partial l}{\partial \alpha_i} = 0, \frac{\partial l}{\partial \beta_i} = 0, \frac{\partial l}{\partial \gamma_j} = 0 \; for \; j = 1,2,..,6$$

An implicit form of the solution to the first equation can be obtained easily as follows:

$$\hat{\mu} = log\left(\frac{\delta}{n}\sum_{k=1}^{m} G_k{}'\right) \tag{12}$$

where:

$$G_k{}' = exp\left(\sum_{i=1}^{k_o}[\alpha_i \sin(i\omega_o t_k) + \beta_i \cos(i\omega_o t_k)] + \sum_{j=1}^{6}\gamma_j I_j(t_k)\right) \tag{13}$$

The other score equations cannot be solved analytically. Therefore, we used Fisher scoring method to estimate the parameters.

Furthermore, we studied the significance of the proposed model using the covariance matrix of the estimators. ML estimation theory states that when the sample size is sufficiently large, as is the case of our call arrival data, the covariance matrix is equal to I^{-1} [40], where I is the information matrix obtained by evaluating the negative expectation of the Hessian matrix of the log-likelihood function. The diagonal elements of the information matrix are:

$$-E\left(\frac{\partial^2 l}{\partial \mu^2}\right) = \delta\sum_{k=1}^{m} G_k \tag{14}$$

$$-E\left(\frac{\partial^2 l}{\partial {\alpha_i}^2}\right) = \delta\sum_{k=1}^{m} G_k sin^2(i\omega_o t_k) \tag{15}$$

$$-E\left(\frac{\partial^2 l}{\partial {\beta_i}^2}\right) = \delta\sum_{k=1}^{m} G_k cos^2(i\omega_o t_k) \tag{16}$$

$$-E\left(\frac{\partial^2 l}{\partial {\gamma_j}^2}\right) = \delta\sum_{k=1}^{m} G_k I_j(t_k) \tag{17}$$

where the operator *E(.)* denotes the expectation of a random variable. The off-diagonal terms are computed by taking mixed partial derivatives of order 2.

We evaluated the variance terms for each parameter and then we used them to conduct Wald's significance test so that H_o: $\theta = 0$ against H_1: $\theta \neq 0$ where θ is any parameter of interest (α_i, β_i, γ_j and μ). Table 8 shows the values of the estimated parameters, their standard errors and p-values of Wald's test.

Table 8. Estimated parameters for λ(t)

Parameter	Estimated value	Std. Error	p-value
$\boldsymbol{\mu}$	12.4851183	0.0002360	< 2e-16
$\boldsymbol{\alpha_1}$	0.6244975	0.0002387	< 2e-16
$\boldsymbol{\alpha_2}$	0.3730669	0.0002675	< 2e-16
$\boldsymbol{\alpha_3}$	0.1122494	0.0002224	< 2e-16
$\boldsymbol{\beta_1}$	-1.2787258	0.0003443	< 2e-16
$\boldsymbol{\beta_2}$	-0.4221888	0.0002767	< 2e-16
$\boldsymbol{\beta_3}$	-0.1487193	0.0002205	< 2e-16
$\boldsymbol{\gamma}_1$	-0.2266414	0.0003971	< 2e-16
$\boldsymbol{\gamma}_2$	-0.5476155	0.0004534	< 2e-16
$\boldsymbol{\gamma}_3$	0.0833744	0.0003486	< 2e-16

The small magnitude of the parameters' p-values confirms that the considered parameters are significant. Parameters with p-values larger than 0.05 were removed since their presence would be a nuisance to the model and might contribute to variance inflation. The significance of the model's parameters reflect the significance of the model itself and that it explains the variability in the data as can be seen in the plot in Figure 22.

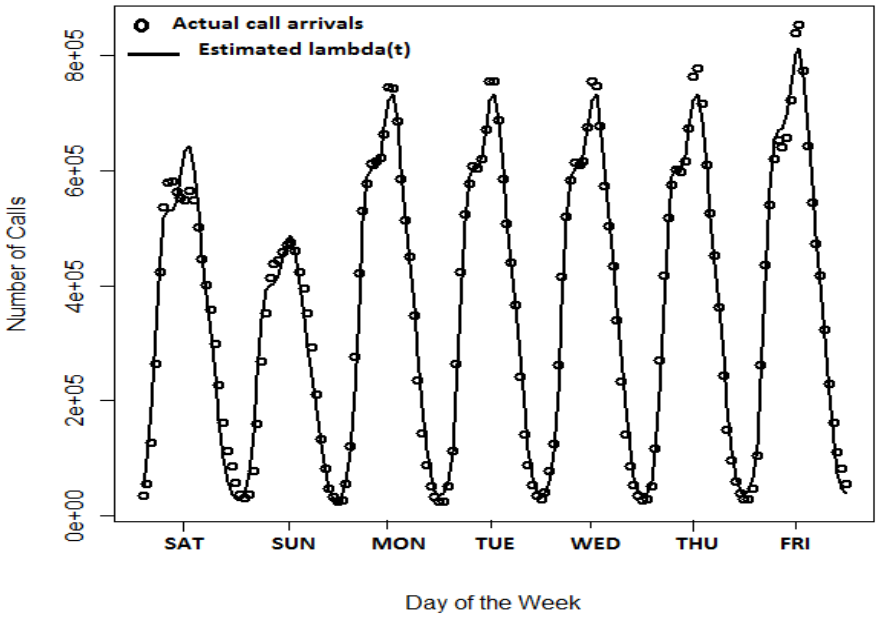

Figure 22. Fitting actual call arrivals to the suggested model

The model significance can also be evaluated by conducting the likelihood ratio test where the test statistic used here is evaluated as the ratio of the likelihood function for the restricted model (call arrivals follow a homogenous Poisson process) and of likelihood of the full model (call arrivals follow a NHPP with $\lambda(t)$). The null distribution of the test statistic is a *chi square* whose number of degrees of freedom equals the number of parameters minus 1. For our model and sample data, this value is equal to 34,676,131 with 8 degrees of freedom corresponding to a p-

value that is practically 0. Such very small p-value confirms our earlier results that the considered model is a very good fit to the data.

5.2.3 Traffic Prediction

The importance of using a significant model lies in the capability of such model to predict future data. In this section we use our proposed framework to construct a model and estimate its parameters based on data collected in week 1 and then we use the model to predict data for other weeks. We compare the predicted data to the actual data that we already have for these weeks. Figure 23 shows a plot of the predicted data against the actual data of two random weeks. The figure shows clearly that the actual observations fall very close to the curve of the estimated model.

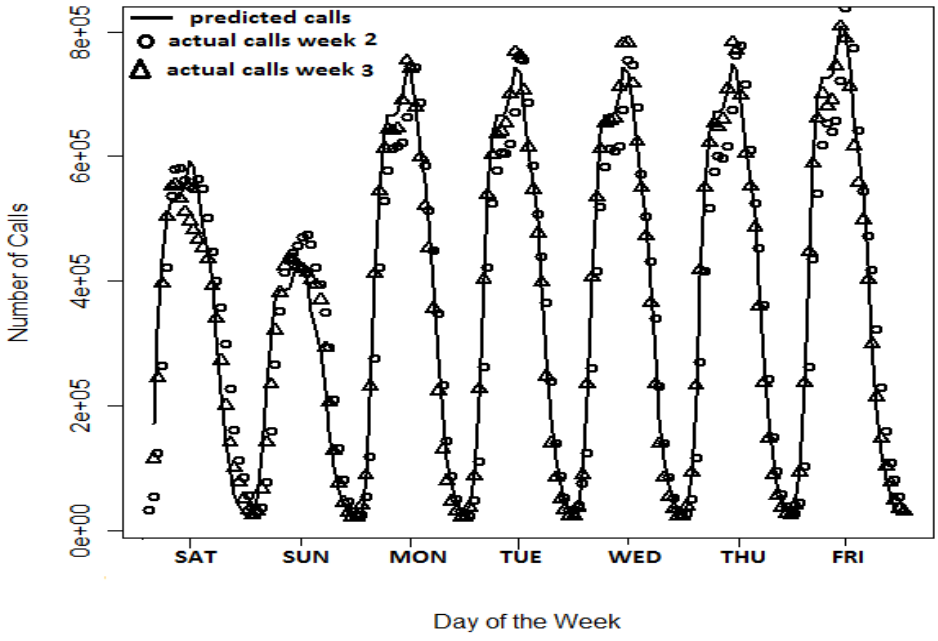

Figure 23. Predicted against actual call arrivals for two random weeks

5.2.4 Summary

The empirical data shows that the traditional Poisson process is not appropriate to model the VoIP traffic, while a non-homogeneous Poisson Process is able to capture the traffic behavior. A major contribution of this research is modeling the call arrival rate as a function of calendar time under the non-homogenous Poison Process framework. We validated the model behavior with real traffic data over several months. The statistical analysis of predicted data and actual data shows strong model validity and goodness-of-fit. This traffic engineering model could support network management systems to develop a dynamic resource allocation procedure. During the peak time of voice traffic, more network resources are allocated to the voice application. When voice traffic is low, more network resources are allocated for data services.

5.3 Normality Approximation of Call Arrivals under Heavy Traffic Condition

After presenting the frame work for modeling call arrival rate as a NHPP, we went one further step and proposed a new model for call arrival process under heavy traffic conditions. We use empirical and analytical evidences to prove that such call arrivals can be approximated as linear Gaussian processes. We show that this approach can provide an intuitive and accurate representation for different traffic patterns. In addition, the Gaussian approximation allows finding explicit mathematical equations for the model parameters and also provides easy model validation and significance testing. The model is illustrated by using large number of call records collected from the tandem network described in Section 4.1. We used least-square estimation

method to build the model and conduct goodness-of-fit tests to validate it. We achieved a coefficient of determination, R^2, of 0.9973. This means that 99.73% of the variability in the data is explained by the proposed model. The significance of the proposed model is confirmed empirically by its accurate prediction of future traffic.

Ideally engineers would like to use a model that can be easily fit to the data, and is directly related to the systems factors and variables and whose parameters have a physical meaning. Generalized linear models, such as the Poisson, which are used to fit discrete stochastic processes such as the number of calls, lack these simplistic characteristics. Gaussian linear models on the other hand benefit greatly from such properties.

Let *N(t)* denote the number of calls that arrive between time *t* and *(t-1)* then we can write the model as:

$$N(t) = \mu(t) + \sigma(t)\epsilon(t) \qquad (1)$$

where $\mu(t)$ and $\sigma(t)$ are the expected number of calls between *t* and *(t-1)* and their variance, $\epsilon(t)$ is the sampling error at time *t* which represents the random component of the number of calls and is assumed to follow the standard Gaussian distribution.

Advantages of using a Gaussian model are many. For instance, tests of significance for both the parameters and the model can be easily constructed and assumptions related to model building can be easily checked and validated. Also we can build confidence intervals for future observations that allow us to predict the system behavior.

The validity of such model resides in the fact that Poisson Process behaves like a Gaussian process when its expected value is large [41] as is the case in the tandem network which

operates under heavy traffic condition. The accuracy of such approximation is a direct consequence of the Berry-Esseen Theorem which puts a bound on the discrepancy between certain distributions and the Gaussian distribution. In the case of the call arrival process, the difference between the Poisson distribution and its Gaussian approximation at time *t* is inversely proportional to the square root of the expected number of calls at that time. This relation is shown in the Berry-Esseen equation below [42]:

$$\sup_{x\geq 0}|F_{Poisson}(x,\mu_t) - F_{Gaussian}(x,\mu_t)| \ \leq \frac{0.7164}{\sqrt{\mu_t}} \quad (1)$$

For example, if the phone system receives 100 calls during a time interval, then the approximation error is less than 0.072 which is a quite tolerable bound. Each one of the tandem offices from which the data was collected receives millions of calls every day making the Gaussian approximation more applicable.

5.3.1 Model Building

As we did in the NHPP section, we build a model for call arrivals that takes into consideration the daily arrival patterns and has the time-dependent mean function of:

$$\begin{aligned}\mu(t) = \ \alpha_o + \sum_{i=1}^{\frac{T}{2}} [\alpha_i \sin(i\omega_o t) + \ \beta_i \cos(i\omega_o t)] \ + \sum_{j=1}^{6} \gamma_j I_j(t) \\ + \sum_{i=1}^{T/2}\sum_{j=1}^{6} \rho_{ij} \sin(i\omega_o t) \ I_j(t) \\ + \sum_{i=1}^{T/2}\sum_{j=1}^{6} \varphi_{ij} \cos(i\omega_o t) \ I_j(t) \quad (2)\end{aligned}$$

The *sine* and *cosine* functions are used to build the periodic variation into the model. For a function with period *T*, there are *T/2* possible cycles since harmonics with higher frequencies (*i>T/2*) are aliased to frequencies between 0 and 0.5.

The function $I_j(t)$ is the Day Indicator where $1 \leq j \leq 6$ with 1 corresponding to Saturday, 2 to Sunday, 3 to Monday, 4 to Tuesday, 5 to Thursday, and *6* to Friday. $I_j(t)$ assumes the value 1 if $t \in$ *the day represented by j* and *0 otherwise*. The indicator function corresponding to the seventh day is intentionally removed to avoid having a model with linearly dependent variables. The coefficients ρ_{ij} and φ_{ij} are parameters that represent the effect of the interactions between the indicator function $I_j(t)$ and the harmonics $sin(i\omega_o t)$ and $cos(i\omega_o t)$ respectively. Adding these interaction terms allows us to investigate the relationship between a given harmonic term and the number of calls in a given day. The effects of the interactions coefficients ρ_{ij} and φ_{ij} are different from those of α_i and β_i which represent the effects of harmonic terms on the call numbers throughout the week.

Since the empirical data does not exhibit non-constant variability due to sampling error, we can safely assume that the model is homoscedastic and that the variance function $\sigma(t) = \sigma$ is constant over time. When that is not the case, we apply certain transformations to stabilize the data or use weighted least squares to estimate the parameters [43]. Examples of efficient transformations for count-type data include the logarithm, square root and quadratic root.

5.3.2 Parameter Estimation

We used the least-squares (LS) method [43] to estimate the parameters in the proposed model of $\mu(t)$. The advantages of least-squares estimation are that it does not require knowledge

of the underlying distribution of the error component $\epsilon(t)$ and when the model is linear it delivers explicit expressions for the parameters' estimates. In addition when $\epsilon(t)$ is assumed to have a Gaussian distribution, it is possible to make inferences about the parameters' significance and about the model's validity and usefulness. From which, we can use the model to make predictions about future observations.

As explained in [84], we processed call arrival data by aggregating arrivals into non-overlapping time intervals of length $\delta t = 1$ second, or 1 minute or 1 hour. Thus we will use the total number of calls within time intervals rather than the exact call time arrival.

Let $n_1, n_2, .., n_{m-1}, n_m$ denote the number of calls arrived at the system in non-overlapping time intervals $(t_1, t_2], (t_2, t_3], \ldots, (t_{m-1}, t_m]$, such that $t_k = t_{k-1} + \delta t$. Thus, least-squares estimators are obtained by minimizing the loss function in (3), which is expressed as sum of squared deviations between the observed and expected numbers of calls:

$$SS = \sum_{k=1}^{m} [n_k - \mu(t_k)]^2 \qquad (3)$$

replacing $\mu(t_k)$ by its expression in (2), we obtain:

$$SS = \sum_{k=1}^{m} \Bigg[n_k - \alpha_o - \sum_{i=1}^{T/2} [\alpha_i \sin(i\omega_o t_k) + \beta_i \cos(i\omega_o t_k)] - \sum_{j=1}^{6} \gamma_j I_j(t_k) - \sum_{i=1}^{T/2} \sum_{j=1}^{6} \rho_{ij} \sin(i\omega_o t_k)\ I_j(t_k) - \sum_{i=1}^{T/2} \sum_{j=1}^{6} \varphi_{ij} \cos(i\omega_o t_k)\ I_j(t_k) \Bigg]^2 \qquad (4)$$

The LS estimators are obtained when minimizing *SS*, with respect to each of the model parameters α_i , β_i, γ_j ρ_{ij} and φ_{ij}. This is done through taking partial derivatives of *SS* with respect to the parameters which leads to the following system of linear equations:

$$\sum_{k=1}^{m}[n_k - \mu(t_k)] = 0$$

$$\sum_{k=1}^{m} sin(i\omega_o t_k)[n_k - \mu(t_k)] = 0;\ i = 1, \dots, T/2$$

$$\sum_{k=1}^{m} cos(i\omega_o t_k)[n_k - \mu(t_k)] = 0;\ i = 1, \dots, T/2$$

$$\sum_{k=1}^{m} I_j(t_k)[n_k - \mu(t_k)] = 0;\ j = 1, \dots, 6$$

$$\sum_{k=1}^{m} sin(i\omega_o t_k) I_j(t_k)[n_k - \mu(t_k)] = 0;\ \ i = 1, \dots, \frac{T}{2};\ \ j = 1, \dots, 6$$

$$\sum_{k=1}^{m} cos(i\omega_o t_k) I_j(t_k)[n_k - \mu(t_k)] = 0;\ \ i = 1, \dots, \frac{T}{2};\ \ j = 1, \dots, 6$$

Define the m-dimensional vectors $n=[n_1,\dots,n_m]$, $1_m = [1, \dots, 1]'$ and $0_m = [0, \dots, 0]'$, and the coefficient vectors $\alpha = [\alpha_1, \dots, \alpha_{T/2}]'$, $\beta = [\beta_1, \dots, \beta_{T/2}]'$, $\gamma = [\gamma_1, \dots, \gamma_6]'$ and $\theta = [\alpha', \beta', \gamma']'$ where the prime sign is used to denote the transpose of a matrix or vector. Therefore, θ is the vector that contains all the coefficients in the linear model. Also, define the *(m*x*T/2)* matrices, M_1 and M_2 whose $(k,i)^{th}$ entries are respectively, $sin(i\omega_o t_k)$ and $cos(i\omega_o t_k)$; *(m*x*6)* matrix, M_3

whose $(k,j)^{th}$ entry is $I_j(t_k)$; and $(m\text{x}(6T/2))$ matrices, M_4 and M_5 whose $(k,(j-1)T/2+i)^{th}$ entries are respectively $sin(i\omega_o t_k)\ I_j(t_k)$ and $cos(i\omega_o t_k)\ I_j(t_k)$. If we define the $(m\text{x}(7T+6))$ matrix $X=[1_m, M_1, M_2, M_3, M_4, M_5]$, then the above system of $7(1+T)$ equations can be written as:

$$1_m'(n - X\theta) = 0$$

$$M_1'(n - X\theta) = 0$$

$$M_2'(n - X\theta) = 0$$

$$M_3'(n - X\theta) = 0$$

$$M_4'(n - X\theta) = 0$$

$$M_5'(n - X\theta) = 0$$

This set of equations can be further summarized as:

$$X'(n - X\theta) = 0_m \qquad (5)$$

which is equivalent to the equation:

$$(X'X)\theta = X'n \qquad (6)$$

If the size of the data, m, was larger than the number of parameters $7(1+T)$, the design matrix X would have a full rank, *i.e.*, its columns would be linearly independent, so $X'X$ would be nonsingular and the solution to the linear system of equations (4) would be:

$$\hat{\theta} = (X'X)^{-1}X'n \qquad (7)$$

However, this might not always be the case, and the solution to (6) might involve a generalized inverse of the matrix $X'X$, which we denote by $(X'X)^-$ and the solution is:

$$\hat{\theta} = (X'X)^- X'n \qquad (8)$$

The vector $\hat{\theta}$ contains the LS estimates of the parameters in model (2).

By combining equations (1) and (4), we can show that the covariance of $\hat{\theta}$ is $\hat{\sigma}(X'X)^{-1}$ or $\hat{\sigma}(X'X)^-$ depending on the rank of *X*, where $\hat{\sigma} = \sum_{k=1}^{m}[n_k - \mu(t_k)]^2 / (m - (7T + 8))$.

The standard errors of the parameters are the diagonal of the covariance matrix and they will come in handy when we conduct inferences about the parameters. Notice that in order to obtain these estimators; we do not need to make any assumptions about the distribution of the number of calls. The Gaussian assumption becomes of outmost importance when we test the significance of the parameters and their real contribution to model. The assumption is also crucial to test the usefulness of the model as a whole in explaining the behavior of the call arrival data.

To conduct the test of significance about a parameter: H_o: $\theta_i = 0$ against H_1: $\theta_i \neq 0$, we use Wald's test statistic, $\hat{\theta}_i / s.e.(\hat{\theta}_i)$, the ratio of the estimate of θ_i and its standard error as shown in Table 9. This statistic has a Student *t*-distribution with $m - (7T + 8)$ degrees of freedom. Since the number of observations is very large, the test statistic has an asymptotic Gaussian distribution. A parameter is significantly different from zero at *5%* level if its Wald's test statistic is larger than *1.96* in absolute value. When applied to each parameter in the model, the test of significance allows us to remove all non-significant parameters and keep only the variables and factors that seem to affect the behavior of the call arrival numbers.

Table 9. Parameters' estimation and std. errors

Parameter	Estimate	Std. Error	t value	Pr(>|t|)
α_o	401200	1377	291.359	< 2e-16
α_1	203157	1947	104.324	< 2e-16
β_1	-325874	1947	-167.341	< 2e-16
α_2	23597	1742	13.547	< 2e-16
β_2	-25652	1947	-13.173	< 2e-16
α_3	-27364	1590	-17.210	< 2e-16
β_3	-36522	1742	-20.968	< 2e-16
α_4	-28370	1485	-19.110	< 2e-16
γ_1	-83936	3152	-26.631	< 2e-16
γ_2	-169175	3079	-54.944	< 2e-16
γ_6	34884	3079	11.329	< 2e-16
$\varphi_{1,1}$	85330	4530	18.837	< 2e-16
$\rho_{1,1}$	-91083	4384	-20.777	< 2e-16
$\varphi_{2,1}$	-47459	4461	-10.639	< 2e-16
$\rho_{2,1}$	-26264	4368	-6.012	1.47e-08
$\varphi_{3,1}$	41411	4301	9.628	< 2e-16
$\rho_{3,1}$	-12106	4384	-2.762	0.006512
$\varphi_{1,2}$	136490	4354	31.345	< 2e-16
$\rho_{1,2}$	-141163	4354	-32.418	< 2e-16
$\varphi_{2,2}$	-12283	4354	-2.821	0.005477
$\rho_{2,2}$	53830	4266	12.617	< 2e-16
$\varphi_{1,6}$	-10486	4354	-2.408	0.017318
$\rho_{1,6}$	-11654	4354	-2.676	0.008320
$\varphi_{2,6}$	-15842	4354	-3.638	0.000384
$\rho_{2,6}$	-11968	4266	-2.805	0.005737

The significance of the model is investigated in the ANOVA Table 10 below:

Table 10. ANOVA model significance

Source	d.f.	Sum of squares	Mean Squares	F-Statistic	p-value
Model	25	1.12 e+13	4.47 e+11	2478.71	<2e-16
Error	141	2.55 e+10	1.81 e+8		
Total	166	1.12 e+13			

The very small p-value indicates that the proposed model is highly significant and explains the different patterns and the general behavior of the call arrival data. This is confirmed by the large value of coefficient of determination (adjusted for the number of variables) $R^2 = 0.9973$. This means that 99.73% of the variability in the data is explained by the proposed model.

Figure 24 illustrates the proposed Gaussian model fitted to actual data.

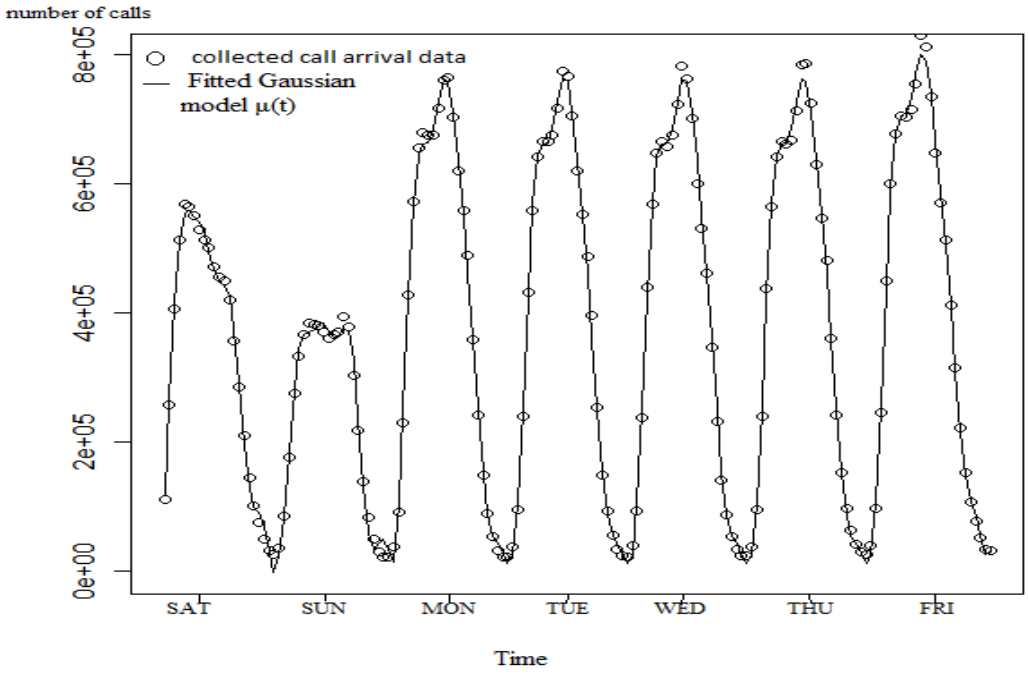

Figure 24. Fitted Gaussian model $\mu\hat{}(t)$against collected data

5.3.3 Model Validation

The estimation and significance testing operations rely heavily on the validity of the assumptions we make. The analysis conducted in the previous sections is based on the following assumptions:

- The number of calls per time unit follows a Gaussian process
- The proposed model in (2) is unbiased and the variance of the observed number of calls is constant.

Checking these assumptions is through the analysis of the residuals defined as the difference between observed number of calls n_k and the estimate of the expected number of calls $\hat{\mu}(t_k)$ (also known as the predicted values), we denote by:

$e(t_k) = n_k - \hat{\mu}(t_k), k = 1, \dots, m.$

When the numbers of calls follow a Gaussian process, the residuals themselves follow a standard Gaussian (Normal) distribution. The results of normality tests are presented in Table 11 below:

Table 11. Normality test results

Normality Test	**p-value**
Anderson-Darling	0.2708
Kolmogorov-Smirnov	0.8346
Shapiro-Wilks	0.4129

The null hypothesis is the normality of the residuals distribution. We can see in Table 11 that the p-values are large. Therefore, we conclude that the normality assumption holds and that the call arrival rate follows a Gaussian distribution.

The unbiasedness of the model is verified by plotting the residuals against time. Such graphs exhibit some sort of nonrandom pattern when the considered model is biased. The graph shown in Figure 25 bellow shows a completely random behavior of the residuals against time. This confirms the accuracy of our proposed model. The graph also shows that there is no reason to believe that the residuals' variance changes over time.

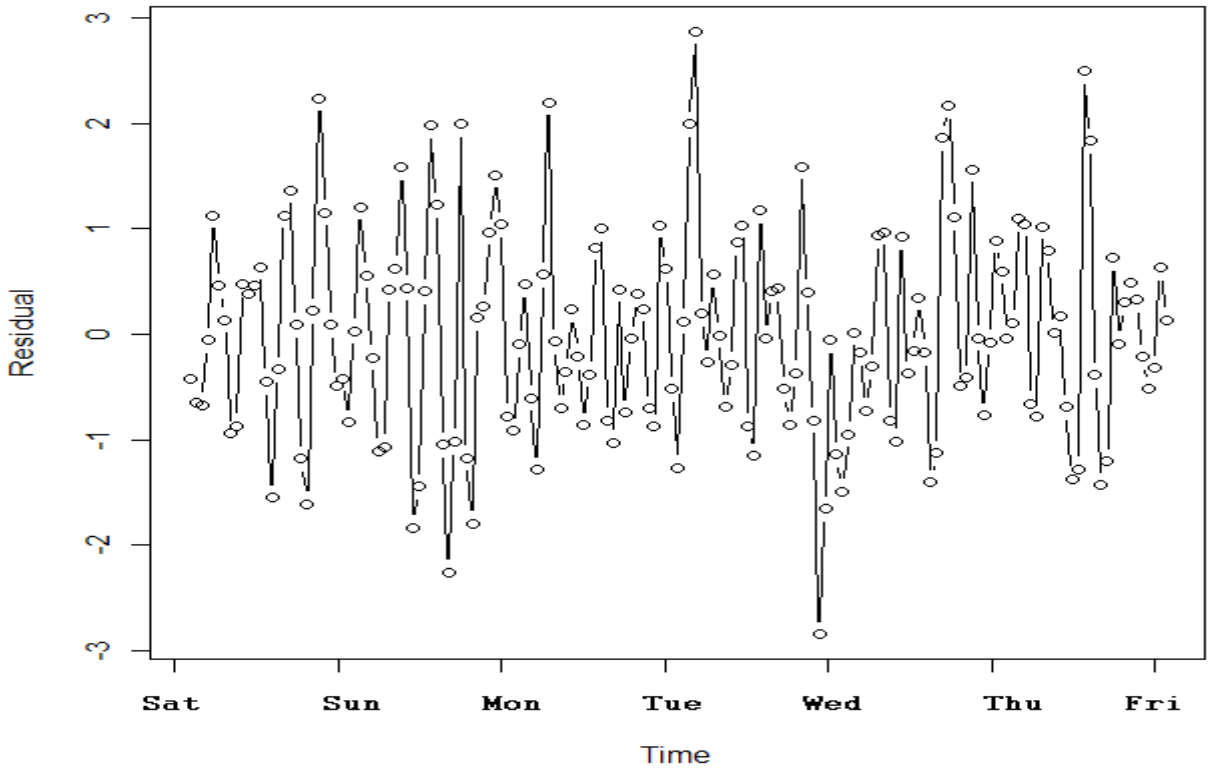

Figure 25. Residuals against time

Furthermore, the stability of the process is checked by plotting the residuals against the predicted values as shown in Figure 26. The range over which the residuals vary does not change with the predicted values which confirms the homoscedasticity (constant variance) of the process. This conclusion is also reached through homoscedasticity tests of hypothesis such as the celebrated Engle's Lagrange multiplier [85] and Li-McLeod [86] tests. These tests are used to detect heteroscedasticity, or non-constant variance, when the explanatory variable is time. In his award winning work [87], Engle explained that non-constant variance is generated by an autoregressive conditional heteroscedasticity (ARCH) effect of the errors. This effect can be detected by testing the significance of autocorrelation coefficients of squared residuals at different lags. This class of tests is successfully used by financial engineers, econometricians and time series analysts to investigate the non-stationary behavior of stocks, futures and bond interest rates [88].

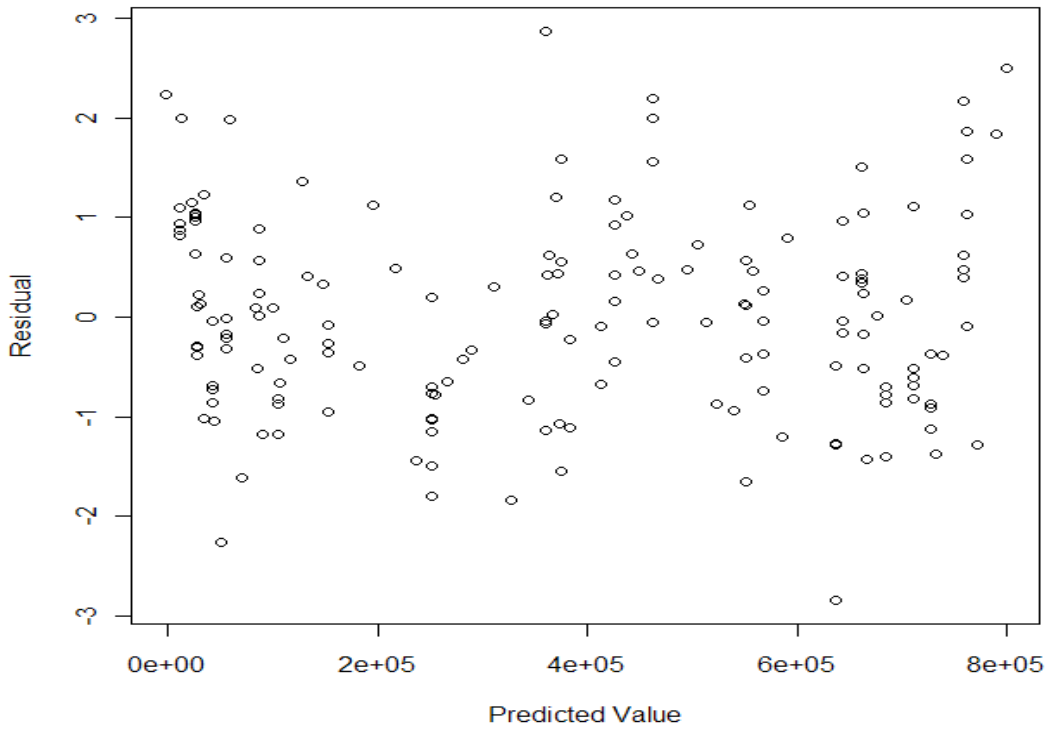

Figure 26. Residuals against the predicted values

5.3.4 Prediction and Model Comparison

The usefulness of model lies in its ability to explain the behavior of the system under study and in predicting the future state of the system [89]. In section 5.2.3 we proved that our proposed NHPP model can be used to predict future traffic behavior. Hence, our Gaussian approximation can be useful only if we can use it in a similar way for prediction. In this section, we use the proposed methodology to construct a model based on data collected in week 1 and then use the model to predict data for following two weeks. In Figure 27, we compare the predicted call numbers based on week 1's model with the observed (actual) call numbers that we collected in weeks 2 and 3 on the same tandem switch. The figure shows that the actual observations are very close to the curve of model prediction.

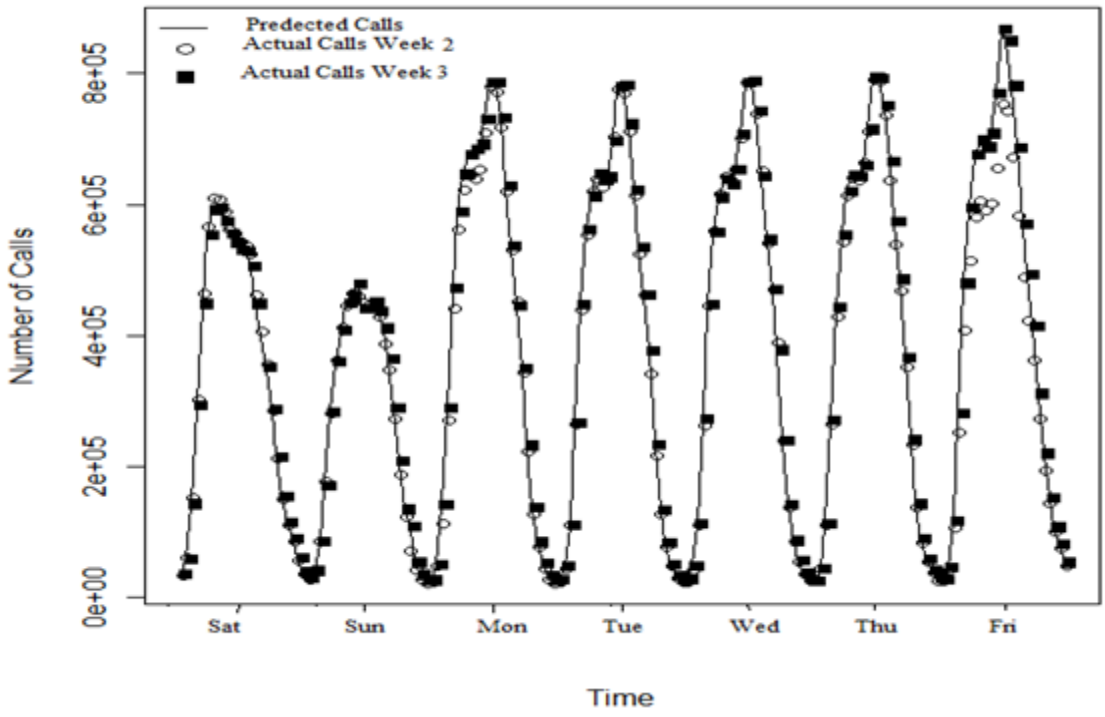

Figure 27. Predicted against actual call arrivals for two random weeks

The Gaussian model for call arrival rate allows us to build different time-dependent models based on the specific engineering requirements. For example, one might consider the variation of call arrivals from one week to another, or from one month to another, etc. It is also possible to use our methodology to consider the variation of calls from one hour to another. Holidays, and special days can be effectively modeled by giving them indicator functions.

As demonstrated in Section 5.3, the Berry-Esseen ensures that the Gaussian approximation to Poisson processes model is accurate when the arrival rates are large. This is confirmed by the plots in Figure 28, which represents Gaussian and Poisson fits to the call arrival data. The difference between the two models is insignificant, which justifies the proposed approximate model.

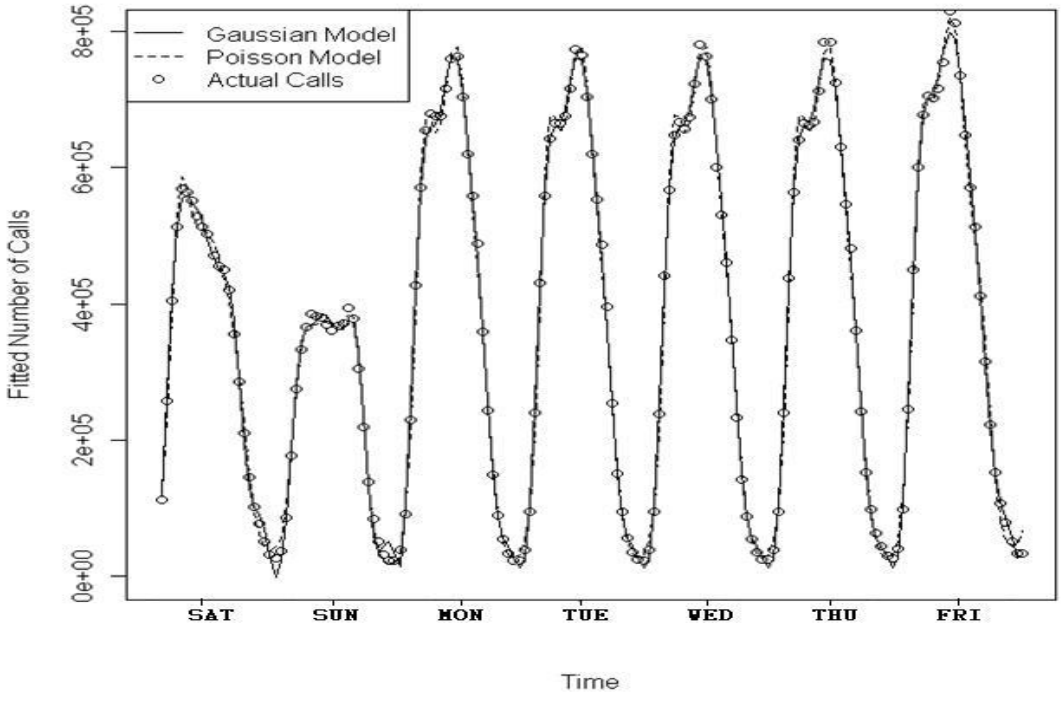

Figure 28. Comparison between the Poisson and Gaussian Models

5.3.5 Summary

In this section we fit a linear Gaussian model to call arrivals under heavy traffic conditions instead of traditional Poisson models. The choice of such model is motivated by the simplicity of Gaussian process and by its adequacy to fit high traffic call arrivals. The benefits of using Gaussian models are: (1) The model is intuitive and easily interpretable (2) The parameters are easily estimated and (3) The model is easily validated. We provide mathematical details to justify the Gaussian assumption, and to assess the performance of the proposed model through the ANOVA significance test. Then we check the model adequacy through goodness-of-fit tests and split-sample validation.

CHAPTER 6

6 VSIM Description and System Design

After the traffic analysis and modeling, we implemented the resulting models to engineer the VoIP network. The system is designed to integrate the proposed Call Arrival Rate and the Call Holding Time models. In addition, the system provides an alternate non-parametric engine for increased accuracy and reliability. The traffic engineering model provides (a) prediction of future traffic, (b) resource requirements estimation, (c) traffic graph generation, and (d) service quality reports and analysis. The mathematical models that we studied for the traffic are complex and hence a queuing analytical solution is not feasible. Therefore, we used the stochastic approach and developed a simulation system for VoIP traffic engineering. The proposed system is called the VoIP traffic engineering Simulator (VSIM). In this chapter we will describe the VSIM and its capabilities, specifications, design and development environment.

6.1 Development Approach

Because of the complexity of the Call Arrival and the Call Holding Time models, an analytical solution with mathematical formula is too complex, if not impossible. We conclude that a solution like the Erlang-B formula for modern VoIP traffic engineering is not feasible. Therefore, we will use simulations to study the relationships between model parameters and to estimate system performance under various traffic loads.

Our approach combines both statistical analysis and stochastic modeling where we collect live traffic information from a telecommunications network. The traffic data is used to build the Call Arrival model and the Call Holding Time model as discussed in Chapter 5. These models are then be used to predict the future traffic intensity. In our preliminary work, we used the data of previous week to predict traffic behavior of the next two weeks. In addition to traffic prediction, VSIM provides an intensive simulation environment that facilitates network resource and quality studies. The integration of the Call Arrival Rate and Call-Holding Time is traffic intensity, also known as Erlang. Unlike the Erlang-B model, traffic intensity in this study is a function of time rather than a constant number. As a result, our proposed system can be adopted in a resource allocation algorithm to dynamically allocate resources (example: bandwidth) to meet the traffic demand. The final output of the Traffic Engineering simulation model is the service quality report and it is measured by the call blocking probability or the required resources.. The functional modules of the simulation system are given in Figure 29.

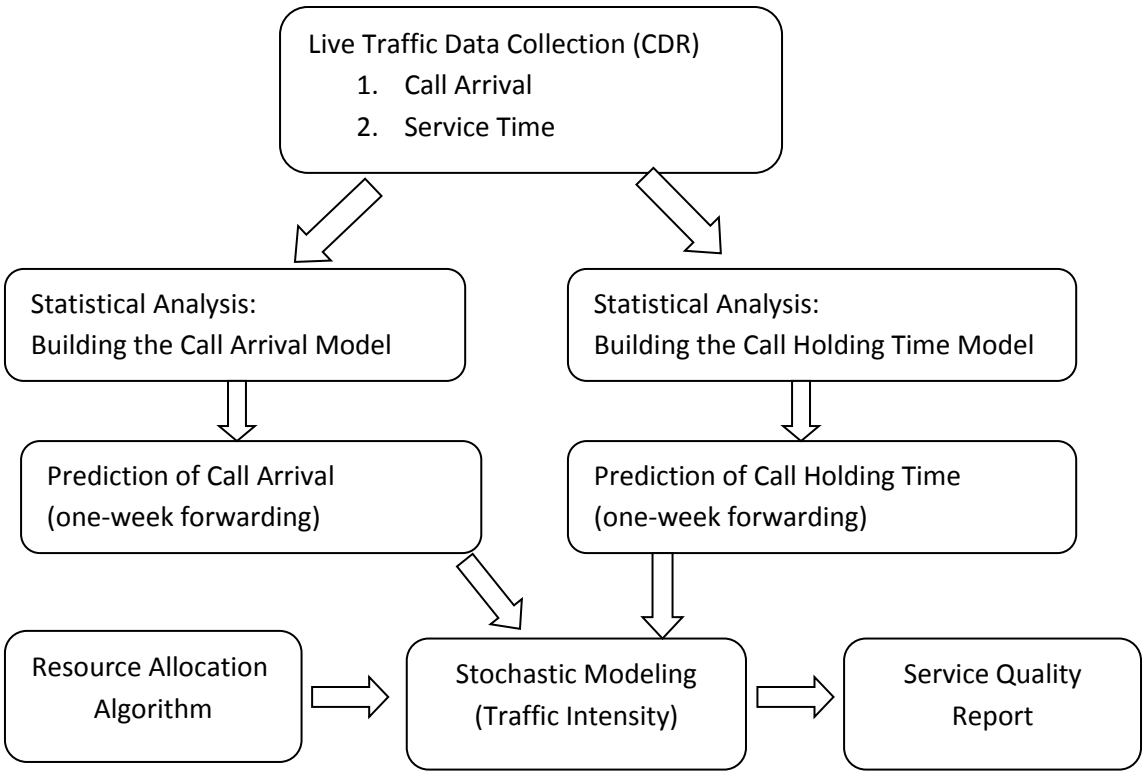

Figure 29. Functional Modules of VSIM

6.2 VSIM Design

VSIM is a modular traffic engineering suite composed of multiple tools performing different functions. The core of VSIM is a Java-based simulator that implements both parametric and non-parametric simulation engines. VSIM includes a traffic collection module which is connected to a production VoIP network for near real-time traffic data collection. The collected data is processed and then it is passed to the various VSIM modules for traffic prediction, resource optimization, and quality of service calculations. Figure 30 shows a high level design of the proposed traffic engineering system.

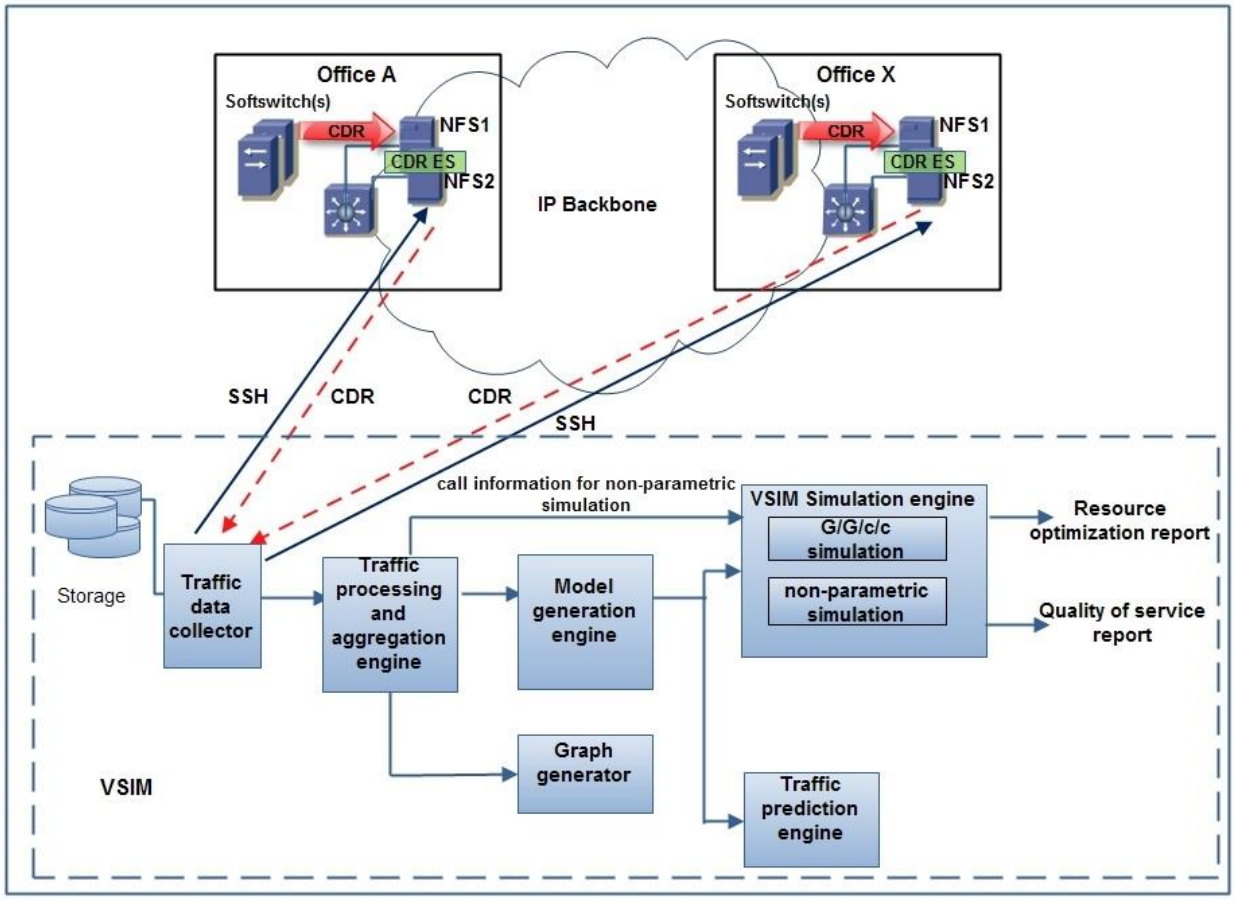

Figure 30. VSIM high level design

The major components of VSIM are:

6.2.1 Traffic Data Collector:

This is a Linux machine connected to the internal management network of the VoIP carrier. The server has access to all the distributed Call Detail Record (CDR) servers in all the remote switch sites. Data collection is performed as described in Section 4.1.2. The network under study processes over 400 million minutes of traffic every day. Because of these huge amounts of traffic information, we will take samples of traffic for different central offices. Also

we filter out non-essential fields of each CDR before transferring the data to the collector. Traffic collection and filtering process is achieved by a combination of Perl and Linux shell scripts. Traffic information is stored on an external storage for portability and extendibility. Figure 31 shows a sample of the collected raw call data. Each raw carries information for one call (CDR). The CDR Fields that we collect are (respectively):

- Record type (STOP means a completed call, ATTEMPT means a call attempt that was not completed)
- Switch name
- Start date
- Start time
- End date
- End time
- Call duration in 100s of seconds.
- Trunk group name (used to categorize traffic into landline, wireless, or VoIP)

Notice that for ATTEMPT records we don't have end date and end time because the call did not get through for some reason.

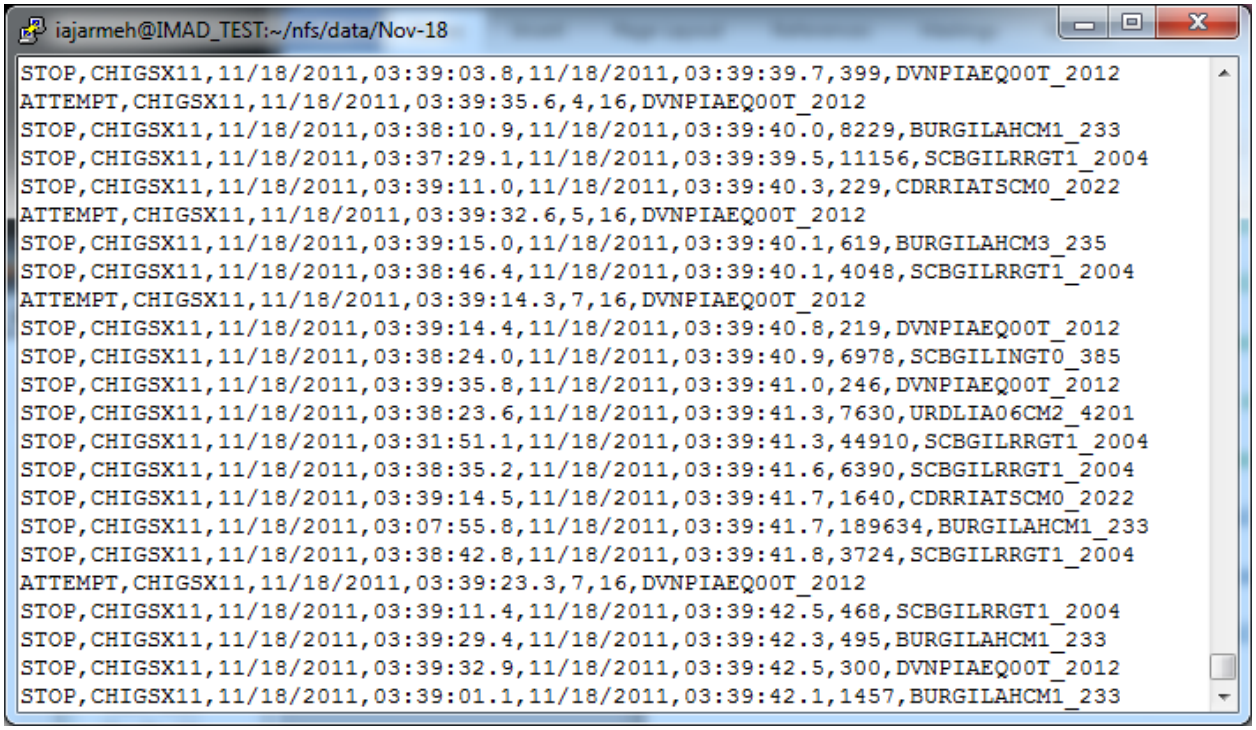

Figure 31. Sample collected call data

6.2.2 Traffic Processing and Aggregation Engine

This is a comprehensive library of Perl and Linux shell scripts that we developed in order to process and aggregate the raw traffic information into a format that is suitable for VSIM modeling and simulation engines. A highlight of the processing and aggregation process is given in Section 4.1.2. This engine has two major outputs: (a) call holding time information in the form of TOA (time of arrival) against call duration and (b) call arrival rate aggregation in the form of the number of calls that arrived within a certain time interval. The aggregation engine provides 1, 10, 100, 1200, and 3600 seconds aggregated call arrival information. Aggregation interval is a parameter that can be entered to the aggregation process and it is used to determine the accuracy and confidence interval of the modeling and prediction.

6.2.3 Model Generation engine

This is a critical module for VSIM. The input for this model is:

- Processed traffic data coming from traffic processing and aggregation engine.
- Modeling generation variable such as the required number of harmonics for call arrival linear model.

The output of this model is:

a. Estimated NHPP generalized linear model for call arrival rate.

The NHPP is constructed and estimated using R-language as explained in section.5.2. Figure 32 shows an example of the NHPP generalized model estimation output. The output shows the used generalized model, its estimated parameters, standard error and p-value for each parameter. The example shows that we are 4 harmonics for the generalized linear model.

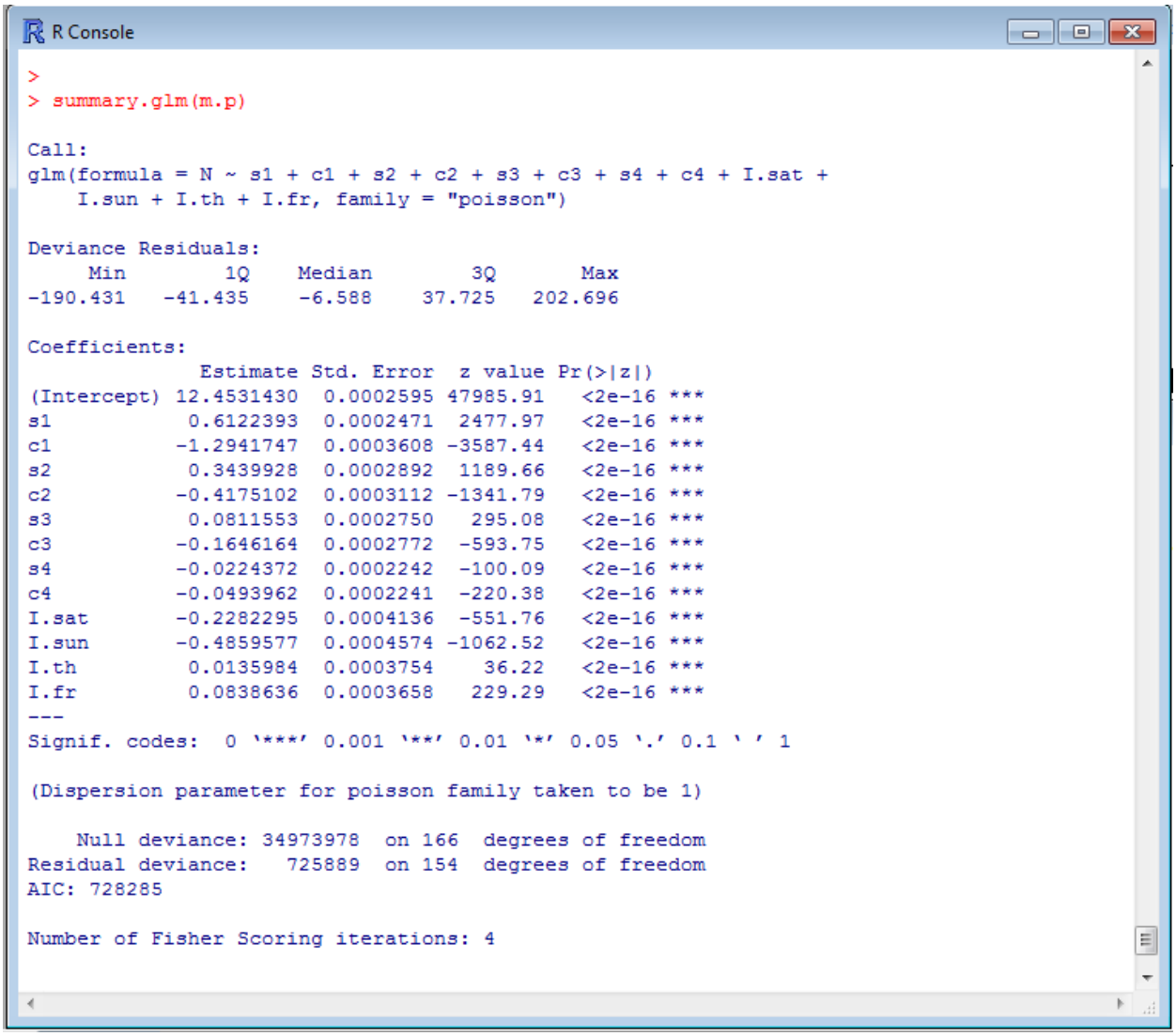

Figure 32. Sample modeling output (NHPP model estimation)

The model generation engine also produces a graph that shows the actual input traffic data against the generated model. An example is shown in Figure 33.

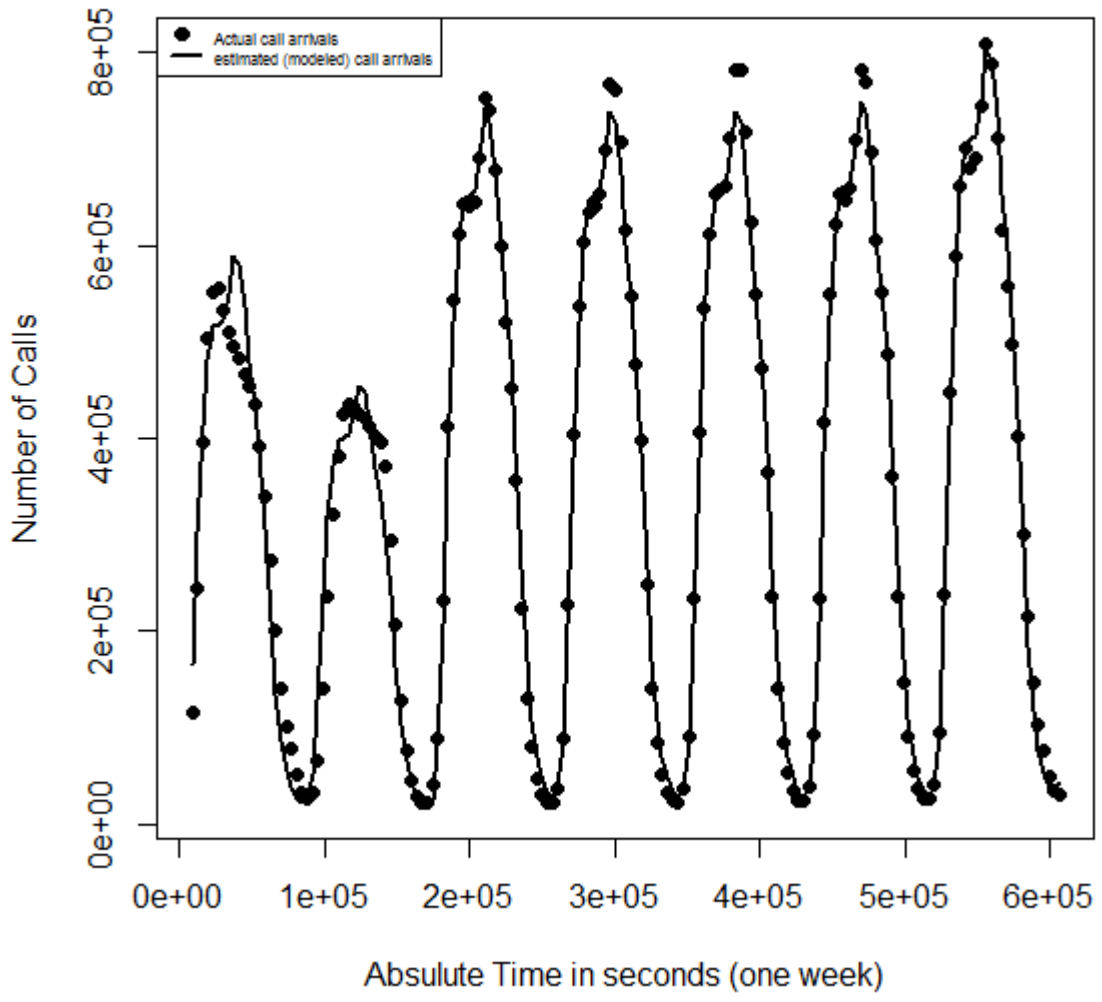

Figure 33. Sample modeling output (NHPP model graph)

The NHPP model generation engine generates multiple statistical values and graphs that show the accuracy of the model and its estimated parameters. For example Figure 34 shows a plot of residuals VS time. The random output means that chosen model is significant.

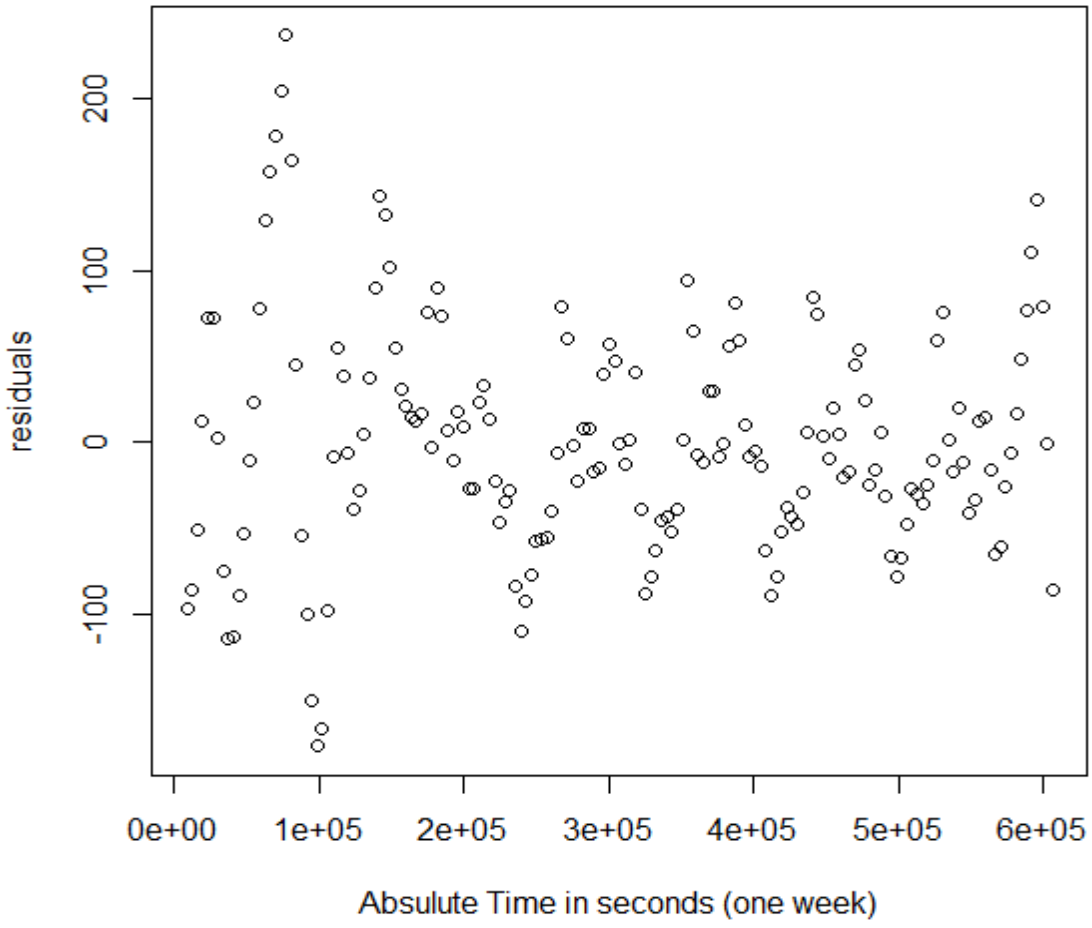

Figure 34. Residuals Vs Time

b. Generalized Gamma model for call holding time.

We used R language and SAS software in order to implement methodology and statistical process described in section 5.1. The choice for using SAS for this problem was based on the difficulties we faced in implementing and estimating the generalized Gamma model using R-Language. The modeling process starts by showing a histogram of the input data, and then the data is fit to a Generalized Gamma model. An example of the histogram data is shown in Figure 35.

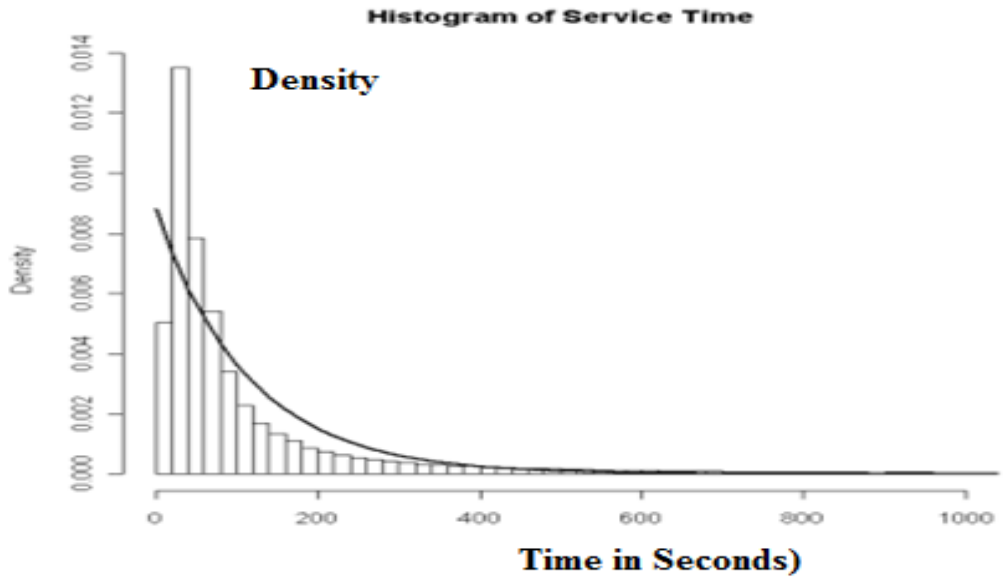

Figure 35. Histogram of call holding time

Generalized Gamma model parameters are estimated, and a graph is generated for the model as shown in Figure 36.

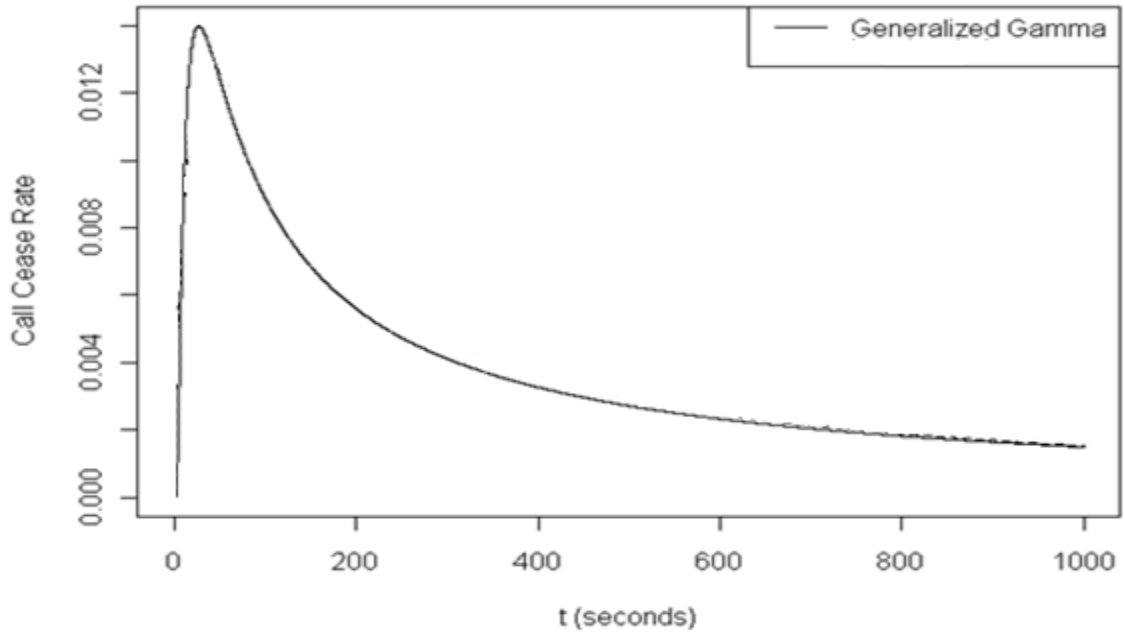

Figure 36. Call holding time modeling

6.2.4 VSIM Graph Generator:

The graph generator utilizes gnuplot and R-language in order to generate graphs for call holding time as well as call arrival rate and. The graphs will provide different time aggregations and scales for various engineering and business needs. In this research, we used VSIM to generate many graphs for analysis and validation of the model. Some samples of the graphs are presented in this thesis.

6.2.5 VSIM Traffic Prediction Engine:

This is a utility that takes traffic data for a certain period of time as input, constructs a model based on this data, and then uses this model to predict future traffic patterns. Prediction can be done using NHPP modeling and the normal approximation modeling. Figure 37 shows an example of the traffic prediction using NHPP model, and compares it with the actual data. Note that traffic data of week-1 is used to predict the incoming traffic of week-2 and week-3.

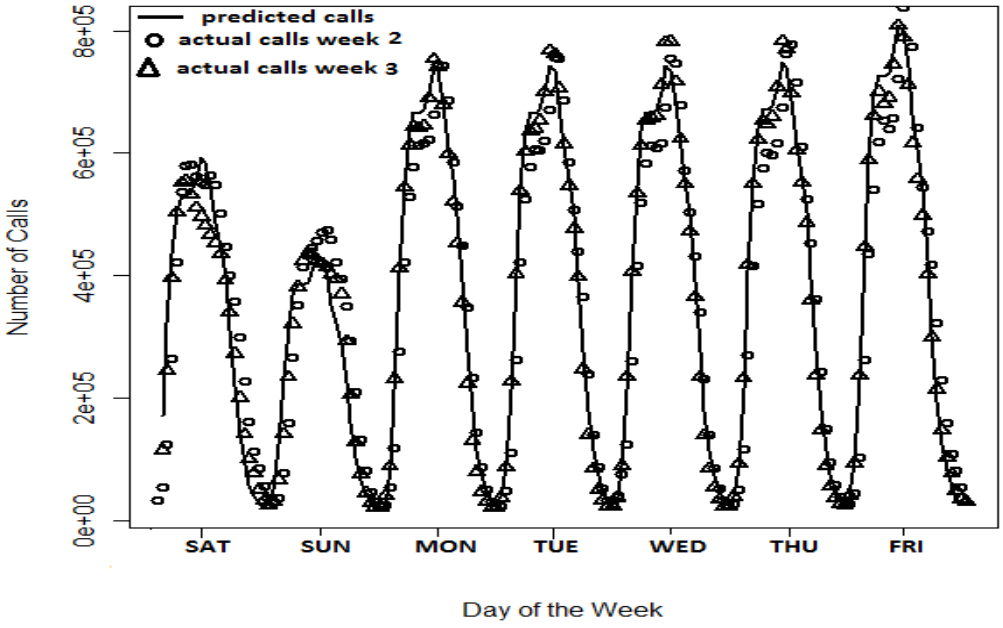

Figure 37. Prediction using NHPP model

Figure 38 shows an example of traffic prediction using the normal approximation model.

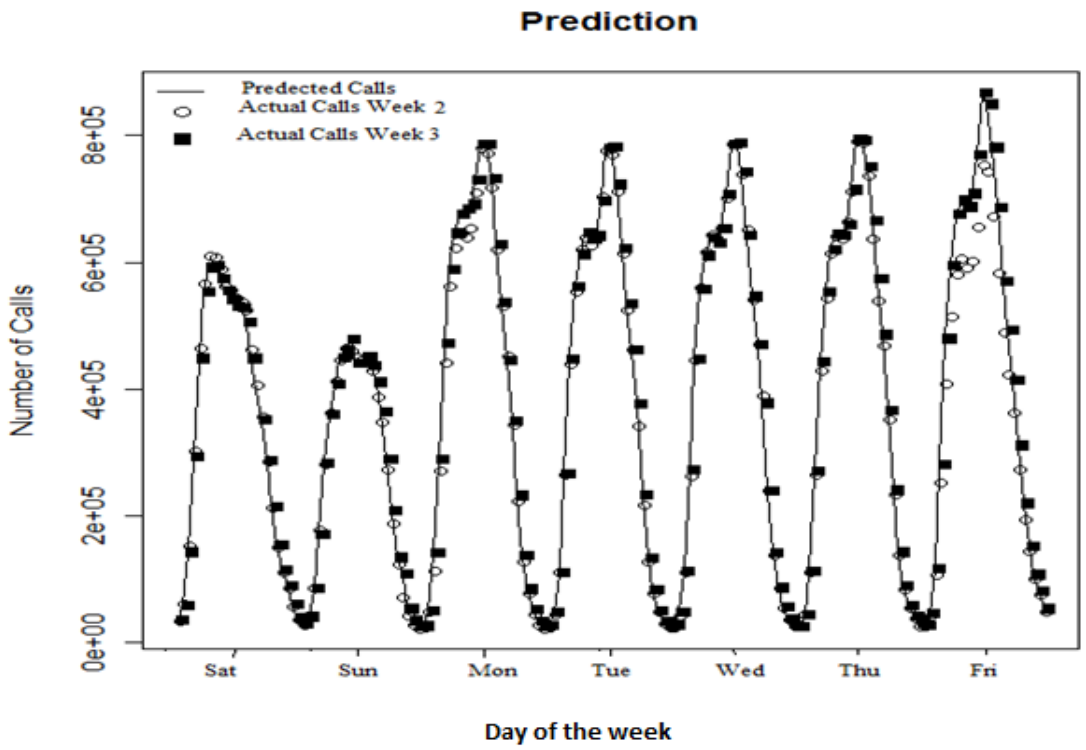

Figure 38. Traffic prediction using the Gaussian/normal model approximation

From Figure 37 and Figure 38, we can see that both NHPP and the normal/Gaussian approximation models yield accurate prediction results. It is easier to build the Gaussian model and estimate its parameters, and traffic prediction using Gaussian approximation is easier to perform and requires less statistical and mathematical analysis than that of NHPP model. However, the normal approximation assumes heavy traffic condition (high call arrival rate) as we explained in section 5.3. Hence, NHPP-based prediction is suitable for all traffic scenarios but it is more complex. In case of heavy traffic, we can use the simpler Gaussian-based prediction and still achieve comparable results.

6.2.6 VSIM simulation Engine

This is the core of the VSIM system. This engine is a discrete event simulator [65] in which the VoIP network is modeled as chronological sequence of call arrivals and terminations. VSIM is a Java-based simulator and its input is traffic models taken from the model generation engine. The next chapter provides the design, development, validation and verification of VSIM simulation.

CHAPTER 7

7 VSIM Simulation Implementation and Experiments

It has become clear that it is necessary to find modern traffic engineering models that can help design cost-efficient systems and study their performance under various conditions. In this chapter we provide a new VoIP simulation suite that consists of a parametric simulator based on Non-homogeneous Poisson Process (NHPP) call arrival model, and a non-parametric simulator based on real traffic data. Our simulators are validated against real call data obtained from multiple offices of a production VoIP carrier network. The purpose of the simulator is to provide a stochastic solution for the traffic problem given the modern traffic models developed in this research. The goal is to build a simulation system that can provide resource optimization and quality of service reports for VoIP networks. This data can be used by engineers in order to optimize network design, and build cost-efficient systems. A dynamic resource allocation scheme can be developed based on the resource and performance reports generated by our simulator. Such scheme, can dynamically allocate network resources according to the traffic demand which can be more useful for converged networks where the same network is shared

between voice and data, more bandwidth can be dynamically allocated for voice during the busy hours, and after the busy hours more bandwidth can be given to the data applications.

7.1 Introduction

The majority of the previous work in VoIP traffic engineering and modeling is based on the exponential approximations for call arrival rate and call holding time [90] [91] [92] [93] . The exponential approximation allows finding analytical solution for the traffic queuing model but the approximation might be too aggressive that it will result in poorly engineered systems. The Erlang B model was introduced several decades ago to solve the phone system traffic queuing problem. This model is based on the traffic intensity of the busiest hour in the busiest week of the year (Busy Season Busy Hour: BSBH). BSBH traffic is assumed constant throughout the entire year and its arrival rate is modeled as a Poisson/exponential distribution. This assumption makes traffic calculations easier but using a constant call arrival rate for the entire year causes inefficient resource utilization.

In our call arrival process analysis and modeling work [84] we proposed using a Non-homogeneous Poisson Process (NHPP) for the call arrival rate. In NHPP modeling call arrival rate is a non-constant function of time. Whereas in the legacy Erlang B approach calls are assumed to arrive according to a Poisson process with a constant arrival rate. Therefore, using NHPP helps avoid the approximation and assumption errors associated with a constant arrival rate over the whole engineering period. Our NHPP model development was based on real call data extracted from a production VoIP carrier network. Examining the arrival data, we constructed a model that describes the variation of call arrival rates during a week since traffic patterns were observed to be repeated weekly. It is common in statistical analysis to model the

logarithm of λ(t) instead of λ(t) itself for *count* data. Such transformation would guarantee that the estimate of the intensity function is always non-negative. Our model takes into consideration the daily arrival patterns and has the time-dependent intensity function of:

$$log[\lambda(t)] = \mu + \sum_{i=1}^{k_o} [\alpha_i \sin(i\omega_o t) + \beta_i \cos(i\omega_o t)] + \sum_{j=1}^{6} \gamma_j I_j(t) \qquad (1)$$

where: $\boldsymbol{\lambda(t)}$ is a function of time (t).

$\boldsymbol{I_j(t)}$ is day Indicator function where $\boldsymbol{j}$ is the day of the week. The value of $\boldsymbol{I_j(t)}$ is 1 if the time $\boldsymbol{t \in j}$ and 0 otherwise. $\boldsymbol{k_o}$ is the number of harmonics in the model. $\boldsymbol{\mu}$ represents the model central tendency without daily effects. $\boldsymbol{\gamma_j}$ is the effect of day *j* and represents the difference between $\boldsymbol{\mu}$ and the mean number of calls for day *j*. $\boldsymbol{\alpha_i}$, and $\boldsymbol{\beta_i}$ are the contribution of the *ith* harmonic to the model.

We used Maximum likelihood estimation to fit the proposed λ(t) to the actual call arrivals. In addition, we used likelihood ratio test and Wald's test in order to verify the significance of the model and its parameters. All the statistical test results verify that call arrivals is best fit by a NHPP rather than a constant Poisson process. We provide the detailed statistical analysis in section 5.2. In this chapter, we introduce a comprehensive VoIP simulation suite (VSIM). VSIM consists of a G/G/c/c simulation model based on NHPP call arrival rate and also a non-parametric simulator based on real traffic data. The simulation models are validated against traffic data collected from an operational VoIP carrier network.

7.2 Telecommunication System Simulation

In telecommunication traffic engineering, it is always preferable to find analytical solutions for the queuing and traffic problems. However, the analytical solution might involve too many approximations in order to fit the data into exponential or other probability distribution functions. Such approximations will result in inaccurate engineering results. Simulation approach offers accurate and flexible model construction and validation, and can be used whenever analytical solutions are not practical [94].

Simulation models can be discrete or continuous. Discrete event simulations are suitable for problems where variables change in discrete time fashion. On the other hand, continuous simulations are suitable for problems in which the variables might change continuously [95]. Discrete event simulations are suitable for telecommunication network queuing problems since the events happen on discrete times [96]. Using discrete event simulators has been attracting more researchers' attention during the past few years because such simulations can help solving sophisticated problems which are impossible to solve using analytical approaches [94] [97]. In addition, the availability of low-cost powerful computers and capable simulation packages makes the simulation-based solutions more accurate, capable, reliable, and easier to implement.

With the rapid increase of VoIP residential, enterprise, and carrier deployments, researchers realized the need for modern traffic simulation models that can be used in studying and designing reliable and cost-efficient VoIP networks. In [98] VoIP traffic sources were modeled as on-off sources with exponentially distributed of on-and-off times. In [99] and [100] the authors used Markov Modulated Poisson Process (MMPP) traffic model to analyze VoIP performance for wired and wireless networks. In [101] [102] [103] [104] the authors provide

VoIP traffic performance and evaluation simulation tools that focus on the packet performance without taking into consideration the distribution of calls arriving at the system which will have significant impact on the packet performance and QoS design. In this work we go a further step by providing two VoIP traffic simulators based on modeling the calls arriving at the system. The first one is a parametric model and uses a NHPP to represent the time-dependent call arrival rate, and the second one is non-parametric and uses the real traffic data to simulate the system behavior.

7.3 VSIM simulation engines

VSIM simulation model is part of a larger traffic engineering system that starts by collecting call data from a production network. The collected data is processed and then fed into the NHPP parameter estimation model. NHPP model parameters are passed to VSIM to be used in G/G/c/c engine. For non-parametric simulation we skip NHPP model estimation process and feed the processed call arrival and call holding time data directly into VSIM. The simulators we created during this research are built using traffic models developed based on real traffic data of a large production VoIP carrier network. We followed the framework described in section 5.2 in order to generate the NHPP model and estimate its parameters.

A closer look at the raw traffic pattern we notice large variation in the arrival rate. For example, at one second we might receive 10 calls and at the next second we might receive no calls. This variation is smoothed if we average the traffic data over longer time intervals. Figure 39 below shows the raw traffic data for 1s, 10s, 3600s averages, and the generated NHPP model. The figure illustrates the accuracy and significance of the generated model. This accuracy has

been established through the extensive mathematical and statistical analysis we provided in section 5.2. The accuracy of the input NHPP model will result in accurate simulation results as proven in section 7.5.

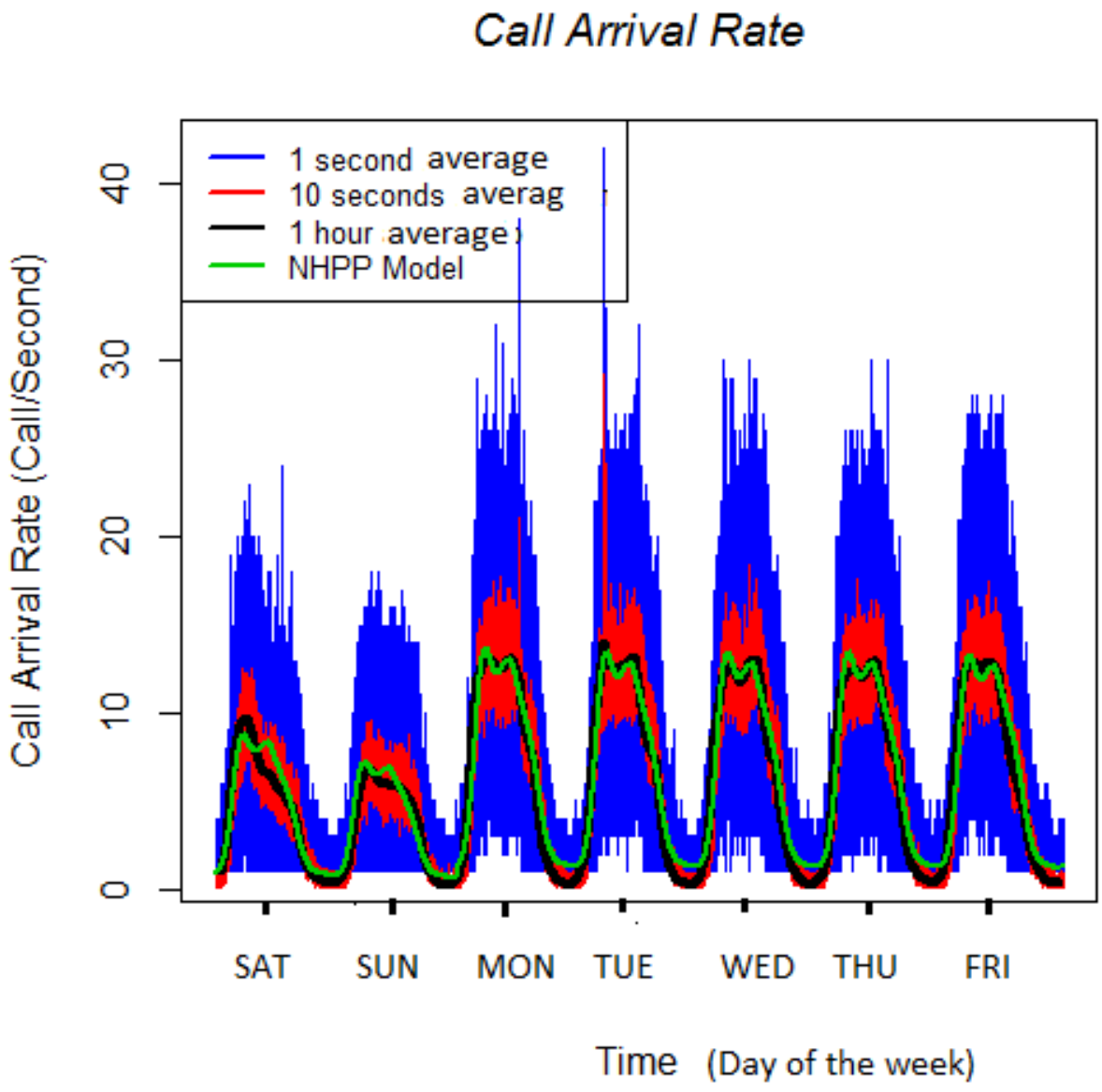

Figure 39. Call arrival data analysis and modeling

The VoIP simulation model (VSIM) estimates:

a) GoS (blocking probability): this scenario is used in sensitivity analysis for system design. In this case we vary the number of IP trunks and observe the effect on GoS (blocking probability for a certain traffic pattern. Figure 40 shows a sample of VSIM output for this case.

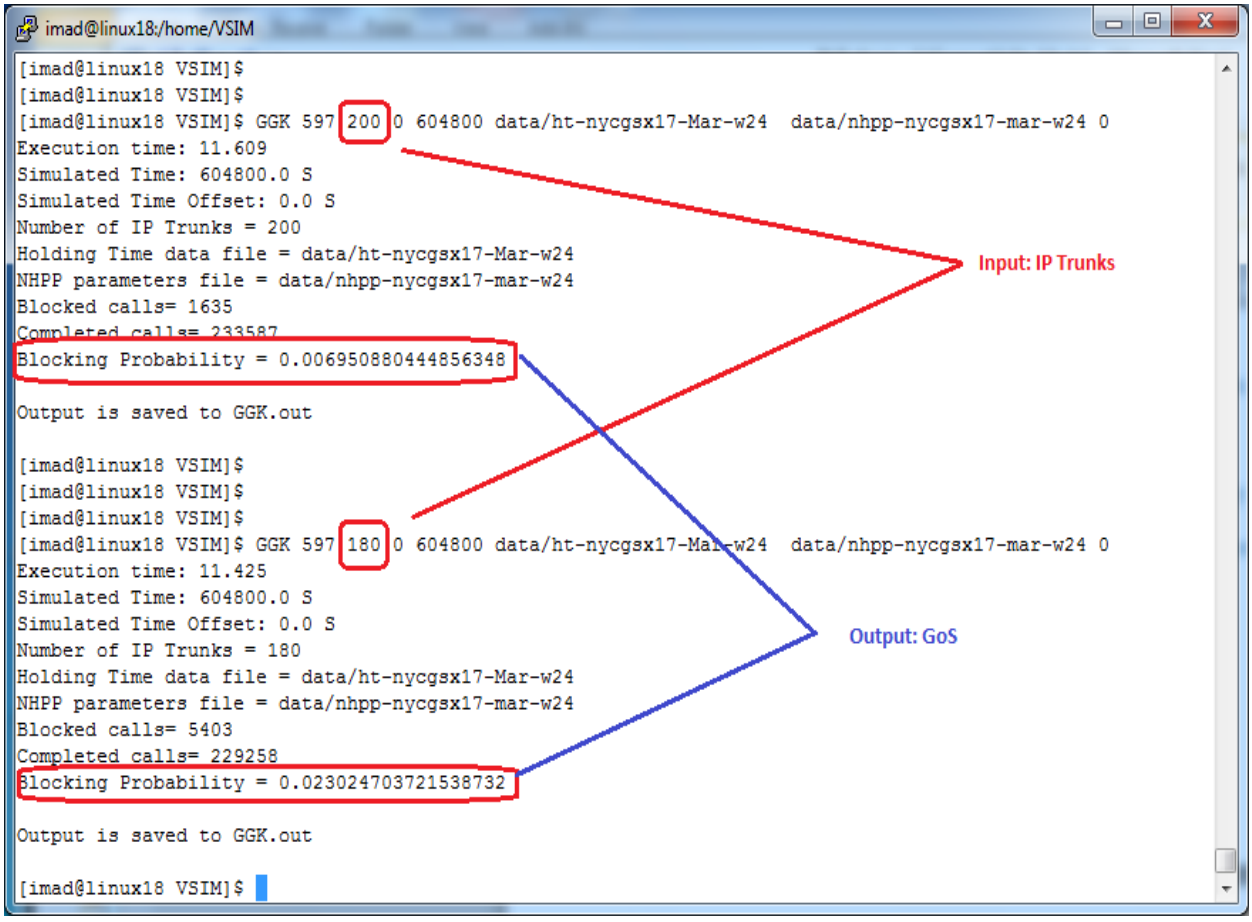

Figure 40. VSIM output: effect of the number of IP trunks on GoS

b) Resource requirements (IP trunk size): In this case we have a target GoS and a given traffic pattern and we need the estimate the number of IP trunks needed to achieve this

GoS. VSIM is used in an iterative manner to compute the required number of IP trunk groups.

VSIM is a flexible Java-based tool developed using CSIM for Java library [4], and therefore it can be easily ported to different computing platforms. VSIM can be used to estimate the trunk group size, generate GoS reports, and perform what-if analysis for VoIP networks.

VSIM is composed of two different simulation engines; the first is a parametric G/G/c/c simulator and the second is a non-parametric simulator. Both are discrete event simulators in which the VoIP system is modeled as chronological sequence of call arrivals and terminations. In the G/G/c/c engine we model the call arrival rate using the time-dependent function shown in (1), model parameters are estimated based on the collected data sample. This call arrival function is used to generate random variables for call inter-arrival times. Once a call is generated, the simulation code polls a random call holding time from a list of real holding times. The simulation engine allocates a trunk for the duration of simulated call holding time. A separate thread is created for each call so that we can collect statistics for each individual call and trunk. Once the simulated call time is over, the completed calls counter is incremented by one and the trunk will be released back to the trunk pool. The same procedure is repeated for the next calls until the pool of trunks is depleted. Once all trunks are busy we increment the blocked calls counter for each call that arrives while no trunks are available. Figure 41 illustrates the internal VSIM simulation algorithm.

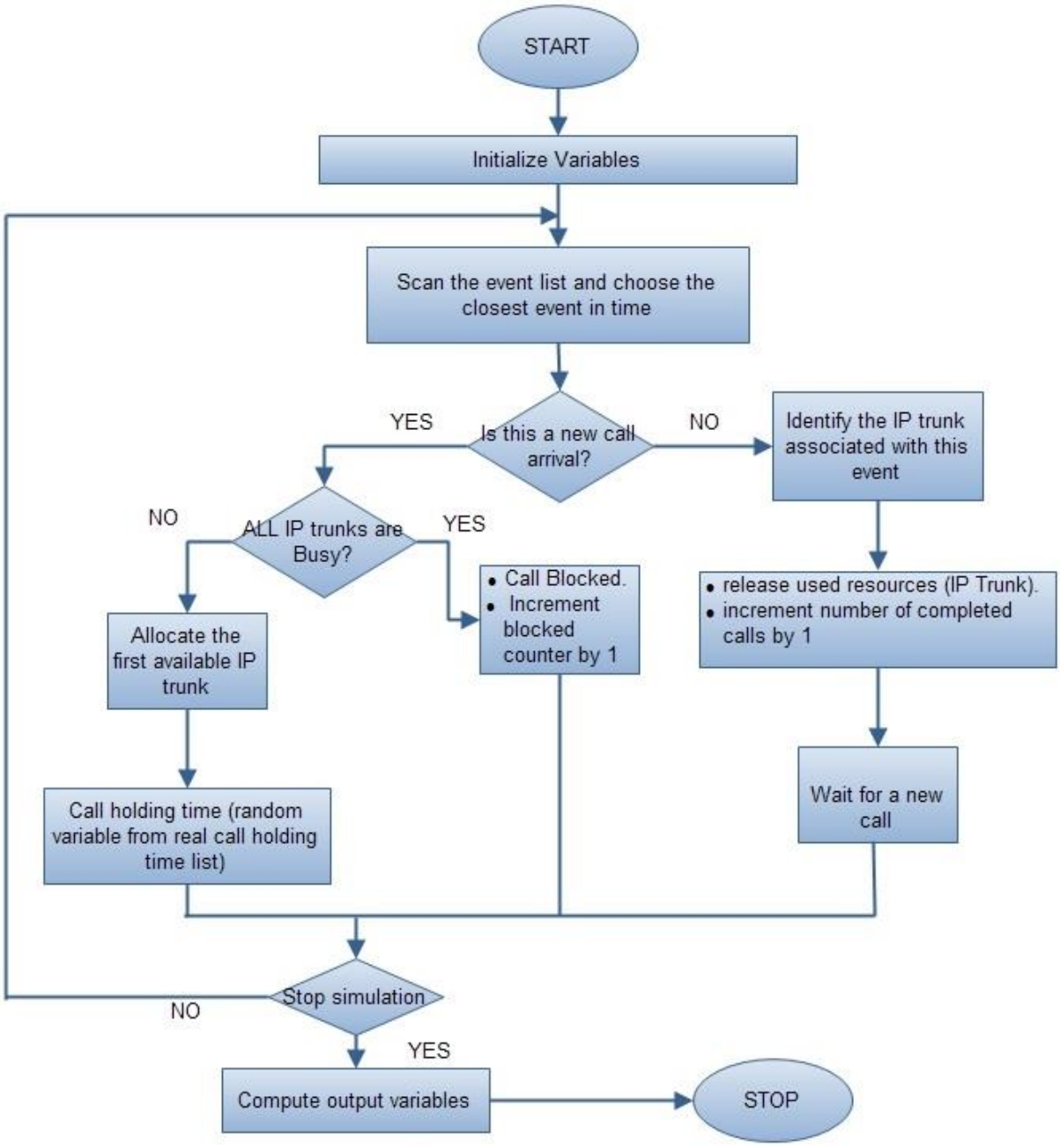

Figure 41. VSIM simulation algorithm

The non-parametric simulator follows the same algorithm with the exception that we poll the inter-arrival time variable from a real data file rather than using a NHPP function to generate it.

7.3.1 Parametric VSIM G/G/c/c simulator

Parametric simulation is done by collecting traffic data and then developing statistical models that best approximate the collected data. Model parameters are estimated based on the data sample and then these parameters are used in the simulation. We developed a G/G/c/c simulation model for VoIP traffic engineering. The model consists of a loss multi-server queuing system with waiting queue length equal to zero (blocked calls are cleared from the system). The implementation of general call arrival rate and general call holding time in the simulator allows for arbitrary distributions and that increases the flexibility and usability of our simulation model. The examples given in this thesis focus on modeling call arrival rate as NHPP using a generalized linear model that captures the variability in call arrival rate with respect to time. NHPP model parameters are estimated based on the real traffic data extracted from the production VoIP network under study.

7.3.2 Non-parametric VSIM simulator

In addition to the parametric G/G/c/c simulation engine implemented in VSIM, we also provide another non-parametric simulation engine. Non-parametric simulation is achieved throughout replaying the real traffic data without generating statistical models or estimating parameters. In other words, we use actual observations in the simulation rather than generating random variables from a statistical distribution. Therefore, non-parametric simulation would yield more accurate results since no approximation or data fitting are involved. This type of simulation is preferred when we have large amount of data. Another advantage of non-

parametric simulation is that developing statistical distributions is not necessary and hence the simulation process becomes easier.

We use the non-parametric simulator in this paper in order to verify the correctness of our G/G/c/c model and to validate its results. The input of our non-parametric simulator is real call information that consists of call inter-arrival time and call holding time. The simulator will regenerate the calls based on the given data, and we can study the system and compute the required resources and GoS.

7.4 VSIM model verification

It is important to verify the correctness of any simulation model before applying it to real-life problems. Simulation verification should cover the simulation engine algorithms as well as the random variables generated from the simulator statistical models. Therefore, we split VSIM model verification into two steps; the first is discussed in section 7.4.1 and aims to verify the correctness of the NHPP random variables generated and used by the simulator. The second step is discussed in section 7.4.2 and aims to verify the correctness of the simulation algorithms, timers and procedures.

7.4.1 Internal simulation random variables

We instrumented VSIM and obtained the call arrival rate generated by the model based on the implemented NHPP linear model. This call rate is used as an internal input to the G/G/c/c

simulation algorithm. Figure 42 shows the internally generated NHPP call rate variable along with the actual traffic data.

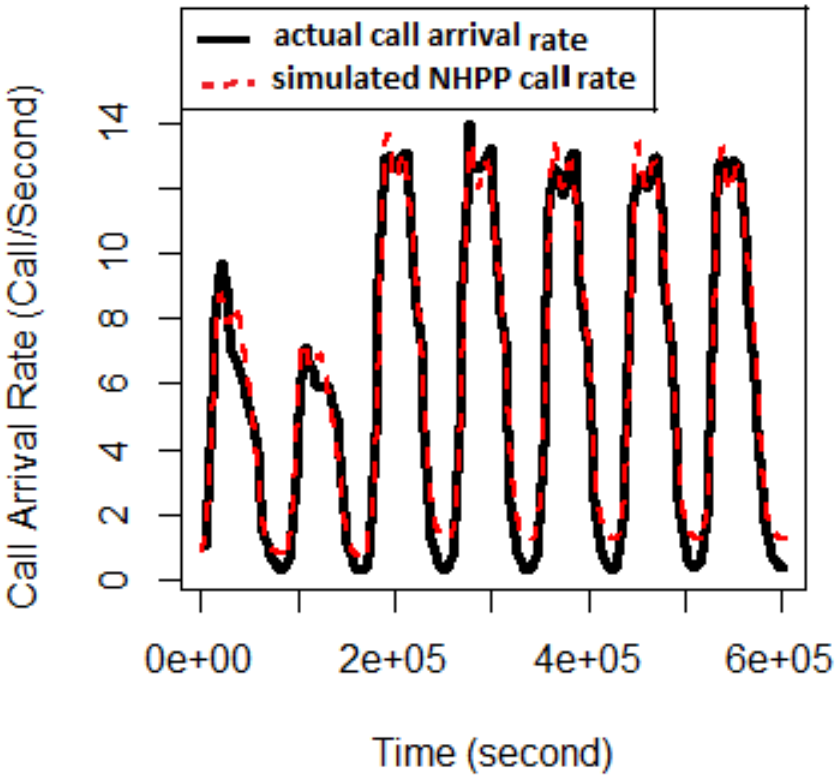

Figure 42. Simulated call arrival rate

As illustrated in Figure 42, the results of this verification process indicate that the NHPP model function used to generate call arrival random variables is correct and accurate. For the non-paramedic simulation engine, we don't need this verification graph since we are not using any model to generate random variable for the simulation; instead, the actual data is passed to the simulation process.

7.4.2 VSIM simulation engine algorithms

A common approach to verify the correctness of a new simulation model is by comparing it to other well-established simulators. Unfortunately, we could not obtain any G/G/c/c model against which we could verify our work; therefore, we used the exponential M/M/c/c special case of our model and compared the simulated result to the calculated results based on the Erlang B model. Our M/M/c/c utilizes the same simulation algorithms as our G/G/c/c and the only difference is that we use exponential distributions for both call arrival rate and call holding time. The goal is to verify the correctness of our simulation clock, event handling and algorithms. The validation of simulation results will be discussed in the next section against real traffic data.

In this process we used different data samples each one consists of one week of traffic; an example of the results is shown below:

Mean call holding time = 185 second

Busy Hour Traffic (BHT) = 12.6 call/ second

Using the Erlang B calculator we need around 2520 trunks in order to carry this traffic without blocking (Blocking probability nearing zero).

Using the M/M/c/c simulator with the same traffic parameters we found the required number of trunks to be 2515 for the same blocking probability. We ran multiple simulation runs for different data sets and comparable results were obtained for all the samples under test. These results verify the correctness of our VSIM simulation code and algorithms. In another example, we used VSIM to study the effect of available resources on GoS for a given traffic load, then we

computed the same GoS using Erlang-B calculator, the results are shown in Figure 43 and the match between VSIM simulated results and the calculated results is very high.

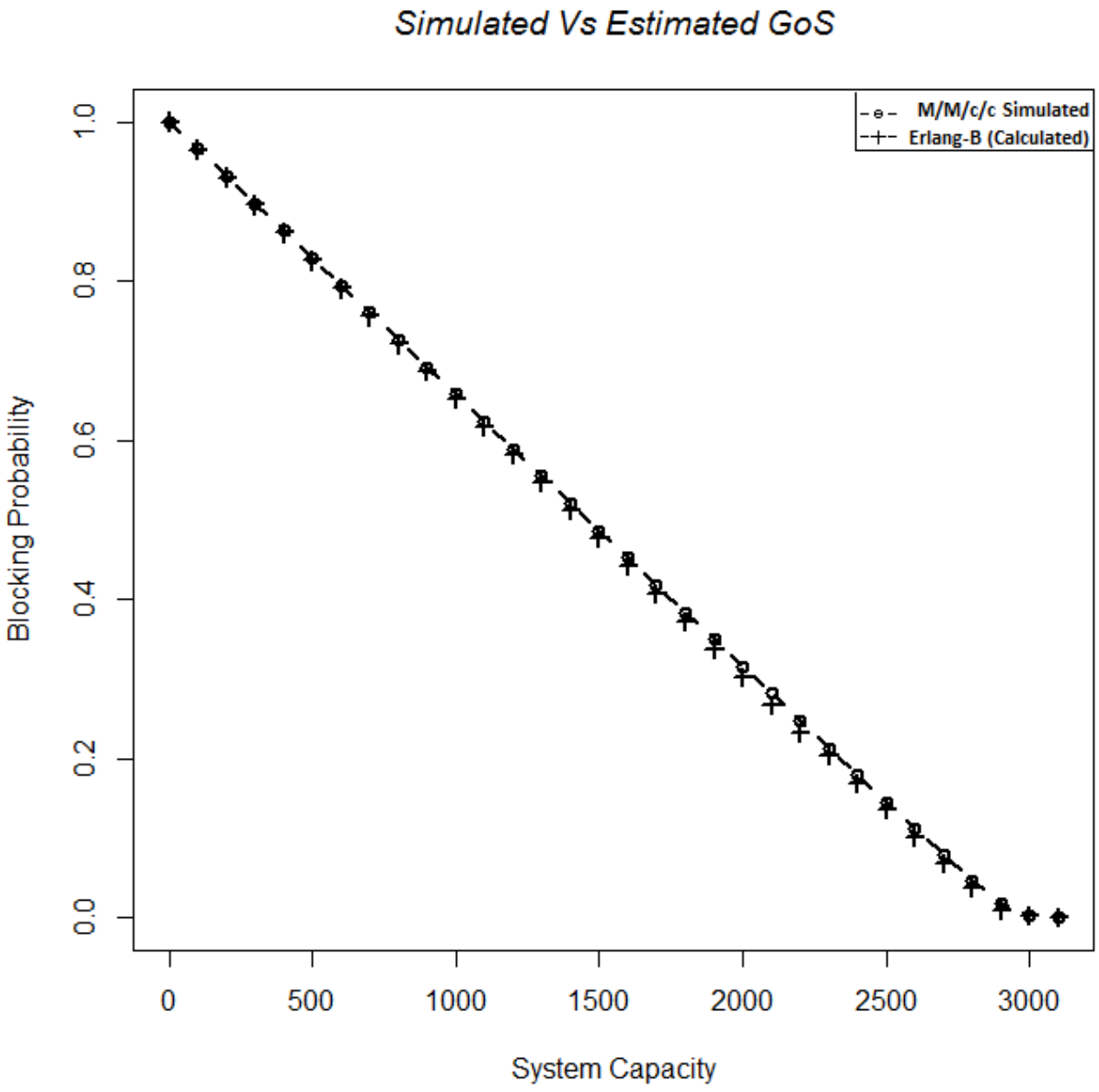

Figure 43. Simulated Vs Estimated GoS (Exponential special case)

7.5 VSIM Simulation Model Validation

The most important aspect of any simulator is that it should produce valid, correct and dependable results. The best approach to establish the validity of a simulator results is by comparison to real data. Therefore, we obtained the real resource utilization (number of simultaneous calls) from the production system for the period of time corresponding to the call arrival and holding time data used to develop the models. We used this data to validate VSIM simulation results. Different traffic samples and different simulation runs were used and all results agree with data obtained from the real network. In addition, we also used our nonparametric VSIM simulator to replay the same data samples and the results agree with those obtained from the system and those obtained from G/G/c/c simulation.

Table 12 shows an example of the actual used trunks obtained from the system compared to VSIM simulated output at GoS nearing zero. Figure 44 shows a comparison of VSIM results against actual system resources for the whole week at 10-minute intervals.

Table 12. Simulated Vs actual IP trunk requirements

	Maximum Call load (Pr[B] ≈ 0)
Actual (Observed)	1807
G/G/c/c (simulated)	1936
Non-Parametric (Simulated)	1810

It can be seen from Table 12 that both simulation models yield satisfactory results although the non-parametric model is a little better. The reason is because we don't have any modeling or estimation approximations for the non-parametric case. We used multiple data samples and executed multiple simulation runs and all the results are similar and indicate high accuracy of our VSIM for both G/G/c/c and non-parametric while the latter shows a little better results.

VSIM G/G/c/c simulator is based on using a function of time to model call arrival rate, and therefore VSIM can provide the resource requirements as a function of time as well. This function is important for system design, analysis and requirement studies, especially for converged networks where voice and data ride the same IP infra-structure. This function is also available for VSIM non-parametric simulator because we have real call information that depends on the time. Figure 44 illustrates sample resource functions (number of required IP trunks Vs. Time) generated by VSIM along with the corresponding simultaneous calls observed in the actual system at 10-minute intervals (real data validation). The figure shows the effectiveness of VSIM as demonstrated by its ability to compute the required system resources (IP trunks) as a function of time accurately. The resource time function provided by VSIM can be utilized for dynamic resource allocation scheme in which resources are allocated for different applications based on the actual or expected demand. Such scheme helps achieve better resource utilization and hence better engineering and cost reduction.

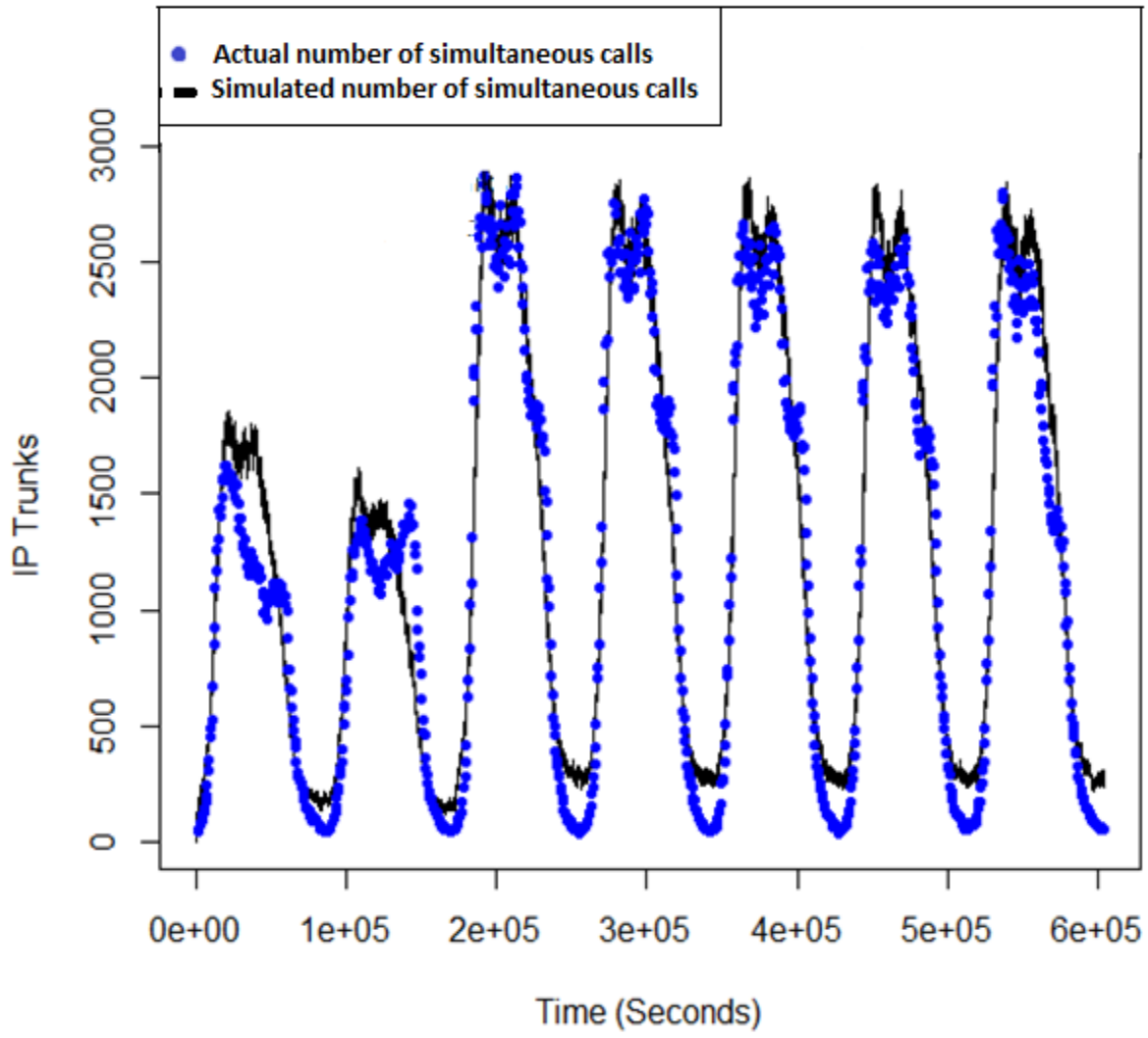

Figure 44. Number of IP trunks as a function of time (resource time function)

7.6 Simulation results and Analysis

It is important to notice that Erlang B and M/M/c/c models suggest a linear relation between blocking probability and system capacity (maximum number of simultaneous calls). However, our VSIM G/G/c/c and non-parametric simulators suggest a non-linear relation as seen in Figure 45. The figure shows identical match between the results obtained from the calculated Erlang B

and the simulated M/M/c/c. In order to use Erlang B and M/M/c/c we need to compute the average call arrival rate of the busiest hour and the average call holding time. For the results shown in Figure 45 and based on the data sample, we used 10.6 call/second for the arrival rate and 183.01 seconds for call holding time Also the figure shows close match between the G/G/c/c and non-parametric simulator. These results verify the correctness and validity of our procedure and modeling process. The deviation between the straight line calculated by the traditional Erlang B model and the curve generated by VSIM is significant and can affect the design and engineering decisions for the system. For example, if we want to build a switching system with 1600 maximum simultaneous calls (system capacity), the Erlang B approach suggests that the blocking probability will be 0.19 (P.19) while the VSIM G/G/c/c model results in a blocking probability of 0.02 (P.02). The difference between these two approaches is significant in the telecom world. VSIM nonparametric approach for the same data sample results in a blocking probability of 0.006 (P.006). Using the same example, we found that in order to achieve blocking probability of 0.01 (P.01), we will provision 1665 IP trunks using VSIM G/G/c/c model, or provision 1550 IP trunks using the VSIM non-parametric simulator. On the other hand we will provision 1991 trunks if we engineer the system using the Erlang B model. Therefore, we can see that using the VSIM model can save 28% of the resources over Erlang B at the P.01 blocking probability. Furthermore, Figure 45 shows that we can achieve better than 28% resource saving if higher blocking probabilities is desired.

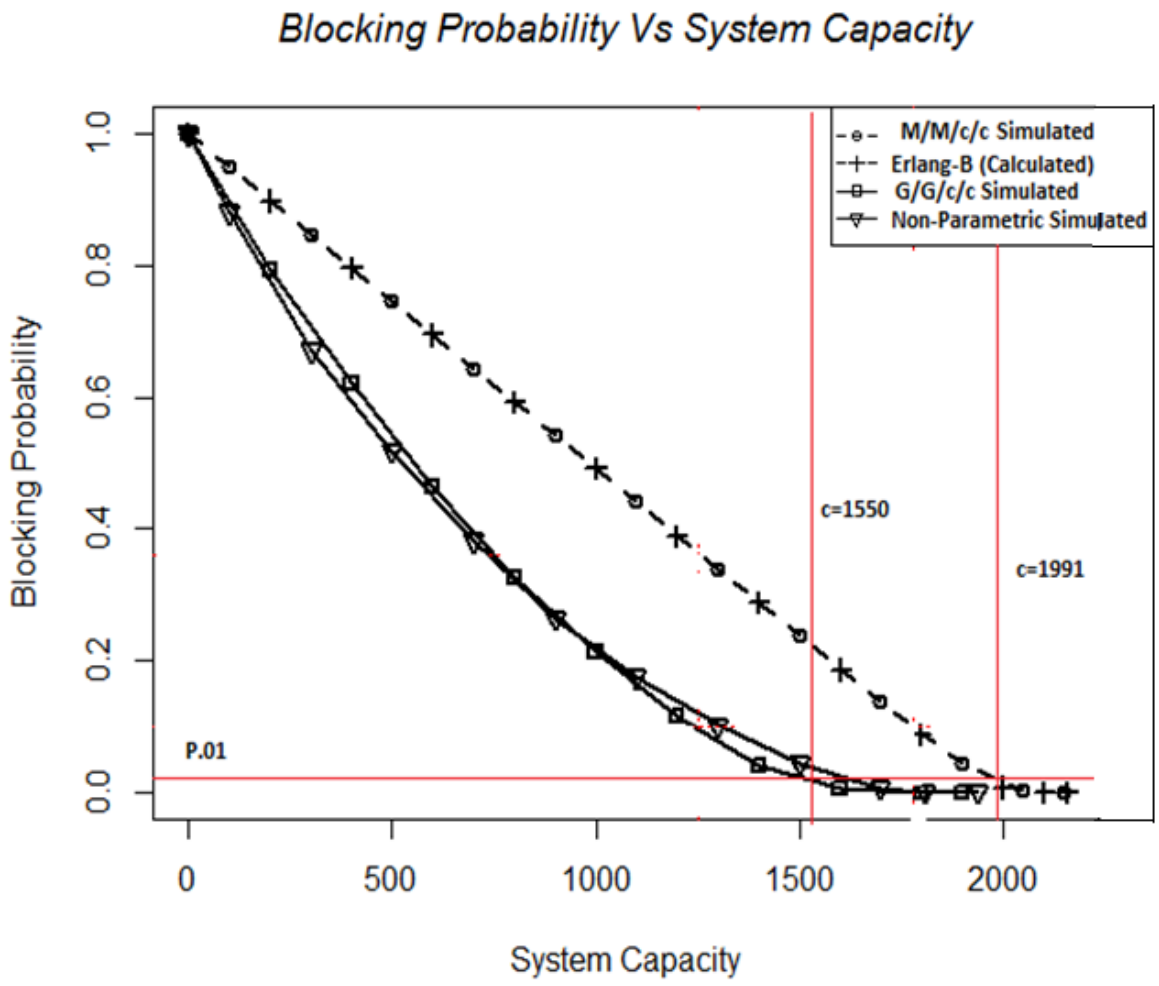

Figure 45. Blocking probability Vs system capacity (switch A)

We conducted many simulation runs using different data samples collected from different switch offices located in different cities. Similar results were obtained throughout this study. Figure 46 shows another example where the data is collected from a different office with more traffic load. The graph shows almost identical relation between the blocking probability and system resources. For the results shown in this figure and based on the data sample, we used 14.0125 call/second as the arrival rate and 204.186 seconds as call holding time for Erlang-B calculations

and M/M/c/c simulations. The graph indicates that we need 2500 IP trunks in order to carry the offered traffic load at blocking probability of 1%. On the other hand using the Erlang B model we will need 2885 IP trunks in order to achieve the same GoS. This sample shows that VSIM model could yield a saving of 15% of network resources.

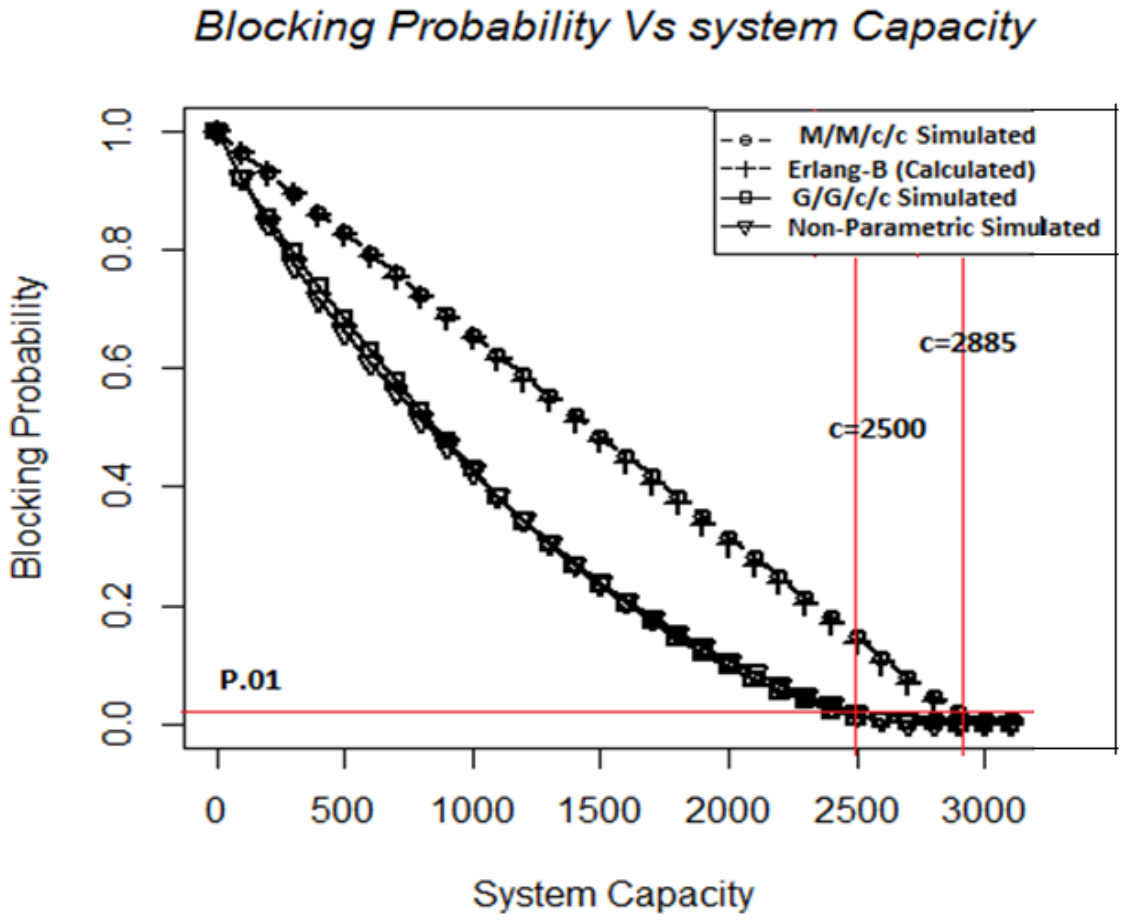

Figure 46. Blocking probability Vs system capacity (switch B)

Figure 47 shows a third example for a switch with larger capacity. The example shows that in order to achieve 1% blocking probability we need 4000 IP trunks if we design the system using VSIM simulation. On the other hand we will need 4660 if we design the system using Erlang-B

model. Therefore, for this traffic sample using VSIM can save 16.5% of the resources. In this example Erlang-B calculation is based on call arrival rate of 23.678 call/second and 196.33 s call holding time, the total number of calls generated during the week is 6.15 million calls.

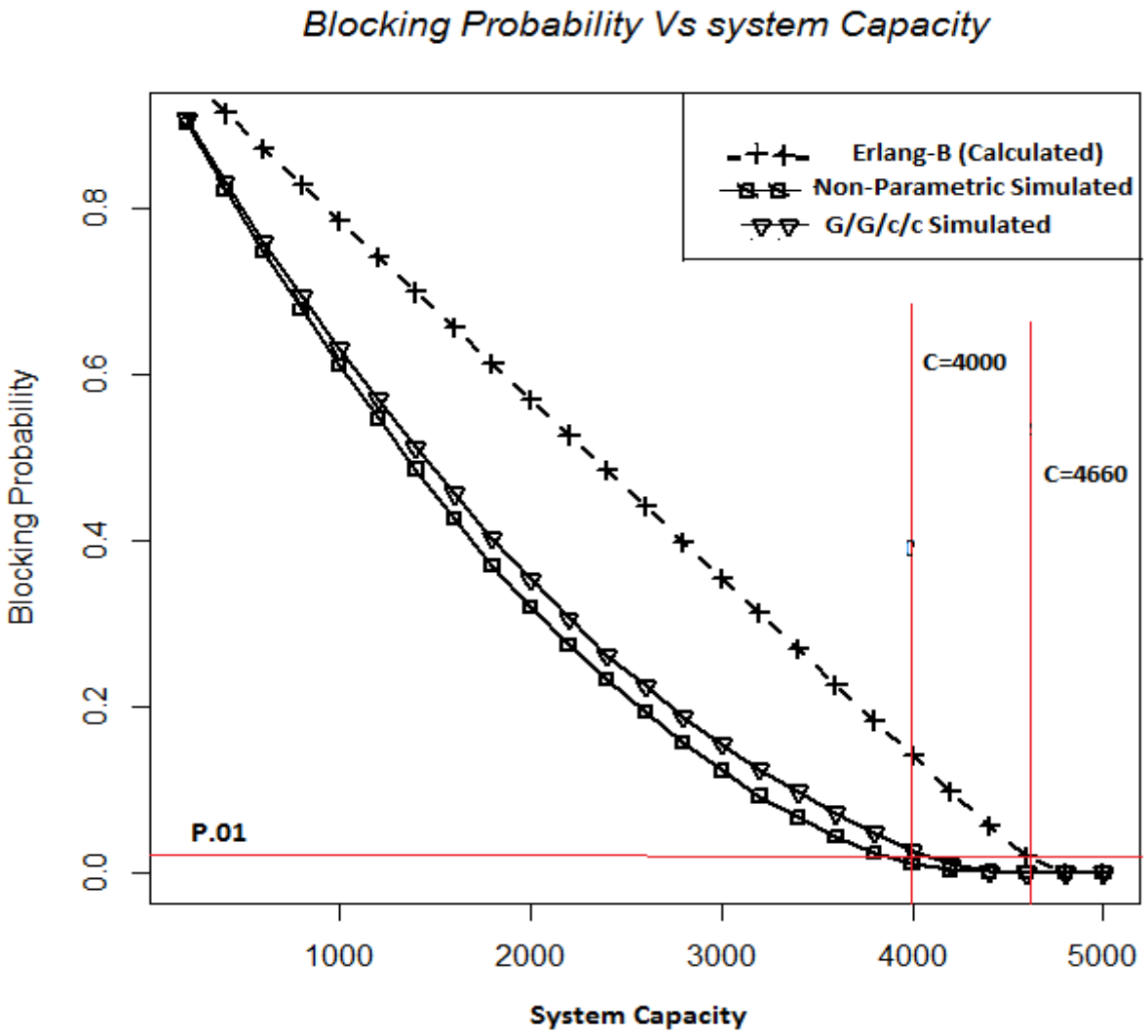

Figure 47. Blocking probability Vs system capacity (switch C)

It takes VSIM simulation engine about 5 minutes in order to run a simulation time of a complete week for the switch shown in Figure 46 (3.62 million calls per week). Similarly, VSIM needs 13

minutes to simulate a week worth of traffic for the example shown in Figure 47 (6.15 million calls per week). The performance of the G/G/c/c and the non-Parametric is comparable. The reasons for this performance are:

- We are running VSIM on a simple lab machine that has a single CPU and only 4G of memory.
- We are using Java for building VSIM. Java is not the best programming language for performance; however, CSIM API is available only for Java under the Demo license.

7.7 Summary

The advantage of using complex statistical models to capture traffic patterns is that these models provide accurate representation of the actual data. On the other hand the disadvantage of using complex models such as NHPP is that an analytical solution is not feasible. The availability of powerful computer systems makes the simulation approach feasible and effective, and hence we can obtain accurate results. We provided two simulators: the first is based on NHPP call arrival rate and the second is based on non-parametric data. VSIM is capable of solving the traffic modeling problem for modern VoIP systems accurately using arbitrary and complex traffic models or by using the raw traffic information without estimation of parameters. Our results are validated against real data collected from multiple offices of a production VoIP carrier network. We observed that the non-parametric simulator results are more accurate. Real traffic data proves that using VSIM could save from 15% up to 28% of the resources over the Erlang B model or other exponential-based models.

CHAPTER 8

8 Conclusions and Future Work

We provided an in-depth study for VoIP resource requirements, and we concluded that the traditional method of calculating resource requirements based on bandwidth alone is not enough. We proposed to use packet throughput in addition to bit throughput (bandwidth) in calculating VoIP network resource requirements. Also we introduced a new metric (Max Call Load) for VoIP network capacity. Call admission Control (CAC) systems can adopt the Maximum Call Load Metric and decide whether to accept or reject a call. This approach allows for using traffic engineering models in order to engineer VoIP network in a manner similar to PSTN networks.

The legacy traffic engineering models such as Erlang B are based on the assumption that calls arrive according to a Poisson distribution with fixed rate, and last for a call holding time that follows a negative exponential distribution. These assumptions make it easy to find analytical solutions for the traffic problem; however they include huge amount of approximation errors especially for modern VoIP systems. Instead of using Poisson process with fixed rate, we proved that using a Non-Homogeneous Poisson Process (NHPP) with call rate that varies as a

function of time provides more accurate representation for the call arrival process. We based our work on hundreds of millions of call data collected from a production VoIP carrier network over 3 years. The data shows that our NHPP approach provides strong model validity and goodness-of-fit. This model could support network management systems to develop a dynamic resource allocation procedure that helps optimize converged networks. During the peak time of voice traffic, more network resources are allocated to the voice application. When voice traffic is low, more network resources are allocated for data services.

We introduced the normality (Gaussian) approximation of call arrival rate for networks that operates under heavy traffic loads. We based this conclusion on the fact that Poisson Process behaves like a Gaussian process when its expected value is large (heavy traffic condition) as is the case in the tandem network under study. The accuracy of this approximation is based on Berry-Esseen Theorem which puts a bound on the discrepancy between certain distributions and the Gaussian distribution. We conclude that the normal distribution can provide an intuitive and accurate representation for call arrival rate on large-scale networks. The Gaussian approximation allows finding explicit mathematical equations for the model parameters and also provides easy model validation and significance testing. For a sample of data under study, we achieved a coefficient of determination, R^2, of 0.9973 which means that 99.73 of the variations in the empirical data are captured by the proposed Gaussian model.

We concluded that the exponential approximation of call holding time implemented by Erlang B model falls short from capturing the variation in call durations for modern VoIP systems. Therefore, we introduced a new methodology to modeling call holding time using time-to-event analysis. We introduced Call Cease Rate Function and found that both the log-logistic and the generalized gamma distributions provide a good fit for the data. Our statistical analysis

shows that the approach of modeling call cease rate provides more accurate results than the traditional exponential and log-normal holding time models.

The available Erlang B model has been used for traffic engineering for several decades. This model was developed long time ago where powerful computers did not exist; therefore, the Erlang model opted to do a lot of approximations in order to fit the traffic in simple exponential models so that an easy analytical solution could be obtained. The empirical data proved that this model is not suitable for modern traffic patterns. The availability of powerful low-cost computers encouraged us to opt for more complex models that can accurately capture the traffic. The complex models provide accurate data representation but cannot yield to the derivation of a simple analytical solution. Therefore, we opted to build a simulator that implements these complex models. We introduced VSIM as a VoIP traffic engineering simulator. We performed large number of simulation runs and validated the results against data obtained from a VoIP carrier network, and we conclude that VSIM can provide better results and can help to better optimize the network resource utilization. In addition, we built a second non-parametric simulation engine that can be used to perform traffic engineering studies using the empirical data without building any models. The results of the non-parametric simulation are more accurate compared to the parametric simulation and the reason is that we eliminate any modeling or estimation errors in the case of non-parametric simulation.

Our future work plan will focus on:

- Implementing the generalized gamma distribution for call holding time in VSIM since currently VSIM focuses on implementing NHPP for call arrival rate and we use the empirical data for the service time.
- Adding multiservice concept to VSIM. Currently, VSIM considered one class of service only, and we plan to extend its capabilities to include multiple classes of services, each with different resource requirements and traffic characteristics.
- Building a new dynamic resource allocation scheme that depends on VSIM output and that can be integrated with live networks and change the resource allocation.
- The increasing popularity of Internet-based multimedia applications demands finding more efficient mechanisms for controlling and managing this kind of time-sensitive traffic. The models and approach employed in this research can be easily extended to support multimedia traffic engineering. Therefore, we are planning to expand the models and simulations provided in this study to cover the multimedia traffic over the Internet.
- We understand Java performance limitations, and we are planning to rewrite VSIM using C or C++.

REFERENCES

[1] http://en.wikipedia.org/wiki/Voip

[2] http://www.researchandmarkets.com/reportinfo.asp?report_id=1202379&t=d&cat_id=

[3] http://www.in-stat.com/press.asp?ID=2761&sku=IN1004832CT

[4] http://en.wikipedia.org/wiki/Teletraffic_engineering

[5] One way Transmission time, ITU-T Recommendation G.114, May 2003

[6] A. Markopoulou, F. Tobagi, and M. Karam, "**Assessing the Quality of Voice Communications over Internet Backbones,**" IEEE/ACM Transactions on Networking, Vol.11, Issue 5, October 2003, pp.747-760

[7] Cisco, "**VoIP Call Admission Control**" http://www.cisco.com/univercd/cc/td/doc/cisintwk/intsolns/voipsol/cac.htm

[8] Solange R. Lima, Paulo Carvalho, and Vasco Freitas. "**Admission Control in Multiservice IP Networks: Architectural Issues and Trend**," IEEE Communications, Vol. 45 No. 4, April 2007, 114-12

[9] Shenquan Wang, et. al. "**Design and Implementation of QoS Provisioning System for Voice over IP**," IEEE Transactions on Parallel and Distributed Systems, Vol 17 No. 3, March 2006

[10] Xiuzhong Chen, et. al. "**Survey on QoS Management of VoIP**," International Conference on Computer Networks and Mobile Computing, IEEE 20-23 October 2003, P.68-77

[11] Sueng Jae Bae, Jin Ju Lee, Bum-Gon Choi, Sungoh Kwon, Min Young Chung: "**A Resource-Estimated Call Admission Control Algorithm in 3GPP LTE System,**" ICCSA (2) 2009: 250-260

[12] Zhang Dandan, Fang Xuming, and Zhu Longjie , "**Novel multimedia traffic modeling based CAC scheme for CDMA communication systems,"** Journal of Electronics(China) 2007 24(1) : 39-45 ISSN: 0217-9822 CN: 11-2003/TN

[13] Jarosław Bylina, Beata Bylina, Andrzej Zoła, Tomasz Skaraczyński, "**A Markovian Model of a Call Center with Time Varying Arrival Rate and Skill Based Routing,**" Computer Networks: 16th Conference, CN 2009, Wisla, Poland, June 2009

[14] R. Baloch, I. Awan, and G. Min, "**A mathematical model for wireless channel allocation and handoff schemes,**" Telecommunication Systems. [Online]. Volume 45, Number 4, 275-287, DOI: 10.1007/s11235-009-9267-5 (2010)

[15] U. Narayan Bhat, "**An Introduction to Queuing Theory**," Boston: Birkhauser, 2008 pp. 229-231

[16] Y. Fang, "**Modeling and performance analysis for wireless mobile networks: a new analytical approach,**" IEEE/ACM Transactions on Networking, Vol.13, No.5, pp. 989-1002, October 2005.

[17] A.K. Erlang, "**The Application of The Theory of Probabilities in Telephone Administration,**" Scandinavian H.C. Orsted Congress in Copenhagen, 1920

[18] Thomas Bonald, "**The Erlang Model with non-Poisson Call Arrivals,**" Proceedings of ACM/Sigmetric and Performance Conference, pp. 276–286Saint Malo, France, 2006.

[19] G. Hess and J. Cohn, "**Communications load and delay in mobile trunked systems**," Proc. IEEE Vehicular Technology Conf., pp. 269–273, 1981.

[20] Zvezdan Stojanovic and Dorde Babic "**Bandwidth Calculation for VoIP Networks Based on PSTN Statistical Model,**" FACTA UNIVERSITATIS (NIS), SER.: Electrical Engineering. vol. 23, no. 1, pp 73-88, Apr. 2010

[21] Irina Buzyukova, Yulia Gaidamaka, Gennady G. Yanovsky , "**Estimation of GoS Parameters in Intelligent Network,**" 9th International Conference, NEW2AN 2009 and Second Conference on Smart Spaces, ruSMART 2009, St. Petersburg, Russia, September 2009

[22] Duffy DE, McIntosh A, Rosenstein M, Willinger W. "**Statistical analysis of CCSN/SS7 traffic data from working CCS subnetworks,**" IEEE Journal on Selected Areas in Communications 1994; 12: no. 3, April.

[23] L. G. Afanas'eva and E. E. Bashtova , "**Limit theorems for queueing systems with doubly stochastic poisson arrivals (Heavy traffic conditions),**" Problems of Information Transmission, Vol. 44, No. 4. pp. 352-369, December 2008

[24] Bashtova, E.E., "**The Small Workload Mode for a Queueing System with Random Unsteady Intensity,**" Mat. Zametki, 2006, vol. 80, no. 3, pp. 339–349 [Math. Notes (Engl. Transl.), vol. 80, no. 3–4,pp. 329–338], 2006

[25] F. Barcelo and S. Bueno, "**Idle and inter-arrival time statistics in public access mobile radio (PAMR) systems**," in Proc. IEEE Globecom, Phoenix, AZ, Nov. 1997.

[26] H.C.Tijms, "**Stochastic Modelling and Analysis,**" John Wiley & Sons, pp.330-401, 1986.

[27] Lawrence Brown, Noah Gans, Avishai Mandelbaum, Anat Sakov, Haipeng Shen, Sergey Zeltyn, and Linda Zhao, **"Statistical Analysis of a Telephone Call Center A Queueing-Science Perspective,**" Journal of the American Statistical Association, March 2005, Vol. 100, No. 469, Applications and Case Studies

[28] Duncan S. Sharp, Nikola Cackov, Nenad Laskovic, Qing Shao, and Ljiljana Trajkovic, "**Analysis of Public Safety Traffic on Trunked Land Mobile Radio Systems,**" IEEE Journal on selected areas in Communications, VOL. 22, NO. 7, September 2004

[29] José Ignacio Sánchez, Francisco Barceló and Javier Jordán, "**Inter-arrival Time Distribution for Channel Arrivals in Cellular Telephony,**" Proceedings of 5 Intl. Workshop on Mobile Multimedia Communication MoMuc'98, October 12-14 1998, Berlin

[30] V. Bolotin., "**Telephone Circuit Holding Time Distributions,**" Proc. 14th ITC, Amsterdam: Elsevier Science B.V. pp. 125-134, 1994.

[31] E Chlebus, "**Empirical validation of call holding time distribution in cellular communications systems**," Teletraffic Contributions for the Information Age (Proc. 15th ITC), vol. 2b, Elsevier, Amsterdam, 1999, pp. 1179-1188

[32] Francisco Barcelo, Javier Jordan, "**Channel holding time distribution in cellular telephony,**" Electronics Letters, vol. 34 no. 2, pp. 146-147, 1998.

[33] C. Jedrzycky, V. Leung, "**Probability distribution of channel holding time in cellular telephony system,**" Proc. IEEE Veh. Technol. Conf., Atlanta, GA, May 1996

[34] Francisco Barceló and Javier Jordán ,"**Channel Holding Time Distribution in Public Cellular Telephony,**" Proc. 16th ITC, Amsterdam: Elsevier Science B.V pp. 102-116, 1999

[35] Ramaswami, V., D. Poole, S. Ahn, S. Byers, A. E. Kaplan. "**Containing the effects of long holding time calls due to Internet dial-up connections,**" Proc. IEEE PACRIM Conf. Victoria, British Columbia, Canada, 2003

[36] Y. Fang, I. Chlamtac, and Y.-B. Lin, "**Modeling PCS Networks under General Call Holding Time and Cell Residence Time Distributions**," IEEE/ACM Trans. Networking, vol. 5, no. 6, pp. 893-906, 1998.

[37] Y. Fang, "**Modeling and performance analysis for wireless mobile networks: a new analytical approach**," IEEE/ACM Transactions on Networking, Vol.13, No.5, pp. 989-1002, October 2005

[38] George Casella, Roger Berger, "**Statistical Inference,**" Duxbury Press; June 2001

[39] Dobson, Barnett, "**An Introduction to Generalized Linear Models**," Third Edition, Chapman & Hall/CRC Textx in statistical Science. 2008

[40] Deng X. and Yuan M, "**Large Gaussian covariance matrix estimation with Markov structures,**" Journal of Computational and Graphical Statistics, 18(3), 640-657, (2009)

[41] DasGupta, Anirban. **Asymptotic theory of statistics and probability**. Springer Texts in Statistics. Springer, New York, 2008

[42] Peccati, G.; Solé, J. L.; Taqqu, M. S.; Utzet, F. "**Stein's method and normal approximation of Poisson functionals,**" Annals of Probability, 38 (2010), no. 2, 443–478

[43] Ryan, Thomas P. "**Modern regression methods,"** Second edition. Wiley Series in Probability and Statistics, John Wiley & Sons, Inc., Hoboken, NJ, 2009.

[44] B. Baynat and P. Eisenmann, "**Toward s an Erlang- like formula for GPRS /EDGE network engineering**," I n Proc. of I EEE ICC , Paris, France, June 2004

[45] Scott R. Eliason "**Maximum likelihood estimation: logic and practice**," Volume 96, Sage Publications 1993, pp. 7-10

[46] Adriaan van den Bos "**Parameter estimation for scientists and engineers**," Wiley-Interscience; first edition (July 16, 2007), pp 201

[47] Joop J. Hox, "**Multilevel analysis: techniques and applications**," Routeledge 2010. Pp. 46

[48] Adrian Colin Cameron "**Microeconometrics: methods and applications**," Camridge University Press 2005, pp 224-225

[49] William Gould, Jeffrey Pitblado, William Sribney, "**Maximum Likelihood Estimation with Stata**," Stata Press, 4 edition (October 27, 2010). Pp

[50] David G. Kleinbaum, Mitchel Klein, Erica Rihl Pryor "**Logistic Regression: A Self-Learning Text**," Springer Third Edition 2010, pp. 134

[51] Jianqing Fan, and Qiwei Yao, "**Nonlinear time series: nonparametric and parametric methods**," Springer Series in Statistics 2005, pp. 405-410

[52] John P. Klein, and Melvin L. Moeschberger, "**Survival analysis: techniques for censored and truncated data**," Springer, 2nd edition (February , 2003), pp. 21-22

[53] Antal Kozak, C. Staudhammer, and S. Watts, "**Introductory probability and statistics: Applications for Forestry and the Natural Sciences,**" CAB International 2008, pp 14-18

[54] Oliver Nelles, "**Nonlinear system identification**," Springer 2001, pp 104- 105

[55] C. T. Kelley, "**Iterative methods for optimization**," Society for Industrial and Applied Mathematics 1999, pp. 47 – 48

[56] Janet M. Box-Steffensmeier, and Bradford S. Jones, "**Event history modeling: a guide for social scientists**," Cambridge University Press 2004, pp. 120

[57] Paul Allison, "**Survival Analysis Using SAS: A Practical Guide,**" SAS Publishing, November 1995

[58] David Collett, "**Modeling Survival Data in Medical Research,**" Chapman and Hall/CRC; March 2003

[59] Jeff B. Cromwell, Walter C. Labys, and Michel Terraza, "**Univariate tests for time series models**," Sage Publications 1994, pp. 66

[60] By Xiaobo Zhou, Stephen T. C. Wong, "**Computational systems bioinformatics: methods and biomedical applications**," World Scientific Publishing 2008, pp. 132 – 134

[61] John R. Wolberg, "**Data analysis using the method of least squares: extracting the most Information from experiments**," Springers 2006, pp 31 – 34

[62] R. W. Farebrother, "**Linear least squares computations**," Marcel Dekker Inc 1988, pp 49 – 50

[63] Henry C. Thode, "**Testing for normality**," Eastern Hemisphere Distribution 2002, pp 3-10

[64] http://www.r-project.org/

[65] Mathew N. O. Sadiku and Mohammad R. T Ofight "**A Tutorial on simulation of Queueing**," Int. J. Elect. Enging. Educ., Vol. 36, pp. 102–120. Manchester U.P., 1999. Printed in Great Britain

[66] Solange R. Lima, Paulo Carvalho, and Vasco Freitas. "**Admission Control in Multiservice IP Networks: Architectural Issues and Trend**," IEEE Communications, Vol. 45 No. 4, April 2007, 114-12

[67] Bowei Xi, Hui Chen, Williams S. Cleveland, and Thomas Telkamp, "**Statistical analysis and modeling of Internet VoIP traffic for network engineering**," Electronic Journal of Statistics, Vol. 4, ISSN 1935-7524, pp. 58-116, 2010

[68] Michael D. Logothetis and Ioannis D. Moscholios, "**Call-level Multi-Rate Teletraffic Loss Models**," The Second International Conference on Internet Monitoring and Protection, Silicon Valley, USA 2007

[69] Ioannis D. Moscholios, and Michael D. Logothetis, "**The Erlang multirate loss model with Batched Poisson arrival processes under the bandwidth reservation policy**," Computer Communications Journal, Volume 33, November, 2010

[70] I. D. Moscholios, J. S. Vardakas, M. D. Logothetis, and A. C. Boucouvalas, "**A Batched Poisson Multirate Loss Model Supporting Elastic Traffic Under The Bandwidth Reservation Policy**," IEEE International Conference on Communications (ICC2011), Japan, June 2011

[71] Vassilios G. Vassilakis Georgios A. Kallos Ioannis D. Moscholios Michael D. Logothetis, "**The wireless Engest Multi-rate Loss Model for the Call-level analysis of W-CDMA Networks**," The 18th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'07), Zürich , Switzerland 2007

[72] Yuguang Fang, "**Hyper-Erlang Distribution Model and its Application in Wireless Mobile Networks**," Wireless Networks 7, pp 211–219, 2001

[73] S. Asmussen, O Nerman, and M. Olsson "**Fitting phase-type distributions via the EM algorithm**," Scandinavian Journal of Statistics, Vol. 23, pp 419-441, 1996

[74] Thomas Kaare Christensen, Bo Friis Nielsen, and Villy Bæk Iversen, "**Phase-Type Models of Channel-Holding Times in Cellular Communication Systems**," IEEE Transaction on Vehicular Technology, VOL. 53, NO. 3, MAY 2004

[75] P. V. Orlik and S. S. Rappaport, "**A model for teletraffic performance and channel holding time characterization in wireless cellular communication with general session and dwell time distributions**," IEEE J. Select. Areas Commun., vol. 16, pp. 788–803, June 1998.

[76] S. Pattaramalai, V.A Aalo, and G.P Efthymoglou, "**Call Completion Probability with Weibull Distributed Call Holding Time and Cell Dwell Time,**" IEEE Global Telecommunications Conference, GLOBECOM '07, Washington DC, 2007

[77] R. Litjens and R.J. Boucherie, "**Elastic Calls in an Integrated Services Networks: the Greater the Call Size Variability the Better the QoS**," Performance Evaluation, vol. 52, pp. 193-220, May 2003

[78] Noah Gans, Ger Koole, and Avishai Mandelbaum , "**Telephone Call Centers: Tutorial, Review, and Research Prospects**," Manufacturing ans Service Operations Management, Vol. 4, pp. 208-227, 2003

[79] Anum L. Enlil Corral-Ruiz, Andrés Rico-Páez, Felipe A. Cruz-Pérez, and Genaro Hernández-Valdez, "**On the Functional Relationship between Channel Holding Time and Cell Dwell Time in Mobile Cellular Networks**," IEEE Globecom 2010, Miami, Florida 2010.

[80] Corral-Ruiz, A.L.E., Cruz-Perez, F.A., and Hernandez-Valdez, G., **"Channel holding time in mobile cellular networks with heavy-tailed distributed cell dwell time,"** IEEE Wireless Communications and Networking Conference (WCNC), Cancun, Mexico 2011

[81] Marco Ajmone Marsan, Gabriele Ginella, Roberta Maglione, and Michela Meo, "**Performance Analysis of Hierarchical Cellular Networks With Generally Distributed Call Holding Times and Dwell Times**," IEEE Transaction on Wireless Communications, VOL. 3, NO. 1, January 2004

[82] A. Ariza, E. Casilari, and F. Sandoval , "**Impact of Call Holding Time Distribution on QoS Routing**," IASTED International Conference on Advances In Communication (AIC), Rhodes (Grecia) , 2001

[83] Ramon M. Rodriguez-Dagnino and Hideaki Takagi, "**Wireless cellular networks with Pareto-distributed call holding times,**" Proc. SPIE 4523, 202 (2001); doi:10.1117/12.434315

[84] Imad Al Ajarmeh, James Yu and Mohamed Amezziane, "**Framework of Applying a Non-Homogeneous Poisson Process to Model VoIP Traffic on Tandem Networks**," 10th WSEAS International Conference on Informatics and Communications, Taipei, Taiwan, August 2010

[85] Engle, R. "**Autoregressive conditional heteroscedasticity with estimates of the variance of UK inflation**," Econometrica, 50 (1982), 987–1008

[86] McLeod, A. I. and W. K. Li. "**Diagnostic checking ARMA time series models using squared residual autocorrelations**," Journal of Time Series Analysis 4, pp. 269-273, 1983

[87] Engle, R. "**Risk and volatility: Econmetric models and financial practice. Nobel Lecture,** " www.nobelprize.org/nobel_prizes/economics/laureates/2003/engle-lecture.pdf

[88] Tsay, R. "**Analysis of financial time series**," Wiley, 2010, pp. 1-9

[89] Williams, C. K. I. (1998b) "**Prediction with Gaussian processes: from linear regression to linear prediction and beyond**," In Learning in Graphical Models, Jordan, M. I., editor, Kluwer Academic, pp. 599-621

[90] K. Salah and A. Alkhoraidly, "**An OPNET-based Simulation Approach for Deploying VoIP,**" International Journal of Network Management, Volume 16 Issue 3, May 2006, Pages 159-183

[91] Sangho Shin, Henning Schulzrinne, "**Experimental Measurement of the Capacity for VoIP Traffic in IEEE 802.11 WLANs**," INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE, May 2007

[92] Trang Dinh Dang, Balázs Sonkoly, and Sándor Molnár, "**ractal Analysis and Modeling of VoIP Traffic**," 11th International Telecommunications Network Strategy and Planning Symposium, Austria 2004

[93] Martin J. Fischer, and Denise M. Bevilacqua Masi, "**Modeling Overloaded Voice over Internet Protocol Systems**," The Telecommunications Review 2006, pp. 94 - 106

[94] Nurul I. Sarkar, Syafnidar A. Halim, "**A Review of Simulation of Telecommunication Networks: Simulators, Classification, Comparison, Methodologies, and Recommendations**," Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications (JSAT), March Edition, 2011

[95] A. M. Law and W. D. Kelton, "**Simulation modeling and analysis**," third ed. New York: McGraw-Hill, 2000

[96] Onur Özgün, and Yaman Barlas, "**Discrete vs. Continuous Simulation: When Does It Matter?**," Proceedings of the 27th International Conference of The System Dynamics Society, July 26 – 30, 2009, Albuquerque, NM, USA

[97] G. Yuehong, Z. Xin, Y. Dacheng, and J. Yuming, "**Unified simulation evaluation for mobile broadband technologies**," IEEE Communications Magazine, vol. 47, Issue 3, pp. 142-149, March 2009

[98] V. Rakocevic, R. Stewart and R. Flynn, "**VoIP Network Dimensioning using Delay and Loss Bounds for Voice and Data Applications**," Technical Report, 2008.

[99] Howon Lee and Dong-Ho Cho, "**VoIP Capacity Analysis in Cognitive Radio System**," IEEE Communications Letters, VOL. 13, NO. 6, June 2009

[100] J.-W. So, "**Performance analysis of VoIP services in the IEEE 802.16e OFDMA system with inband signaling**," IEEE Trans. Veh. Technol., vol.57, no. 3, pp. 1876–1886, May 2008

[101] Mathias Bohge, and Martin Renwanz, "**A realisitic VoIP traffic generation and evaluation tool for OMNeT++,**" International Workshop on OMNeT++, Marseille, France, March 2008

[102] A. Bacioccola, C. Cicconetti, and G. Stea, "**User-level Performance Evaluation of VoIP Using ns-2**," NSTools'07, October 22, 2007, Nantes, France

[103] Das Gupta, Jishu and Howard, Srecko and Howard, Angela, "**Traffic behaviour of VoIP in a simulated access network,**" International Transactions on Engineering, Computing and Technology, 18 . pp. 189-194. ISSN 1305-5313, 2006

[104] Reyadh Shaker Naoum, and Mohanand Maswady, "**Performance Evaluation for VOIP over IP and MPLS**," World of Computer Science and Information Technology Journal (WCSIT) ISSN: 2221-0741 Vol. 2, No. 3, 110-114, 2012

[105] http://www.mesquite.com/documentation/documents/CSIM_for_Java-UserGuide.pdf [retrieved: Feb, 2013]