INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI

A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor MI 48106-1346 USA
313/761-4700 800/521-0600

Estimating the Future Performance of an ATM Network Based on its Current Performance

By

Sandra Stark

*A thesis submitted to
the Faculty of Graduate Studies and Research
in partial fulfillment of
the requirements for the degree of*

Master of Engineering

*Department of Systems and Computer Engineering
Carleton University
Ottawa, Ontario, Canada K1S 5B6*

National Library
of Canada

Acquisitions and
Bibliographic Services

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale
du Canada

Acquisitions et
services bibliographiques

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-22129-6

The undersigned recommend to
the Faculty of Graduate Studies and Research
acceptance of the thesis

"Estimating the Future Performance of an ATM Network
Based on its Current Performance"

submitted by Sandra Stark, B.Sc.
in partial fulfillment of the requirements for
the degree of Master of Engineering

I. Lambadaris - Chair

S.A. Mahmoud - Department

B. Pagurek - Thesis Supervisor

N.W. Dawes - Thesis Co-Supervisor

Abstract

The ability to estimate performance parameters and congestion rates in high speed networks is a feature that is very useful to network service providers. It lets them anticipate congestion problems before they occur, allowing them to avoid the problem. This thesis provides empirical evidence that supports a method that uses the past traffic characteristics of a network to predict its future performance. The data that is analyzed in this project comes from an existing telephone company ATM network. The Q value, which is the ratio of the mean traffic on a device to its standard deviation is used to classify the devices' traffic distribution. Evidence is presented that shows that on the studied ATM network, traffic distributions are relatively stable over time and that a devices' Q value can be used to characterize its traffic distribution. The relationship between the Q value and both the sampling interval and the sampling length is explored. This thesis also shows that for self similar data, the Q value can be estimated for a different sampling interval than the interval actually measured.

Problems associated with simulating a large, high speed ATM network are presented in this thesis. A method that takes a simulation generating a large number of cells per second and scales the simulation down to run in a reasonable time frame is outlined.

Acknowledgments

I would like to thank Professor Nick Dawes and Professor Bernie Pagurek for their supervision and guidance throughout the work of this thesis. The motivation and direction they provided was greatly appreciated.

I would also like to thank Tom Kanary, from Bell Sygma, for the time he invested in this project and for the ATM data that he provided.

This work was funded by the Natural Science and Engineering Research Council.

Contents

Abstract.....	i
Acknowledgments.....	ii
Contents	iii
List of Figures	v
List of Tables	viii
Glossary.....	ix
 Chapter 1: Introduction	1
1.1 Objectives.....	3
1.2 Assumptions.....	4
1.3 Overview.....	4
 Chapter 2: Traffic Characteristics and Statistical Background.....	6
2.1 Self Similar Traffic Characteristics	7
2.1.1 Aggregated Ethernet Traffic	10
2.1.2 Variable Bit Rate Video Traffic.....	11
2.1.3 ISDN Office Automation Traffic.....	12
2.2 Traditional Poisson Based Traffic Characteristics	12
2.3 Normalization	14
 Chapter 3: Simulating an ATM Network.....	15
3.1 Simulation Experiments.....	15
3.1.1 Model Development and Validation	15
3.2 Simulation Goals and Constraints	17
3.3 Network Description	18
3.4 Simulation Model.....	19
3.5 Limitations of Simulating Large, High Speed Networks	21
3.6 Distributed Simulation.....	23
3.7 Discussion	24

Chapter 4: Estimating Congestion Using the Q Factor.....	26
4.1 Data Description.....	26
4.2 Definition and Meaning of the Q Factor.....	27
4.3 Steps Necessary to Predict Congestion Using the Q Factor	29
4.4 Detemining that Distributions are Stable over Time.....	30
4.5 Predicting Traffic Distributions from Q Values	44
4.5.1 Bell ATM Network Distribution Observations	45
4.6 Relationship Between Q Factors and Sampling Intervals	53
4.6.1 Q Factors for Various Sampling Intervals on Bellcore Data	54
4.7 Relationship Between Q Factors and Sampling Length	55
4.7.1 Bellcore Data Results.....	55
4.7.2 Bell ATM Network Data Results.....	61
 Chapter 5: Combining Q Values at a Device.....	 68
5.1 Determing the Q Value for Any Network Device	68
5.2 Theoretical Q Combination Rule	70
5.3 Heuristical Q Combination Rules	72
5.4 Results of Combining Q Values on Bell ATM Network	72
5.5 Discussion	77
 Chapter 6: Exploitation of Results.....	 78
6.1 Developing a Library of Traffic Distributions	78
6.2 Using a Spreadsheet Program to Predict Network Performance.....	80
6.3 Spreadsheet and Distribution Library Example.....	83
6.4 Monitoring the Accuracy of Prediction Results	86
6.5 Fast Simulation Using the Q Value	87
 Chapter 7: Conclusions, Contributions and Future Research.....	 88
7.1 Contributions.....	89
7.2 Future Research.....	90
 References.....	 91

List of Figures

Figure 3.1	Simulation model for an ATM Switch	19
Figure 4.1	Line 1 Summer and Fall Distributions Normalized to Summer Means.....	33
Figure 4.2	Line 1 Summer and Fall Distributions Normalized to Fall Means	33
Figure 4.3	Line 1 Summer and Fall Hourly Cell Counts.....	33
Figure 4.4	Line 2 Summer and Fall Distributions Normalized to Summer Means.....	34
Figure 4.5	Line 2 Summer and Fall Distributions Normalized to Fall Means	34
Figure 4.6	Line 2 Summer and Fall Hourly Cell Counts.....	34
Figure 4.7	Line 3 Summer and Fall Distributions Normalized to Summer Means.....	35
Figure 4.8	Line 3 Summer and Fall Distributions Normalized to Fall Means	35
Figure 4.9	Line 3 Summer and Fall Hourly Cell Counts.....	35
Figure 4.10	Line 4 Summer and Fall Distributions Normalized to Summer Means.....	36
Figure 4.11	Line 4 Summer and Fall Distributions Normalized to Fall Means	36
Figure 4.12	Line 4 Summer and Fall Hourly Cell Counts.....	36
Figure 4.13	Line 5 Summer and Fall Distributions Normalized to Summer Means.....	37
Figure 4.14	Line 5 Summer and Fall Distributions Normalized to Fall Means	37
Figure 4.15	Line 5 Summer and Fall Hourly Cell Counts.....	37
Figure 4.16	Line 6 Summer and Fall Distributions Normalized to Summer Means.....	38
Figure 4.17	Line 6 Summer and Fall Distributions Normalized to Fall Means	38
Figure 4.18	Line 6 Summer and Fall Hourly Cell Counts.....	38
Figure 4.19	Line 7 Summer and Fall Distributions Normalized to Summer Means.....	39
Figure 4.20	Line 7 Summer and Fall Distributions Normalized to Fall Means	39
Figure 4.21	Line 7 Summer and Fall Hourly Cell Counts.....	39
Figure 4.22	Line 8 Summer and Fall Distributions Normalized to Summer Means.....	40
Figure 4.23	Line 8 Summer and Fall Distributions Normalized to Fall Means	40
Figure 4.24	Line 8 Summer and Fall Hourly Cell Counts.....	40
Figure 4.25	Line 9 Summer and Fall Distributions Normalized to Summer Means.....	41
Figure 4.26	Line 9 Summer and Fall Distributions Normalized to Fall Means	41

Figure 4.27 Line 9 Summer and Fall Hourly Cell Counts.....	41
Figure 4.28 Line 10 Summer and Fall Distributions Normalized to Summer Means	42
Figure 4.29 Line 10 Summer and Fall Distributions Normalized to Fall Means.....	42
Figure 4.30 Line 10 Summer and Fall Hourly Cell Counts	42
Figure 4.31 Line 11 Summer and Fall Distributions Normalized to Summer Means	43
Figure 4.32 Line 11 Summer and Fall Distributions Normalized to Fall Means.....	43
Figure 4.33 Line 4 Summer and Fall Hourly Cell Counts.....	43
Figure 4.34 Traffic Distribution of Lines with AggregatedQ Factor ≤ 2.5	50
Figure 4.35 Traffic Distribution of Lines with Aggregated Q Factor from 3.0 to 4.0....	51
Figure 4.36 Traffic Distribution of Lines with Aggregated Q Factor Larger Than 4.0..	51
Figure 4.37 Q Values for OctExt4 Trace, 60 Samples per Q Value	56
Figure 4.38 Q Values for OctExt4 Trace, 300 Samples per Q Value.....	56
Figure 4.39 Q Values for OctExt4 Trace, 600 Samples per Q Value.....	57
Figure 4.40 Q Values for OctExt4 Trace, 1800 Samples per Q Value	57
Figure 4.41 Q Values for OctExt4 Trace, 7200 Samples per Q Value	58
Figure 4.42 Q Values for OctExt4 Trace, 36000 Samples per Q Value.....	58
Figure 4.43 Q Values for OctExt Trace, 60 Samples per Q Value.....	59
Figure 4.44 Q Values for OctExt Trace, 300 Samples per Q Value	59
Figure 4.45 Q Values for OctExt Trace, 600 Samples per Q Value	60
Figure 4.46 Q Values for OctExt Trace, 1800 Samples per Q Value.....	60
Figure 4.47 Q Values for OctExt Trace, 7200 Samples per Q Value.....	61
Figure 4.48 Q Values for OctExt Trace, 36000 Samples per Q Value	61
Figure 4.49 Q Values for Line 1, Bell Network, 24 Samples per Q Value.....	63
Figure 4.50 Q Values for Line 1, Bell Network, 48 Samples per Q Value.....	63
Figure 4.51 Q Values for Line 1, Bell Network, 96 Samples per Q Value.....	63
Figure 4.52 Q Values for Line 2, Bell Network, 24 Samples per Q Value.....	64
Figure 4.53 Q Values for Line 2, Bell Network, 48 Samples per Q Value.....	64
Figure 4.54 Q Values for Line 2, Bell Network, 96 Samples per Q Value.....	64
Figure 4.55 Q Values for Line 3, Bell Network, 24 Samples per Q Value.....	65

Figure 4.56	Q Values for Line 3, Bell Network, 48 Samples per Q Value.....	65
Figure 4.57	Q Values for Line 3, Bell Network, 96 Samples per Q Value.....	65
Figure 4.58	Q Values for Line 4, Bell Network, 24 Samples per Q Value.....	66
Figure 4.59	Q Values for Line 4, Bell Network, 48 Samples per Q Value.....	66
Figure 4.60	Time vs Cell Counts for a Bell ATM Network line - Sampling Interval of 3600 Secs.....	67
Figure 4.61	Bellcore Self Similar Ethernet Data Using Different Sampling Intervals....	67
Figure 5.1	Q incoming vs Q outgoing for switches in the Summer data set using the Theoretical Q Combination Rule.....	76
Figure 5.2	Q incoming vs Q outgoing for switches in the Fall data set using the Theoretical Q Combination Rule.....	76

List of Tables

Table 4.1	Means, Deviations, Q Values of all Network Lines in Summer and Fall	29
Table 4.2	Q Value Ranges for Summer and Fall Data Sets	46
Table 4.3	Distribution Breakdown for Lines in Different Q Ranges in Summer	47
Table 4.4	Distribution Breakdown for Lines in Different Q Ranges in Fall	48
Table 4.5	Aggregated Distribution Breakdown for Lines in Different Q Ranges - Summer Data	49
Table 4.6	Aggregated Distribution Breakdown for Lines in Different Q Ranges - Fall Data.....	49
Table 4.7	Aggregated Q Ranges for Summer Data Set.....	50
Table 4.8	Aggregated Q Ranges for Fall Data Set	50
Table 4.9	Chi-Squared Analysis Between Table 4.5 and Table 4.6	52
Table 4.10	Q Values for Different Ethernet Traces using Different Sampling Intervals..	54
Table 5.1	Q Combination Formula Values for Summer Data Set	74
Table 5.2	Q Combination Formula Values for Fall Data Set.....	75
Table 6.1	Spreadsheet Application Example Before Changes.....	83
Table 6.2	Spreadsheet Example After Increasing M_2 to 50000 and V_2 to $2.5 * 10^9$	84
Table 6.3	Example Distribution for $Q = 1.13$	84
Table 6.4	Spreadsheet Example After Increasing M_2 to 100000 and V_2 to $5.0 * 10^9$..	85
Table 6.5	Example Distribution for $Q = 1.51$	86

Glossary

AAL	<i>ATM Adaptation Layer</i> - A protocol used to convert non-ATM information to and from ATM cells.
ABR	<i>Available Bit Rate</i> - An ATM Forum service category defined to support both connectionless and connection-oriented services. All ABR connections share resources fairly, and receive as much bandwidth as is available on an as-needed basis.
ATM	<i>Asynchronous Transfer Mode</i> - A switching/transmission technology which uses 53 byte cells as a basic unit of transfer. The ATM Cell is divided into 5 bytes of ATM Layer overhead and 48 bytes of ATM payload.
CBR	<i>Constant Bit Rate</i> - An ATM traffic class that is denoted by its constant traffic service profile.
H	<i>Hurst parameter</i> - A measure of the degree of self similarity.
IDS	<i>Index of Dispersion</i> - A measure to define the burstiness of data. It captures the variability of traffic over different time scales
ISDN	<i>Integrated Services Digital Network</i> - ISDN provides a wide range of voice and non-voice services within the same network using a limited set of connection types and multipurpose user-network interface arrangements.
LAN	<i>Local Area Network</i> - A system designed to inter-connect computing devices over a restricted geographical area of approximately one mile.
MMPP	<i>Modulated Markov Poisson Process</i>
PVC	<i>Permanent Virtual Circuit</i> - An end-to-end logical connection.
QoS	<i>Quality of Service</i> - A particular instance of a set of generic parameters that are used to describe traffic characteristics of an ATM connection.
Q Value	The ratio of the mean traffic on a device to the standard deviation of the traffic.
SNMP	<i>Simple Network Management Protocol</i> - A standard for the management of entities in a TCP/IP local area network.
SVC	<i>Switched Virtual Circuit</i> - A connection which is set up on demand via the ATM signalling protocol. Such connections tend to be of shorter duration than PVCs, and are not automatically re-established after a system restart.
UBR	<i>Unspecified Bit Rate</i> - An ATM service category that emulates the connectionless services provided by conventional bridged and routed data networks. UBR provides "best effort" delivery and is used for applications that do not require guarantees of low cell loss or low delay.
UNI	<i>User Network Interface</i> - The interface used to connect user equipment to network equipment.

UPC	<i>Usage Parameter Control</i> - The set of actions taken by the network to monitor and control traffic at the user access point.
VBR	<i>Variable Bit Rate</i> - An ATM service category that provides guaranteed low cell loss and low delay for applications such as video and frame relay, which are characterized by an on or off source with known, predictable transmission patterns. During the on period, cells are transmitted at the peak information rate. No cells are transmitted during the off period.
VCI	<i>Virtual Circuit Identifier</i> - A field in the ATM header that identifies virtual channels. A virtual channel is a logical communication channel that is available across a physical ATM interface
VPI	<i>Virtual Path Identifier</i> - A field in the ATM header that identifies virtual paths. A virtual path is a logical communication channel that is available across a physical ATM interface and that can carry one or more virtual channels.
WAN	<i>Wide Area Network</i> - A geographically dispersed network.

Chapter 1

Introduction

With the emergence of high speed ATM networks, which provide the end user with the potential to request different quality of service guarantees and a flexible amount of bandwidth on demand, ATM service providers must be able to design and provision a cost effective network. Since these ATM networks are a new technology, the performance characteristics of the networks are not well known. It is desirable for service providers to be able to anticipate the impact a change in the network will have on the rest of the network. The changes that are of interest include adding or removing a permanent virtual circuit (PVC) or switched virtual circuit (SVC) from a path in the network. By being able to assess the impact of adding a PVC or SVC, the provider can determine the best route to allocate to a new customer. The provider can also analyze the impact of various traffic loads on the network. Similarly, predicting the effect of increasing or decreasing the capacity of a transmission line would allow the provider to install the appropriate bandwidth between devices. Currently, the service providers install transmission lines that handle a much greater capacity than that required by the traffic on the line. This is due to the lack of understanding of the performance characteristics of the network. If the providers can predict performance of the network with different line rates, then they can save money by installing the required amount of bandwidth, instead of over provisioning.

The ability to predict the impact of a change in the network would be efficient and cost effective. An equally useful idea to service providers is the ability to predict the percentage of time that a network device is congested. The device could be an individual PVC or SVC, a transmission line, a switch or router in the network. By being able to predict the congestion rates of different devices, the service provider can anticipate when a user will need more bandwidth or when a faster transmission line needs to be installed, before network performance is degraded. Currently, changes are not made until congestion occurs in the network. Predicting the congestion

would clearly be beneficial to both the service providers and the users of the network.

This thesis investigates two different ways to predict performance and congestion conditions in an ATM network. Both of the methods researched utilize real data that was collected from an existing ATM network that is operated by Bell Sygma. The first method investigated uses simulation to make the predictions. A technique to speed up the execution time of the ATM network simulation was developed and is described in this thesis. The network can be modeled and changes to the model made. The effects of the changes on the simulation can be obtained in near real time speed.

A second method to predict the performance of the network is described and empirical evidence is presented from the Bell ATM network to support the principles behind the model. This method uses the history of the network to predict the future characteristics of the network. The idea behind this method stems from the observation that a device's traffic distribution is stable with time. This is notwithstanding a known increase or decrease in traffic, such as the addition or removal of a customer. A second observation made is that for any device, the ratio of the mean traffic on a device to its standard deviation is representative of the traffic distribution for that device. This ratio of mean traffic to the standard deviation of the traffic is referred to as the Q factor. Devices with similar Q factors appear to have similar traffic distributions. This thesis provides empirical evidence from the ATM network that for a given Q factor, a traffic distribution can be mapped to it. Combining these two observations allows the prediction of performance measures. For example, a library of traffic distributions that occur on the network can be built and each distribution associated with its corresponding Q factor. Then for a particular device, its Q factor can be calculated by measuring the traffic on the device, and the corresponding distribution can be looked up in the library. From this distribution, the percentage of time the device is expected to be congested can be read off.

This concept of associating an identifying attribute such as the Q factor with a probability distribution is a novel idea. It allows service providers to estimate performance characteristics of the ATM network quickly, and because it is based on the past network characteristics, its

prediction accuracy can be continuously monitored.

1.1 Objectives

The objectives of this thesis are to

1. Investigate the traffic characteristics of a large, existing ATM network that is operated by Bell Sygma.
2. Determine the feasibility of using an existing commercial simulation package to predict the performance impact on the ATM network when changes are made to the network. These changes could include adding a new customer, adding or deleting a PVC, or changing the bandwidth capacity of a transmission link.
3. Investigate a proposed empirical method to estimate congestion rates in an ATM network, based on the networks past traffic characteristics. This method uses the traffic counts collected on a device over a period of time to calculate the Q factor for the device. The Q factor represents the ratio of the mean traffic on that device to its standard deviation. It appears that the Q value can then be associated with a particular traffic distribution. This thesis presents empirical evidence that devices on an ATM network with similar Q values have similar traffic distributions. Given that you can associate a traffic distribution with a Q value, then given a Q value for a device, the corresponding distribution can be looked up. From this distribution, it can be determined what percentage of time the device will be performing over a certain value. If this value corresponds to the congestion point of the device, then congestion rates are predicted. A main goal of this thesis is to provide empirical evidence that this method can indeed be used to predict performance conditions of an ATM network. The following steps are shown in this thesis, to provide the empirical evidence.
 - a) Show empirically that the traffic distributions on the Bell ATM network are stable over time.
 - b) Investigate the relationship between the traffic distribution of a device and the Q factor associated with the device.
 - c) Show that the Q value of a device varies slowly with time for self similar traffic.
 - d) Determine the required sampling length to measure the Q value for the Bell network.
 - e) Show mathematically that Q values for devices entering a switch can be combined to determine the Q value for the switch.

1.2 Assumptions

The work in this thesis was done based on the following assumptions:

- 1) The traffic that exists on the Bell Sygma ATM network can be classified as self similar. This assumption was derived from the Bellcore study, described in [19], that concluded that aggregate Ethernet data is self similar. Since the traffic that was analyzed in this thesis from the Bell Sygma network is aggregate Ethernet traffic, it was assumed that this data is also self similar.
- 2) The distributions of the traffic on the ATM switches is stable over time. The procedure used in Chapter 4 to associate a Q value with a distribution uses this assumption. Section 4.4 supports this assumption with evidence from the network showing that distributions are stable over time, notwithstanding a known increase or decrease in traffic.
- 3) The process of normalizing random variables can be used on the probability distributions from the Bell Sygma ATM network. Normalization is used in Chapter 4 to generate data that can be aggregated together to produce larger data sets.

1.3 Overview

This thesis is organized as follows. Chapter 2 reviews the characteristics of Poisson and self similar traffic. It also describes various studies that show that aggregated Ethernet data, variable bit rate video traffic and ISDN office traffic exhibit self similar characteristics. Chapter 2 also describes the statistical technique of normalizing random variables, a process that is used throughout this thesis.

Chapter 3 describes the components of a simulation along with a description of a simulation that was developed for the Bell ATM network. This chapter discusses the problems that occur when trying to simulate large, high speed networks. A method that can be used to scale down the simulation model is described, which can improve the execution speed of the simulation by several orders of magnitude.

Chapter 4 introduces the empirical method of using the Q factors in a network to predict congestion and performance estimates. This chapter covers the steps required to show that Q factors can be used as performance indicators and includes the results obtained from the Bell ATM network which support the claim.

Chapter 5 explains how the Q factor of a switch can be determined from the Q factors of all of the incoming or outgoing lines to the switch. This claim is supported with results from the Bell ATM network and is also described mathematically for various probability distributions.

Chapter 6 describes the practical applications of the results from this thesis. Chapter 7 lists the conclusions and contributions made, as well as ideas for future research on the topic.

Chapter 2

Traffic Characteristics and Statistical Background

In order to analyze and predict performance characteristics of today's high speed networks, such as frame relay and ATM, it is necessary to understand the characteristics of packet and cell based traffic. If the fundamental traffic model used for network analysis is an inappropriate representation of the actual traffic carried by the network, then the accuracy of the analysis is uncertain. Numerous analytical models have been developed based on Poisson and Modulated Markov Poisson Processes (MMPP). However, recent studies measuring the real traffic on packet based networks have shown that standard traffic models such as the Poisson based models are inadequate representations of the actual data on the network. [7] These studies have shown that packet and cell based traffic appears self similar, meaning that the traffic distribution appears the same regardless of the time scale used for measurement.

This chapter discusses the characteristics of self similar distributions and how they differ from standard Poisson based distributions, emphasizing why Poisson models appear to be inappropriate for modeling packet based networks. Section 2.1 describes the characteristics of self similar traffic, which is also referred to as fractal traffic. There have been several large traffic studies performed on different types of traffic, resulting in the discovery that aggregated Ethernet data, variable bit rate video traffic and ISDN traffic exhibit self similar characteristics. Each of these studies is described and the significance of the results is explained in Sections 2.1.1 to 2.1.3. The final section of this chapter provides some statistical background on the process of normalizing random variables, a method that is used throughout this thesis.

2.1 Self Similar Traffic Characteristics

Conceptually, self similar traffic does not contain a natural length for a burst, meaning that traffic bursts appear on a wide range of time scales. In order to describe self similarity mathematically, the following definitions are required.

Let $X = (X_t; t = 0, 1, 2, \dots)$ be a random variable. The mean of the distribution of X , μ , is defined as the expectation of X , $E(X)$. The mean represents the value about which deviations in one direction exactly equal deviations in the other.[12]

The variance of the distribution of X , σ^2 is defined as the expectation of the squared deviation from the mean, $E(X - \mu)^2$. The variance represents the degree of spread or dispersion in the distribution. The more the values in the distribution differ from each other and from the mean, the larger the variance. [12]

The standard deviation of the distribution of X , σ , is an index of variability in the original measurement units. This is in contrast to the variance, which is in squared units of measurement. The standard deviation is equal to the square root of the variance.

The covariance of two random variables X and Y is defined as $\text{Cov}(X, Y) = E(XY) - E(X)E(Y)$. The covariance represents the direction of the relation between X and Y . If a higher value of X tends to correspond to a higher value of Y , and a lower value of X tends to correspond to a lower value of Y , then the covariance of X and Y is positive. If a higher value of X tends to correspond to a lower value of Y and vice-versa then X and Y are negatively related. [12]

Since the covariance is a product of the means of X and Y , the magnitude of the covariance depends on the units of measurement for X and Y . This makes it hard to determine how strong the direction is between X and Y . The correlation coefficient, ρ , divides the covariance by a product using the same units of measurement as X and Y , which results in a value which is independent of units of measurement. The definition for the correlation coefficient is

$$\rho = \frac{\text{cov}(X, Y)}{\sigma_x \sigma_y}$$

Self similarity falls into a class of functions called time series. These functions have a random variable X_t for each value of time, t , on some interval, generating a random function of

time. [23] Time series are often looked at as stochastic processes. A stochastic process is defined as a family of random variables $\{X_t; t \in T\}$ such that for every finite set of choices of $t \in T$, say $t_1, \dots, t_n$, the set $X_{t_1}, \dots, X_{t_n}$ of random variables has a joint probability distribution function. [23]

Stationary processes, also called stationary time series, can be strictly stationary or weakly stationary. A strictly stationary time series $\{X_t; t \in T\}$ has the property that any finite set of random variables $(X_{t_1}, \dots, X_{t_n})$ has the same joint distribution as the set $(X_{t_1+h}, \dots, X_{t_n+h})$ for any value of h . [23] Therefore, the joint distributions of the set of random variables $(X_{t_1+h}, \dots, X_{t_n+h})$ for different values of h are all the same and depend only on the time differences $t_2 - t_1, t_3 - t_2, \dots, t_n - t_{n-1}$. The mean value for $\{X_t; t \in T\}$ is a constant, $E[X_t] = c$, for all t .

The definition of self similarity uses weakly stationary time series. Weakly stationary time series assumes that for all t , $E[X_t]$ is a constant which can take on the value 0. This type of time series also has the weaker restriction that the distributions in the set $(X_{t_1+h}, \dots, X_{t_n+h})$ have the same covariance matrix for all h , instead of the same joint distribution. This means that the covariance matrix only depends on the time differences $t_2 - t_1, t_3 - t_2, \dots, t_n - t_{n-1}$ and as a result the covariance of X_{t+h} and X_t will be a function of h only, $E[X_t X_{t+h}] = y_h$. [23]

The covariance function of the time series $\{X_t; t \in T\}$ means that the covariance y_h , is considered as a function of h . y_h is referred to as the auto-covariance with lag h .

Let $X = (X_t; t = 0, 1, 2, \dots)$ be a covariance stationary stochastic process with mean μ , variance σ^2 and autocorrelation function $r(k)$, $k > 0$. X has an autocorrelation function of the form

$$r(k) \sim k^{-\beta} L(t) \dots\dots\dots (2.1)$$

where $L(t)$ is a slowly varying function.

Autocorrelation refers to the correlation between successive terms of the series. For example, the correlation between X_1 and X_2 or X_n and X_{n-1} is autocorrelation.

Let $X^{(m)}$ be the covariance stationary time series obtained by averaging the original series X over non-overlapping blocks of size m .

Using these definitions, self similarity can be identified in three equivalent ways [19].

1. The variance of the mean decays slowly. This means that for large m , the variance of the sample mean decreases more slowly than the sample size.

$$\text{Var}(X^{(m)}) \sim am^{-\beta} \text{ as } m \rightarrow \infty \text{ and } 0 < \beta < 1 \dots\dots\dots(2.2)$$

2. Long-range dependence exists. This occurs when the autocorrelation function decays hyperbolically fast ($k^{-\beta}$), implying a non-summable autocorrelation function.

$$\sum_k r(k) = \infty \dots\dots\dots(2.3)$$

3. The spectral density function obeys a power law near the origin. This means that the spectral density function $f(\partial)$ increases without limit as the frequency tends to zero.

$$f(\partial) \sim a\partial^{-\gamma} \text{ as } \partial \rightarrow 0, 0 < \gamma < 1, \gamma = 1 - \beta \dots\dots\dots(2.4)$$

The most evident feature of self similar processes is that their aggregated processes $X^{(m)}$ possess a nondegenerate correlation structure as $m \rightarrow \infty$ [19]. This is demonstrated by measuring the distribution over different time scales. The graphs of each time scale would all look similar to each other.

Since in practice the data sets used for analysis are finite, it is not possible to decide whether or not the asymptotic relationships in equations 2.1 to 2.4 hold [19]. However, two different methods have been developed to estimate the degree of self similarity (H) in a process.

The first method employs time domain analysis to estimate the self similarity parameter H . It consists of taking logarithmically spaced values of n and plotting $\log(R(n) / S(n))$ versus $\log(n)$, which yields the pox diagram of R/S . The Hurst parameter, H , is the slope of the graph. If H is between zero and one then the traffic is self similar. The R/S statistic is the rescaled adjusted range statistic and is signified by $R(n)/S(n)$. Mathematically,

$$\frac{R(n)}{S(n)} = \frac{1}{S(n)} * (\max(0, W_1, W_2, \dots, W_n) - \min(0, W_1, W_2, \dots, W_n))$$

where $W_k = (X_1 + X_2 + \dots + X_k) - kX(n)$ $k \geq 1$

and $S(n)$ is the sample variance and $X(n)$ is the sample mean.

The second method analyzes the variances of the aggregated process. The variance time plot is derived by plotting $\log(\text{var}(X^{(m)}))$ versus $\log(m)$ and by fitting a least squares line through the points. Values of the slope (β) between -1 and 0 suggest self similarity, with the Hurst parameter being $1 - (\beta/2)$.

2.1.1 Aggregated Ethernet Traffic

A large LAN traffic study was performed at Bellcore, based on the recorded arrival of hundreds of millions of packets of Ethernet traffic on their network. The packets were recorded without loss and with the arrival timestamp accurate to 100 microseconds. The data was collected between August 1989 and February 1992 on various Ethernet LANs at the Bellcore Morristown Research Center. The traffic on these LANs is typical of a research or software development environment department where workstations are the primary machines used by employees [19]. It is also typical in that the initial LAN installation was well thought out and planned but then grew randomly.

The traffic was collected in four different sets, at different times of the year and on different LAN segments. Each data set contains between twenty and forty consecutive hours of Ethernet traffic. Two of the data sets were taken from an Ethernet network that was used by researchers who utilized the network for a variety of tasks, including software development. The traffic on the network is mainly IP based.

The other two data traces monitor a segment of the network that handles the Bellcore Internet traffic and therefore not only saw internal Bellcore traffic but also saw the traffic between Bellcore and the rest of the world. The traffic traces are representative of the different types of traffic that are present in various segments of the LAN.

The conclusions made from this study stemmed from the observation that regardless of the time interval used to measure the data, the traffic distributions for these different time scales were all 'similar'. On all time scales, the traffic looks the same and is bursty, with no natural length for the burst. This is in contrast to Poisson traffic, which has the characteristic that the longer the measurement interval, the smoother the traffic.

The main results ascertained from the Bellcore Ethernet study are as follows [19].

1. Ethernet LAN traffic is statistically self similar, regardless of where in the network the data was collected from and regardless of what period of time it was collected over.
2. The Hurst parameter, H , which measures the degree of self similarity can be used to measure the burstiness of data. The higher the value of H , the burstier is the traffic.
3. Different types of Ethernet LAN traffic, such as external LAN and external TCP traffic, have the same self similar characteristics as the overall LAN traffic.
4. Currently used packet traffic models do not accurately model the characteristics of packet data.

2.1.2 Variable Bit Rate Video Traffic

The transport of variable bit rate (VBR) video images from applications such as video conferencing and full motion broadcast is expected to become a major traffic component of ATM networks. The work described in [2] analyzes a wide variety of video data ranging in length from a few hundred frames (15 seconds) to 170 000 frames (2 hours). The traffic represents a variety of scenes and was recorded using different types of VBR video software and hardware codecs. The intent of the analyses was to investigate properties of VBR video data that were universal to VBR video, independent of scene and codec.

The R/S analysis and variance analysis methods described in Section 2.1 were applied to the VBR data. It was also analyzed using a more rigorous periodogram based method. The estimates of H , the self similarity parameter, from the three different methods were all in agreement. The researchers concluded that long range dependence was apparent in all VBR traffic.

independent of scene and codec.

2.1.3 ISDN Office Automation Traffic

In addition to the previous studies on the Ethernet LAN data and VBR video data, a small study on ISDN traffic was made and was also found to be self similar. These measurements were taken from a standard application of X.25 and ISDN. The entire data stream transmitted and received on a users ISDN channel (16 kbps) was measured. User applications were mainly electronic mail, file retrieval and word processing. Users consisted of secretaries, managers and engineers. The study consisted of one week of continuous data, resulting in 116 000 packets in total. [7]

The study showed that the extreme variability in packet arrival times seems to be best described with heavy tailed distributions and that the traffic exhibited the characteristics of self similarity.

2.2 Traditional Poisson Based Traffic Characteristics

Traditionally, traffic models for packet and cell based networks have been built on Poisson or Markov Modulated Poisson Processes (MMPP). This is partly due to the strong understanding of the characteristics of these models by the research community. The previous sections have described the characteristics of self similar traffic and the existence of self similar attributes in traditional LAN traffic and VBR video traffic. This section describes the attributes of Poisson based models and how they differ from self similar models. It becomes evident that using Poisson based models to represent packet and cell based traffic is not appropriate.

Typical packet traffic models have the property that their aggregated processes $X^{(m)}$, tend to second order pure noise for large values of m [19]. Mathematically, for all $k \geq 1$,

$$r^{(m)}(k) \rightarrow 0 \text{ as } m \rightarrow \infty \dots\dots\dots(2.5)$$

This is in contrast to self similar models, which have

$$r^{(m)}(k) \rightarrow r(k) \text{ as } m \rightarrow \infty \dots\dots\dots(2.6)$$

Existing models can also be identified by the following features, which are clearly different from self similar characteristics [19]:

1. The variance decreases like the reciprocal of the sample mean.

$$\text{var}(X^{(m)}) \sim am^{-1} \text{ as } m \rightarrow \infty \dots\dots\dots(2.7)$$

2. Short term dependence exists. This means that the autocorrelation function decreases exponentially fast.

$$r(k) \sim p^k \text{ where } 0 < p < 1 \dots\dots\dots(2.8)$$

This implies a summable autocorrelation function $\sum r(k) < \infty$. Using a short range dependent process to model a process which is actually long range dependent results in an error that is wrong by a factor which tends to infinity as the sample size increases [2].

3. The spectral density function is bounded by the origin, meaning that the density function is finite.

When Poisson data streams are aggregated together, the result is a smooth graph. The Bellcore Ethernet data shows that aggregating self similar data streams produces a bursty graph.

Another difference between conventional Poisson and MMP processes and self similar models is the index of dispersion (IDS). The IDS is a commonly used measure to define the burstiness of data. It captures the variability of traffic over different time scales. The IDS is the variance of the number of arrivals during an interval of length L , divided by the expected value of the number of arrivals during the same interval [2]. In pure Poisson models the IDS is constant with a value of one. In other conventional models such as batch Poisson, hyperexponential and MMPP, the IDS converges quickly to a fixed value. In contrast, the self similar models produce a monotonically increasing index of dispersion [9].

2.3 Normalization

One of the most useful distributions in statistics is the normal distribution. Many distributions can be approximated using the normal distribution and normalized random variables can be compared with each other. A random variable X that has a normal distribution with mean μ and variance σ^2 has a probability density function defined by

$$f(X) = \left(\frac{1}{\sigma \sqrt{2\pi}} \right) * \exp \left(-\frac{(x - \mu)^2}{(2 * \sigma^2)} \right) \quad -\infty < X < \infty \dots\dots\dots (2.9)$$

X is $N(\mu, \sigma^2)$.

This thesis uses the normal distribution to approximate the distribution of the traffic on an ATM network. In this situation, traffic does not range from negative infinity to positive infinity. An approximation is used and the range of X is between 0 and infinity.

If X has mean equal to zero and variance equal to one, then X has a standard normal distribution, denoted $N(0,1)$. The probability density function of the standard normal distribution is bell shaped and is symmetrical about the mean.

Any normal variable X can be converted to a standard normal variable Z by the formula

$$Z = (X - \mu) / \sigma \quad \dots\dots\dots (2.10)$$

This process of converting a normal variable to a standard normal variable is called normalization.

This thesis uses normalization to compare the distributions of different transmission lines. Since the normal distribution can be used to approximate many different types of distributions, it is assumed that the traffic distributions on the lines in the network can be approximated by the normal distribution. Each line is normalized, using Equation 2.10, resulting in all lines having a mean of zero and variance of one. This means that the normalized data from lines that have similar distributions can be combined to provide more information about the distribution.

Chapter 3

Simulating an ATM Network

3.1 Simulation Experiments

Computer simulation is a useful tool in predicting the outcome of a system under various situations. It is beneficial to develop a simulation of a system before actually creating the system, or as is the case in this thesis, to predict how changes to the existing system will effect it before actually making the changes.

There are several steps involved in developing a simulation of a system. This thesis concentrates on discrete simulation in which the state variables change at finite points of time. These changes in state are referred to as events. In this thesis, which simulates an ATM network, the changes of state correspond to the arrival of cells at switches within the network.

The following section gives a brief description of the steps involved in developing a computer simulation. These steps include model development, verification, validation, experimentation and results analysis. Then, a simulation of an ATM network is described and the problems encountered with the simulation are discussed.

3.1.1 Model Development and Validation

In order to develop a simulation, a model must be constructed. A model is a simplified representation of the problem. The system being modeled must be broken down into different components and the relationships between these different components must be fully understood.

One of the first considerations is to determine the level of detail that is required in the model to provide the simulation. It is normally not feasible to develop a full scale representation of the system, so unnecessary detail has to be taken out. Otherwise, the simulation run time will be too long to be useful. This process of removing detail from the system is referred to as scaling. [8]

When developing the model, the different events that can occur in the system must be considered. Similarly, the model inputs, or the variable parameters of the system must be defined.

The simulation language or program to be used to develop the model is also an important step in the simulation process. Simulation software can be broken down into simulation languages and simulators. Simulation languages allow the user to develop the full model. Simulators are applications, often with a graphical user interface that contain many built in models and utilities.

Verification is the process of making sure that the simulation model matches the developers concept of how the system works. A large part of verification consists of debugging. Each component of the simulation is verified, either against a hand calculation or by just ensuring that the expected result is returned, when a result is known. As with any other software program, the program needs to be stress tested against unusual inputs.

One of the most important steps in simulation is the validation of the model. Validation is the process of checking that the model is working as intended and accurately represents the system being modeled. [20] Validation should be applied all the way through the simulation development process to both the input and output segments of the system. There is no right way to perform validation, and it is one of the most difficult parts of simulation. Validation can be comprised of observation - checking that the model appears to be working as expected. It should also include statistical testing. This could be applying goodness of fit tests such as the chi-square or Kolmogorov-Smirnov test to theoretical input data if it is being used to model empirical data. Validation can also include sensitivity analysis, which involves changing one of the models input by a small amount and investigating how this effects the output.

If the system being modeled already exists, then the output from the simulation can be compared against the output from the real system for validation.

The choice of the probability distributions to use to model the activities within the simulation is an important component of the simulation. If distributions are chosen that do not characterize the system properly, then the results will be meaningless. The distributions of the arrival times and other time related activities in the real system should be analyzed to determine the distribution to use in the simulation. [8] When the simulation is being executed, either a trace of real data can be used, or the closest distribution that fits the data can be used to generate theoretical data.

Once the model has been developed and implemented the experimentation can take place. This consists of designing the simulation runs that are needed, running them and then analyzing the output. When designing the experiments, the values for input parameters must be decided along with the length of the simulated run time, the number of runs to be executed and the random number seeds to be used.

These steps are general guidelines to the approach to take to develop a simulation. The remaining sections describe the simulation of an ATM network that was developed for this thesis along with the results and problems encountered throughout.

3.2 Simulation Goals and Constraint

There are different ways to go about predicting performance and congestion rates in data networks. One approach is to develop a model of the system and simulate it under various conditions. Another possibility is to mathematically model and analyze the system. Alternatively, estimates can be made directly, based on the previous performance of the network under study. Both the simulation and direct estimation approaches were used in this project, although the direct estimation method, described in Chapter 5, yielded far better results than the simulation did.

The level of detail included in a simulation model has a direct impact on the length of time required to execute the simulation. Similarly, the level of detail included in the model yields results of different accuracies. The goals of this simulation study were to

- determine whether the level of detail that could be represented in a simulation of a large size ATM network is enough to give congestion rate predictions in a reasonable time frame.
- analyze the traffic characteristics of the current ATM network traffic in order to accurately model it.

A successful simulation would be used by Bell Sygma to monitor their ATM network and determine the optimum required link capacity between nodes in the network. It would also be used to determine the effect on the entire network of adding a PVC or SVC to a certain path. Finally, the simulation would be used to model the differences between different vendors equipment.

One of the main constraints of the simulation was to use an existing commercial simulation package, to ensure that the maintenance of the simulator was not done by Bell, but by an established simulation company. Different simulation packages were evaluated, and the resulting simulation was developed using Opnet, a package developed by MIL3. Opnet is a simulation language that is specifically oriented towards communications networks. It is built using the C language and thus programming for the model is in C. Opnet was chosen because it allows the user to program the details of their model, instead of being bound to built in models that come with

the system. It also has an interface that is easy to use and can encapsulate different sections of the network into subnets for readability. This was a concern in the ATM network, since it is expected to grow to a very large size fairly quickly.

The following sections describe the existing Bell ATM network and the discrete event simulation model developed. The problems that were encountered when attempting to simulate a network of a large size are discussed in Section 3.5. The main problem found was that attempting to simulate a large network at rates of 155 Mbps requires an unacceptable amount of processing time. Finally, the strategies used to solve the processing time problem, and how they led into the direct estimation method described in the remainder of the paper, are discussed in Section 3.7.

3.3 Network Description

The ATM network used in this analysis consists of 30 switches interconnected across cities and provinces within Canada. All switches are homogenous, although in the near future they will not be. These switches connect LANs and WANs of various sizes with 55 links in the network. The link rates are all either 44 Mbps or 155 Mbps, with several PVCs per link. In the future, there will be hundreds of PVCs per switch, requiring the simulation to scale to a very large network.

Although the network is lightly loaded, the traffic currently on it is typical of LAN traffic that is expected to be prominent on ATM networks. The traffic is believed to be self similar, since it is an aggregation of Ethernet traffic. Section 2.3 showed that the Bellcore data was self similar, and it is also aggregated Ethernet traffic. The traffic rates on the lines vary dramatically throughout the network, with some lines containing two orders of magnitude more traffic than others.

3.4 Simulation Model

Each switch in the Bell ATM network was modeled using the Opnet ATM switch model as the base. This model was extended to include vendor specific features that existed in the Bell network. The model supports dynamic routing and four different quality of services classes which correspond to Variable Bit Rate (VBR), Constant Bit Rate (CBR), Available Bit Rate (ABR) and Unspecified Bit Rate (UBR). Each virtual path has a quality of service associated with it and all virtual channels on that path have the same quality of service. Call admission control is performed based on the algorithm that the sum of the peak cell rates on a link must be less than the available bandwidth of the link. Figure 3.1 depicts the ATM switch model used.

Figure 3.1 Simulation model for an ATM Switch

The ATM Layer provides the signaling required to establish a PVC. All signaling protocols used conform to the UNI 3.1 specifications published by the ATM Forum. The ATM Layer sends the data received from the incoming ATM cells to the ATM Adaptation Layer (AAL) and also receives data from the AAL and sends it to the ATM Management module. The ATM layer also

handles the routing functions for the switch.

The routing algorithm used requires each node to have its own routing table. The table contains records of all known routes to all of the known nodes. The table has a different entry for the different qualities of service available. The optimum route between nodes is used unless the optimum route does not have enough bandwidth to support the call, in which case the next best route is chosen. The routing table is built using a distributed, asynchronous adaptation of the Bellman-Ford shortest path algorithm. This algorithm constructs the set of shortest paths by first examining the shortest paths to each node traversing at most one edge, then examining the shortest paths traversing at most two edges, continuing until the shortest paths traversing at most $N-1$ edges have been found (N is the number of switches in the network). At this point, the shortest paths to all other nodes are known. In the ATM model, the link costs are not constant, they are a function of the available bandwidth. The less bandwidth there is available on a link, the higher the cost of that link. At startup, each node broadcasts information about itself across all its interfaces.

The ATM management module processes all PVC signaling and routing messages. It also allocates VCCs for each call routed through the node. This includes allocating VCCs for calls originating or destined for the node and also includes calls that are just switched through the node.

The ATM switch module is a fast packet switch model which switches outgoing ATM cells. It receives cells from the ATM transport module, which are cells that arrive at the node but are destined for another node. The switch also receives cells from the ATM layer, which are cells that originated at this node. The switch looks at the VPI/VCI values and places the cells in the corresponding output buffer. Each connection has its own output buffer.

The ATM transport module receives cells that arrive at the switch from other nodes. This module determines whether the cells are destined for this node or if they are for another node. If the cells are for another node, then the transport module assigns a new VPI/VCI value to the cell and sends them to the switch module to be routed to the next node in the path. If the cells are destined for this node and are control cells, then they are sent to the management module to be processed. If they are data cells destined for this node, then they are sent to the ATM Layer to be

processed by this switch. When the data cells arrive, regardless of whether they are destined for this node or for another node along the path, the transport module checks to see if the cells conform to the peak cell rate agreed on at connection set up time. If they do not conform, then the usage parameter control (UPC) algorithm will drop the cells. This UPC function implements a leaky bucket algorithm, GCRA, which also conforms to UNI 3.1.

The network model was setup with each switch represented by the previously described ATM switch model. Each LAN source in the ATM network was modeled as a traffic generator. The switches were interconnected with 155 Mbps links, on which the bit error rate and propagation delay could be specified. This model represents the minimum amount of detail that could be modeled and still predict the performance impact of adding and removing individual PVCs. The individual cards in the switch were not modeled, nor was the physical layer of the protocol stack.

3.5 Limitations of Simulating Large, High Speed Networks

The simulation described in Section 3.4 is event driven, as are most commercially available simulation packages. This means that every cell that is passed through a switch raises several events which must be handled. The overhead that is attributed to handling an event is significant, and is one of the main sources of problems with the simulation. When cells are generated at high bit rates, such as 155 Mbps, this results in a large number of events that must be processed. Many of the details of the simulation model described in Section 3.4, such as the routing, the leaky bucket policing method and the quality of service provisions were implemented but were removed to try and speed up the simulation. Even with a far less detailed model, the simulation still yielded a large number of events that need to be processed. Combining the number of events generated per cell with the large number of cells that are required to be generated in an ATM simulation produces a substantial execution time associated with the simulation.

As a result of each cell raising several events in a switch, and the large number of cells generated by an ATM network, the simulation executes very slowly. On the Bell ATM Network with 30 nodes and 25 PVCs, and each source node generating 10000 cells per second, it took

approximately 500 seconds (8.5 minutes) to simulate one second of real traffic. These measurements were taken from a Sparc 10 workstation. This translates into a network capability of 600 cells per second. This value was calculated by the equation

$$(30 \text{ nodes} * 10000 \text{ cells / sec on each node}) / 500 \text{ seconds simulation} = 600 \text{ cells/second}$$

At this rate it would take approximately 105 days to simulate 5 minutes of traffic on this network, which is only generating 10 000 cells/second. In order to fully congest the 155 Mbps links, the nodes would need to generate 365 000 cells/second on each link. This would cripple the simulation. The simulation was repeated on a Sparc 1000, resulting in a fifty percent decrease in processing time. However, even on a Sparc 1000, it would take fifty days to simulate five minutes of traffic.

These large processing times render this simulation inappropriate for use on a large size ATM network. Waiting fifty days for the results of adding or removing a PVC, or increasing the capacity of a link is useless. If a Bell customer wanted increased bandwidth, they would certainly want to be given the bandwidth within fifty days. A possible way to avoid this delay is to isolate the simulation to a small portion of the network path that is expected to be affected by the change made. The simulation would then be applied only to the selected portion. This would reduce the delay to an acceptable level in the current network, however the network size is expected to grow to several hundreds of nodes within the next year. Consequently, each switch will service hundreds of PVCs. Isolating a portion of this large network would result in a network the size of the one used in the current simulation, which is too slow.

In order to improve the speed of this type of event driven simulation to a reasonable response time, the simulation needs to be multitasked, so that independent events can be executed in parallel. Regardless of how fast the CPU is, the fact that each cell generates multiple events and processing each event takes a significant time, requires a multiprocessor simulation to provide faster results. Section 3.6 describes ways that a simulation can be distributed.

An alternative method that could reduce the execution time of the simulation is proposed in

[10]. Instead of distinguishing each individual cell in the system, the cells could be modeled using a fluid flow approach. In this type of model, traffic is not dealt with in individual units, it is viewed as a stream of fluid. A fluid simulation would assume that the fluid flow remains roughly constant over long time periods. Traffic fluctuations are modeled by events which represent a change in the flow rate. [10] Since changes in fluctuations would be expected to occur much less frequently than individual cell arrivals, the computing time can be reduced. However, by using the fluid flow approach to model the traffic, it becomes difficult to model any of the ATM congestion control and call policing mechanisms which deal with individual cells. The fluid flow concept is also fairly complex to model, and most models assume that sources are bursty and use on-off sources based on constant rate, Markov modulated processes [10].

Another problem encountered when simulating a large ATM network with hundreds of PVCs per switch is the readability of the model. Since each PVC is being modeled and altered in the simulations, it becomes very difficult to have this many PVCs on the screen. With the graphical interfaces of most commercial simulators, the intention is to have 'drag and drop' capabilities. To add a link, you just draw it in. However when the simulation becomes very large, it is difficult to handle all of the links on the screen.

3.6 Distributed Simulation

One solution to the processing time problem encountered in this experiment is to execute the simulation in a distributed fashion. There are several different ways that a simulation can be decomposed to be executed on multiple processors. These methods include using parallelizing compilers, distributing experiments, using distributed language functions and distributed events. These methods are described below. [22]

The first method that can be used to decompose a simulation is to use a parallelizing compiler to find pieces of the simulation code that can be executed in parallel on separate processors. A benefit of this method is that the compiler is invisible to the user, and does not

require user interaction to produce the results. Unfortunately, the compilers do not recognize parallelism that is inherently in existence due to the design of the simulation. [22]

Another way to distribute a simulation is to execute independent runs of a simulation on several processors and then average the result. This works well in systems which reach steady state quickly and which have relatively long runtimes. Alternatively, simulations could be run at the same time, each on a different processor, each with a different set of configuration parameters. This accelerates total simulation time, but does not decrease the time required to execute a single run of the simulation. [22]

Distributed languages can be used to assign different tasks to different processors. For example, one processor could handle random number generation, another could calculate ongoing statistics, etc. Distributed languages are widely used but do not exploit any inherent parallelism in the system being modeled.[22]

An alternate way to exploit the parallelism in a simulation is to distribute events to different processors. When an event is raised, it would go through a global list, which would allocate it to a specific processor. This method requires a system with shared memory so that the event list can be accessed by all of the processors in the system.

The main problem with a distributed simulation, as with any distributed software, is to ensure synchronization between the different processors. If the simulation is not distributed properly, the simulation run time may not improve much from the sequential execution run time. However, if the simulation can be distributed, then the run time can be accelerated, depending on the number of processors that the simulation is distributed over.

3.7 Discussion

Since the processing time required to execute the simulation is unacceptable, an attempt was made to scale the simulation down. The source of the problem is that each node is generating tens of thousands of cells per second and each cell raises five or six events on average that must be processed. If the model is scaled everywhere by a factor of K , then the number of cells generated

per second would be smaller, the bandwidths and link speeds available would be smaller and the buffer sizes in the switch would be smaller. Then, the simulation could be executed faster, since there would be less cells being generated.

A method to speed up the simulation is described in Section 4.8 and is based on the Q factor, which is described in the next chapter.

Chapter 4

Estimating Congestion Using the Q Factor

4.1 Data Description

The data analyzed in this project comes from two sources. The main set, used for the entire experiment, is data collected from Bell Sygma's ATM network. This network consists of 30 ATM switches interconnected across cities and provinces within Canada, with either 44 Mbps or 155 Mbps fiber optic links. This network connects many local area networks, with the traffic on the ATM network being mainly LAN extension traffic. The ATM network currently supports permanent virtual connections (PVCs) but does not support switched virtual connections (SVCs). There are on average, three to five PVCs per network link. Although the network is fairly lightly loaded, the traffic on it is representative of the traffic that is expected on ATM networks.

The number of cells transmitted and received on each line in the ATM network have been recorded at one hour time intervals. These hourly cell counts have been broken down into two data sets, one spanning a continuous six week period in the summer of 1995 and the other spanning a continuous four week period in the fall of 1995. All analysis performed on the summer set has been duplicated on the fall set, yielding similar results.

The other source of data used in this project is publicly available Ethernet traffic data from Bellcore. This data, which was described in section 2.2.1, has been shown to be statistically self similar. [19] Four different traces of this Ethernet data were used and each one contains the first one million arrivals of Ethernet packets. The files contain the timestamps (accurate to 10 microseconds) and lengths of the packets received in bytes. Two of the traces (pOct.TL and pAug.TL) contain all traffic on the Bellcore line and last for thirty minutes and sixty minutes respectively. The other two traces (OctExt.TL and OctExt4.TL) record only the external traffic between Bellcore and the rest of the world, and last for 35 hours and 21 hours respectively.

4.2 Definition and Meaning of the Q Factor

The goal of this project is to predict network congestion, based on the past experience of the network. It was noticed from the graphs of self similar data in [19] that the ratio of the mean traffic to the standard deviation is relatively stable with time [4]. This concept is the basis for the work in this thesis. The following sections show that this ratio can be used to classify traffic distributions, and ultimately to predict congestion in networks. The following definitions are used throughout.

S : the number of sampling intervals

C_{dj} : the number of cells counted in the j th sampling interval on device d . $j = 1..S$

M_d : the mean number of cells of C_{dj} over S sampling intervals for device d

$$M_d = (\sum_{j=1}^S C_{dj}) / S \quad V_d: \text{the variance in } C_{dj}$$

H_d : the standard deviation in C_{dj}

$$V_d = H_d^2$$

Q_d : the ratio of the mean to the standard deviation for device d

$$Q_d = M_d / H_d$$

Thus, the Q factor for any device in the network, whether a switch, router or line, is derived from the ratio of the mean traffic to the standard deviation. The higher the value of Q , the smaller the degree of variability in the traffic. The lower the value of Q , the larger the degree of variability. When dealing with congestion, the devices of greatest interest are the ones with very high means and the ones with large fluctuations in traffic. These large fluctuations in traffic correspond to small Q factors.

Since the data from the Bell network provided hourly counts of the number of cells received and transmitted on each line in the network, the means and variances for each of these lines was computed. From these means and variances, the Q values were calculated for several lines in the network. Table 4.1 lists the means, standard deviations and Q values for some lines in the network for both the fall and summer data sets. The mean and standard deviations are taken

over the entire six week period for the summer data, and over the four week period for the fall data. The lines listed in Table 4.1 do not comprise all of the lines in the network, but are the ones that remained up and functioning over the entire summer and fall period. Many of the experiments described in this thesis use sampling intervals of one day, or several days. In these cases, there were many more lines available that could be used for data, since many lines were up for the majority of the data sets.

It is evident from Table 4.1 that several of the lines in the Bell ATM network have traffic characteristics that are stable across both the summer and fall periods. The intent of Table 4.1 is to demonstrate that many lines in the network do have stable traffic distributions over the data collection periods, supporting the premise that data on a network can be stable over time. Line numbers 1, 3, 11, 12, 14, 16 and 17 all have Q values that are almost the same for the summer and fall periods. The remaining lines have quite different Q values for the summer and fall sets, but in all cases the means and/or standard deviations differ significantly. These lines with different Q values have different traffic characteristics due to an increase in the number of customers (and hence an increase in traffic load) between the summer and fall period. The premise that is made in this thesis is that if there have been no additions or removals of traffic on a network then the distributions will be stable over time. Therefore, the lines that have different Q values in the fall and summer sets all have increased traffic loads, so their distribution changes can be explained by the change in load.

One of the requirements for the use of the congestion prediction technique described in this thesis is that the traffic distribution is stable across the data collecting period. This does not mean that the traffic can not fluctuate differently during different times of the day, it just means that across a larger time scale the traffic is stable. If a network designer were to use the methods described in this thesis, they would know when new customers are being added or removed from a transmission line, and would therefore use data from before or after the customer was added or removed.

Line Number	Summer Mean	Summer Deviation	Summer Q Value	Fall Mean	Fall Deviation	Fall Q Values
1	374683	629232	0.60	333339	526580	0.63
2	421802	1873435	0.23	68421	93322	0.73
3	372851	636224	0.59	344239	549480	0.63
4	441452	1915163	0.23	68722	92925	0.74
5	241557	39644	6.09	245010	251718	0.97
6	152031	463584	0.33	78173	107741	0.73
7	223233	1406215	0.16	389400	1253310	0.31
8	8970	1381	6.50	7911	6194	1.28
9	338633	31395	10.8	922730	1399386	0.66
10	7877	945	8.44	8024	1280	6.27
11	22378	9078	2.47	18064	7890	2.29
12	784932	170738	4.60	802811	185836	4.32
13	47394	245342	0.19	329131	254686	1.29
14	98362	109536	0.90	100937	123094	0.82
15	453918	201827	2.25	102837	99923	1.03
16	63749	500231	0.13	70537	440856	0.16
17	19733	15823	1.25	18533	14147	1.31

Table 4.1 Means, Standard Deviations and Q Values of all Network Lines in Summer and Fall

4.3 Steps Necessary to Predict Congestion Using the Q Factor

In order to show that the Q factor can be used to predict network congestion, it must be shown that:

1. The traffic distributions of devices in the network are stable over time. This is done in Section 4.4 for the Bell ATM network by plotting the histogram of the distribution of the same line for a four week period in the summer and a four week period in the fall. The histograms are similar, showing that distributions in this network that do not have a known increase or decrease in customer load do not vary with time.

2. Q factors give you an indication of the traffic distribution. This is done in section 4.5 by showing that similar Q values calculated from the Bell network have similar traffic distributions. As a result, given a Q value you can then map it to a traffic distribution and determine the percentage time that the traffic is above a certain congestion level.

3. Regardless of the sampling interval used to measure the means and variances of the traffic, an accurate value of Q can be found. Section 4.6 shows that Q varies slowly with different

sampling intervals for self similar traffic. This is demonstrated by analyzing the Bellcore data for sampling intervals of 0.01, 0.1, 1.0, 10.0 and 100.0 seconds.

4. There is a length of time required to sample the data for in order to get an accurate Q value. Section 4.7 shows that Q varies with the time of day, so in order to measure the Q value for a device, you need to sample for longer than a day. This section shows that for the Bell network data, the sampling length must be longer than four days.

The above process allows the prediction of performance for any device in a network for which you can measure the Q factor. In the Bell ATM network, since we have calculated the means and variances of each line in the network, the performance and congestion of each line can be predicted. However, it is also necessary to predict the performance of the switches in this network and the actual means and variances of the switches was not collected. Chapter 5 shows that given Q factors for all incoming or outgoing lines of a switch, the Q factor of the switch can be calculated. As a result, the percentage time the switch will be congested can be predicted. This method can be extended to any device in the network, not just switches. If you had the means and variances for all PVCs on a line, then you could calculate the Q factor for that line.

4.4 Determining that Traffic Distributions are Stable Over Time

One of the underlying assumptions in this project is that the traffic distribution of the individual lines and devices in the network does not change over the period of time being analyzed. For the Bell ATM network the period of time analyzed was ten weeks. If this assumption is false, then the traffic distributions change over the time period and a Q value can not be associated with a particular line.

In order to show that the Bell ATM network is composed of lines whose distributions are stable over time, several lines in the network were analyzed. In Table 4.1, it was demonstrated that the Q value in the summer and fall were the same for several lines in the network. This section shows that the actual traffic distributions for these lines are similar during the summer and fall.

Not all of the lines analyzed had similar distributions, but this is due to the fact that the amount of traffic on the line had been increased or decreased between the summer and fall periods. The distribution of the traffic for a four week period in the summer data was determined and plotted as a histogram. The same technique was applied to a four week period in the fall data and this distribution was plotted as a histogram. If both the summer and fall histograms for a line are similar, then it can be concluded that line has a stable distribution for the time period used in this project.

The method used to produce the probability histograms is as follows:

1. The mean value and standard deviation of the traffic counts on a line for the summer and fall periods are calculated.
2. The traffic counts for the line using the summer data are normalized with respect to the summer mean. The traffic counts for the line using the fall data are normalized with respect to the summer mean. This allows comparison of the distribution in the summer and the fall because the same mean is used to normalize both sets of data. This normalization uses the process described in Section 2.3. The resulting traffic counts are a multiple of the standard deviation from the summer mean. This step is repeated using the fall mean. That is the summer data is normalized with respect to the fall mean and the fall data is also normalized with respect to the fall mean. This provides a verification that the distributions are the same, since both normalized distributions should look the same.
3. These normalized traffic counts are divided into intervals, and the number of values in each interval is counted. This produces a histogram which tells us the number of occurrences of cells in each interval.
4. The outcome of the first three steps is four distributions which are:
 - a) The distribution of the summer data for a line with the traffic counts normalized to the summer mean.
 - b) The distribution of the fall data for a line with the traffic counts normalized to the summer mean.
 - c) The distribution of the summer data for a line with the traffic counts normalized to the fall mean.
 - d) The distribution of the fall data for a line with the traffic counts normalized to the fall mean.

5. These four graphs are plotted side by side and are compared. If the distributions are the same then the distributions are stable. If they are different then the distributions can not be considered stable.

6. The graphs of the raw traffic counts for both the summer and fall period for the line are also graphed. These graphs provide an explanation for the cases in which the distributions do not appear to be constant. If the cell counts are significantly different, then traffic has been altered on the line (usually through the addition or removal of a customer).

These six steps described above have been repeated for eleven lines in the Bell ATM network. For each of the eleven lines, the four normalized graphs are shown along with the graphs of the raw cell counts (measured every hour) for both the summer and fall periods. Figures 4.1 to 4.33 depict these graphs. The graphs for lines 1, 2, 4, 6 and 11 show that the distributions for the summer and fall data normalized to the summer mean are both similar. Also, the distributions for the summer and fall data normalized to the fall mean are also similar. Therefore, the distribution is stable over the ten week period. The distribution graphs for the remaining lines are not similar. However, when the graphs of the raw traffic counts are compared for the summer and fall data sets it is evident that the traffic has been changed on these lines between the summer and fall period.

From these graphs it is apparent that not every line in the network has a distribution that is stable over time. However many of the lines do have stable distributions, and these lines can be used in this thesis for analysis.

Figure 4.1 Line 1 Summer and Fall Distributions Normalized to Summer Means

Figure 4.2 Line 1 Summer and Fall Distributions Normalized to Fall Means

Figure 4.3 Line 1 Summer and Fall Hourly Cell Counts

Figure 4.4 Line 2 Summer and Fall Distributions Normalized to Summer Means

Figure 4.5 Line 2 Summer and Fall Distributions Normalized to Fall Means

Figure 4.6 Line 2 Summer and Fall Hourly Cell Counts

Figure 4.7 Line 3 Summer and Fall Distributions Normalized to Summer Means

Figure 4.8 Line 3 Summer and Fall Distributions Normalized to Fall Means

Figure 4.9 Line 3 Summer and Fall Hourly Cell Counts

Figure 4.10 Line 4 Summer and Fall Distributions Normalized to Summer Means

Figure 4.11 Line 4 Summer and Fall Distributions Normalized to Fall Means

Figure 4.12 Line 4 Summer and Fall Hourly Cell Counts

Figure 4.13 Line 5 Summer and Fall Distributions Normalized to Summer Means

Figure 4.14 Line 5 Summer and Fall Distributions Normalized to Fall Means

Figure 4.15 Line 5 Summer and Fall Hourly Cell Counts

Figure 4.16 Line 6 Summer and Fall Distributions Normalized to Summer Means

Figure 4.17 Line 6 Summer and Fall Distributions Normalized to Fall Means

Figure 4.18 Line 6 Summer and Fall Hourly Cell Counts

Figure 4.19 Line 7 Summer and Fall Distributions Normalized to Summer Means

Figure 4.20 Line 7 Summer and Fall Distributions Normalized to Fall Means

Figure 4.21 Line 7 Summer and Fall Hourly Cell Counts

Figure 4.22 Line 8 Summer and Fall Distributions Normalized to Summer Means

Figure 4.23 Line 8 Summer and Fall Distributions Normalized to Fall Means

Figure 4.24 Line 8 Summer and Fall Hourly Cell Counts

Figure 4.25 Line 9 Summer and Fall Distributions Normalized to Summer Means

Figure 4.26 Line 9 Summer and Fall Distributions Normalized to Fall Means

Figure 4.27 Line 9 Summer and Fall Hourly Cell Counts

Figure 4.28 Line 10 Summer and Fall Distributions Normalized to Summer Means

Figure 4.29 Line 10 Summer and Fall Distributions Normalized to Fall Means

Figure 4.30 Line 10 Summer and Fall Hourly Cell Counts

Figure 4.31 Line 11 Summer and Fall Distributions Normalized to Summer Means

Figure 4.32 Line 11 Summer and Fall Distributions Normalized to Fall Means

Figure 4.33 Line 11 Summer and Fall Hourly Cell Counts

4.5 Predicting Traffic Distributions from Q Values

In order to use a devices' Q factor as an indicator of that devices' traffic characteristics, it must be shown that devices with similar Q values have similar traffic distributions. If they do, then you can associate a distribution to a Q value. Then, given a Q value, the percentage of time the device with that Q value spends above a certain threshold can be determined from looking at how often the distribution is above that threshold.

Knowing that Q values in the same range have similar distributions, the distributions of these similar Q values can be combined together to get a more accurate description of the distribution for that range. The steps taken to show that similar Q values have predictable distributions are as follows:

1. The Q values for a large number of lines in the Bell ATM network were computed from the C_{dj} , M_d and H_d of each line. Each Q value was computed from twenty four samples taken at one hour sampling intervals ($j = 1$ hour).
2. For each line, the traffic count values (C_{dj}) were normalized using the standard normalization equation that was described in Section 2.4

$$Z_{dj} = C_{dj} - M_d / H_d \dots \dots \dots (4.1)$$

This produces traffic counts (Z_{dj}) that represent the number of standard deviations each count is from the mean count. By applying this normalization, the normalized traffic counts of different lines with similar Q values can then be combined together. This produces a more accurate distribution for that Q value than if a single line had been used. The assumption that combining normalized traffic counts produces more accurate distributions is made in [4].

3. For each range of Q factors, the normalized traffic counts of lines with a Q value in the given range were combined and the distributions found. To find the distributions, the range of values of traffic counts were partitioned into intervals. Then, the number of traffic count values in each interval were counted. This produces a histogram for each Q factor range.

4. Q ranges were aggregated together to produce larger Q ranges. This was done to produce a sufficient amount of data in each distribution to allow comparison of results from different traces of data. In this experiment, the aggregated result was three Q factor ranges: [0, 2.5), [2.5, 4.0), and [4.0, 5.0).
5. The histograms corresponding to each Q factor range from different data traces were compared to determine if the histograms are similar. If they are similar for each Q factor range, then the traffic distribution corresponding to a given Q factor is predictable and stable. If the histograms are different, then there is no mapping between a line with a particular Q value and that lines' traffic distribution.

4.5.1 Bell ATM Network Distribution Observations

The five steps described in Section 4.5 that are required to show that similar Q values have similar distributions were applied to two different traces of data from the Bell ATM network. One set was the six week long trace from the summer of 1995, and the other set was the four week long trace from the fall of 1995. These data sets are described in section 4.1.

The Q values for 118 lines in the summer set were computed as were the Q values for 120 lines in the fall set. Each Q value was calculated based on a one day sampling length, with a one hour sampling interval. The lines used in the summer set were not the exact same lines used in the fall set, but there was some overlapping. That is, some lines used in the summer set were also used in the fall set, and some other lines used in the fall set were not used in the summer set. Similarly, the Q values were calculated on different days within the trace, they were not all computed from the same days data. Table 4.2 lists the number of lines in the network that had Q values in the corresponding ranges for both the summer and fall set. For example, during the summer, there were 8 lines that had a Q value between 0.5 and 1.0. In the fall, there were 14 lines that had a Q value between 0.5 and 1.0.

Q Value Range	# of lines in summer set	# of lines in fall set
0 - 0.5	8	6
0.5 - 1.0	8	14
1.0 - 1.5	9	13
1.5 - 2.0	5	12
2.0 - 2.5	12	10
2.5 - 3.0	10	10
3.0 - 3.5	12	13
3.5 - 4.0	18	10
4.0 - 4.5	14	13
4.5 - 5.0	22	19

Table 4.2 Q Value Ranges for Summer and Fall Data Sets

The traffic count values for each line were then normalized using Equation 4.1 and the normalized traffic counts were combined for lines with similar Q values. These combined, normalized values were then partitioned into intervals, where each interval represents the number of standard deviations the count is away from the mean count. In the Bell data, the normalized traffic counts ranged from 5 down to -2. This corresponds to the maximum value being 5 standard deviations above the mean and the minimum value being 2 standard deviations below the mean. This implies that the fluctuations above the mean are more pronounced than the fluctuations below the mean [1]. Table 4.3 shows the number of values that occurred in each interval for 10 different Q ranges in the summer data set. Table 4.4 shows the same information for the fall data set.

Interval	Q = 0 - 0.5	Q = 0.5- 1.0	Q = 1.0 - 1.5	Q = 1.5- 2.0	Q = 2.0 - 2.5	Q = 2.5- 3.0	Q = 3.0 - 3.5	Q = 3.5- 4.0	Q = 4.0 - 4.5	Q = 4.5- 5.0
[5,4]	6	2	2	1	1	0	0	0	0	0
(4,3]	1	4	1	1	4	1	2	0	0	0
(3,2]	4	2	7	4	6	4	0	3	5	0
(2,1]	2	19	21	10	20	9	6	2	8	0
(1,0.9]	0	3	4	1	6	7	3	14	0	0
(0.9,0.8]	2	0	0	1	7	12	5	10	1	0
(0.8,0.7]	1	2	3	2	6	6	4	2	3	1
(0.7,0.6]	3	5	3	2	6	10	15	7	4	1
(0.6,0.5]	0	1	2	2	1	5	28	20	5	1
(0.5,0.4]	2	1	2	5	3	13	49	29	3	1
(0.4,0.3]	2	0	1	3	5	19	20	22	6	4
(0.3,0.2]	0	3	8	1	3	18	8	20	8	13
(0.2,0.1]	2	4	7	1	10	8	14	5	15	33
(0.1,0]	2	6	6	1	3	10	0	19	25	198
(0,-0.1]	2	5	5	1	5	5	11	26	53	180
(-0.1,-0.2]	4	4	7	4	6	6	18	32	116	59
(-0.2,-0.3]	78	8	19	23	98	26	6	39	41	14
(-0.3,-0.4]	43	36	27	6	10	6	7	32	15	1
(-0.4,-0.5]	31	21	11	13	18	4	3	36	12	0
(-0.5,-0.6]	0	9	28	6	8	6	5	22	0	0
(-0.6,-0.7]	0	14	9	5	16	18	5	9	0	0
(-0.7,-0.8]	0	35	16	8	10	4	8	11	0	0
(-0.8,-0.9]	0	0	2	10	9	4	5	12	0	0
(-0.9,-1]	0	0	2	0	2	2	9	8	0	0
(-1,-1.1]	0	0	4	0	1	2	14	5	0	0
(-1.1,-1.2]	0	0	4	1	0	6	21	7	0	0
(-1.2,-1.3]	0	0	6	1	6	2	0	1	0	0
(-1.3,-1.4]	0	0	0	2	2	8	6	0	0	0
(-1.4,-1.5]	0	0	0	0	3	0	0	0	0	0
(-1.5,-1.6]	0	0	0	0	0	4	2	0	0	0
(-1.6,-2.5]	0	0	0	0	0	5	0	0	0	0

Table 4.3 Distribution Breakdown for Lines in Different Q Ranges for the Summer Data

Interval	Q = 0- 0.5	Q = 0.5- 1.0	Q = 1.0 - 1.5	Q = 1.5- 2.0	Q = 2.0 - 2.5	Q = 2.5- 3.0	Q = 3.0 - 3.5	Q = 3.5- 4.0	Q = 4.0 - 4.5	Q = 4.5- 5.0
[5,4]	6	4	2	5	0	0	0	0	0	0
(4,3]	0	5	4	4	1	4	4	2	0	0
(3,2]	0	10	12	5	8	3	2	8	2	0
(2,1]	0	13	19	14	19	15	5	1	6	1
(1,0.9]	0	4	3	3	3	3	6	0	2	0
(0.9,0.8]	0	1	3	1	4	3	5	1	2	1
(0.8,0.7]	0	2	7	1	5	3	6	2	4	0
(0.7,0.6]	0	8	5	3	3	3	17	5	4	0
(0.6,0.5]	0	4	5	0	13	5	11	3	8	0
(0.5,0.4]	0	11	9	4	7	6	14	1	22	1
(0.4,0.3]	0	5	5	6	6	8	9	3	15	2
(0.3,0.2]	0	12	1	1	12	9	7	3	14	21
(0.2,0.1]	2	9	2	3	8	11	11	3	7	32
(0.1,0]	0	14	4	3	11	25	7	16	15	131
(0,-0.1]	0	23	3	4	4	19	29	6	60	192
(-0.1,-0.2]	13	7	7	18	6	2	54	101	56	48
(-0.2,-0.3]	115	34	46	127	10	12	31	50	23	2
(-0.3,-0.4]	0	8	58	19	11	29	17	16	17	3
(-0.4,-0.5]	0	34	16	25	10	10	5	1	26	2
(-0.5,-0.6]	0	38	24	4	20	9	3	0	4	1
(-0.6,-0.7]	0	39	16	6	15	12	10	2	2	0
(-0.7,-0.8]	0	21	23	0	25	17	4	2	5	0
(-0.8,-0.9]	0	6	8	1	10	1	3	1	0	1
(-0.9,-1]	0	5	9	5	2	5	4	0	1	0
(-1,-1.1]	0	1	6	8	1	2	4	1	2	0
(-1.1,-1.2]	0	0	0	3	3	1	0	1	0	0
(-1.2,-1.3]	0	0	0	0	2	4	10	0	1	0
(-1.3,-1.4]	0	0	0	0	4	2	4	0	0	0
(-1.4,-1.5]	0	0	0	0	1	2	1	0	0	0
(-1.5,-1.6]	0	0	0	0	2	1	2	0	0	0
(-1.6,-2.5]	0	0	0	0	0	0	0	0	0	0

Table 4.4 Distribution Breakdown for Lines in Different Q Ranges for the Fall Data

The data in Tables 4.3 and 4.4 were aggregated into intervals of size one, meaning each interval is one standard deviation further from the mean than the previous interval. This aggregated data is shown in Table 4.5 for the summer set and Table 4.6 for the fall set.

Interval	Q = 0 - 0.5	Q = 0.5 - 1.0	Q = 1.0 - 1.5	Q = 1.5 - 2.0	Q = 2.0 - 2.5	Q = 2.5 - 3.0	Q = 3.0 - 3.5	Q = 3.5 - 4.0	Q = 4.0 - 4.5	Q = 4.5 - 5.0
[5,4]	6	2	2	1	1	0	0	0	0	0
(4,3]	1	4	1	1	4	1	0	0	0	0
(3,2]	4	2	7	4	6	4	0	3	5	0
(2,1]	2	19	21	10	20	9	6	2	8	0
(1,0]	14	25	36	19	50	108	146	148	72	252
(0,-1]	158	132	126	76	182	81	77	227	237	254
(-1,-2]	0	0	14	4	112	25	43	13	0	0
(-2,-3]	0	0	0	0	0	2	0	0	0	0
(-3,-4]	0	0	0	0	0	0	0	0	0	0
(-4,-5]	0	0	0	0	0	0	0	0	0	0

Table 4.5 Aggregated Distribution Breakdown for Lines in Different Q Ranges - Summer Data

Interval	Q = 0 - 0.5	Q = 0.5 - 1.0	Q = 1.0 - 1.5	Q = 1.5 - 2.0	Q = 2.0 - 2.5	Q = 2.5 - 3.0	Q = 3.0 - 3.5	Q = 3.5 - 4.0	Q = 4.0 - 4.5	Q = 4.5 - 5.0
[5,4]	6	4	2	5	0	0	0	0	0	0
(4,3]	0	5	4	4	1	4	4	2	0	0
(3,2]	0	10	12	5	8	3	2	8	2	0
(2,1]	0	13	19	14	19	15	5	1	6	1
(1,0]	2	70	44	25	72	76	93	37	93	188
(0,-1]	128	215	210	209	113	116	160	179	194	249
(-1,-2]	0	1	6	11	13	12	21	2	3	0
(-2,-3]	0	0	0	0	0	0	0	0	0	0
(-3,-4]	0	0	0	0	0	0	0	0	0	0
(-4,-5]	0	0	0	0	0	0	0	0	0	0

Table 4.6 Aggregated Distribution Breakdown for Lines in Different Q Ranges - Fall Data

The fourth step of the process was then performed, the aggregation of Q ranges. The resultant data has three ranges of Q values, which is shown in Tables 4.7 and 4.8. The data in these tables correspond to the histograms for these Q ranges. Plotting the data for each range of Q values from the summer and fall sets on the same graph allows comparison of the distributions. These graphs are shown in Figures 4.34 to 4.36. Figure 4.34 plots the summer and fall distributions for a Q value less than 2.5. Figure 4.35 plots the summer and fall distributions for a Q value between 3.0 and 4.0. Figure 4.36 plots the summer and fall distributions for a Q value between 4.5 and 5.0. If the distributions are similar for the graphs of each Q range, then the distribution is constant and predictable for a given Q range. It is evident from these graphs that the distributions are similar.

Interval	Q = 0 - 2.5	Q = 3.0 - 4.0	Q = 4.5 - 5.0
[5,4]	12	0	0
(4,3]	11	1	0
(3,2]	23	7	5
(2,1]	72	17	8
(1,0]	146	402	324
(0,-1]	674	385	491
(-1,-2]	30	81	0
(-2,-3]	0	2	0
(-3,-4]	0	0	0
(-4,-5]	0	0	0

Table 4.7 Aggregated Q Ranges for Summer Data Set

Interval	Q = 0 - 2.5	Q = 3.0 - 4.0	Q = 4.5 - 5.0
[5,4]	17	0	0
(4,3]	14	10	2
(3,2]	35	13	10
(2,1]	65	21	7
(1,0]	213	206	130
(0,-1]	875	455	373
(-1,-2]	31	35	5
(-2,-3]	0	0	0
(-3,-4]	0	0	0
(-4,-5]	0	0	0

Table 4.8 Aggregated Q Ranges for Fall Data Set

Figure 4.34 Traffic Distribution of Lines with Aggregated Q Factor ≤ 2.5

Figure 4.35 Traffic Distribution of Lines with Aggregated Q Factor from 3.0 to 4.0

Figure 4.36 Traffic Distribution of Lines with Aggregated Q Factor Larger Than 4.0

The Chi-squared test was performed on the data from Tables 4.5 and 4.6 to show mathematically how consistent the two distributions are. The Chi-Square test uses the following formula to calculate the Chi-Square statistic, χ^2 . R_i represents the number of events in bin i for Table 4.5. S_i represents the number of events in bin i for Table 4.6. A large value for χ^2 implies that the two data sets do not come from the same distribution. Table 4.9 gives the results from

applying the Chi-squared test to each Q Interval in Tables 4.5 and 4.6. The number of degrees of freedom, r is the number of bins the data is divided into minus 1.

$$\chi^2 = \sum_i \frac{(\sqrt{S/R} Ri - \sqrt{R/S} Si)^2}{R_i + S_i} \dots\dots\dots(4.2)$$

$$R \equiv \sum_i Ri \quad S \equiv \sum_i Si$$

Q Interval	Degrees of Freedom (r)	Chi-Square Statistic(χ^2)	Significance probability (α)
0 - 0.5	5	11.194	0.05
0.5 - 1.0	6	14.547	0.025
1.0 - 1.5	6	12.598	0.05
1.5 - 2.0	6	8.259	0.20
2.0 - 2.5	6	68.269	0.0001
2.5 - 3.0	6	21.794	0.001
3.0 - 3.5	6	54.198	0.0001
3.5 - 4.0	6	44.819	0.0001
4.0 - 4.5	4	10.621	0.025
4.5 - 5.0	2	5.488	0.05

Table 4.9 Chi-Squared Analysis Between Table 4.5 and Table 4.6

It can be seen from Table 4.9 that for many of the Q Intervals, the hypothesis that the distribution from the fall data set is consistent with the distribution from the summer data set is accepted with a significance level of at least 0.025. The intervals 0 to 0.5, 0.5 to 1.0, 1.0 to 1.5, 4.0 to 4.5 and 4.5 to 5.0 all have a significance level of at least 0.025. For the intervals in the range 2.0 to 4.0, the significance levels are lower than 0.025. This implies that the distributions between the fall and summer data set are not the same for those specific intervals. The intervals that are of most concern are the low Q values (since they produce the largest fluctuations in traffic), and the low Q values have small Chi-squared statistics'. Therefore, these are the ones that need to have consistent distributions, and they do.

It can be concluded from these graphs and the Chi-squared analysis of the fall and summer data that the traffic distribution for a particular Q value is predictable [4]. If the distribution of traffic for a device is known, then the percentage of time that the device will be busier than a certain

level can be determined. This is done by calculating the area between the graph and that level. If this level is the congestion point of the device, then the calculation gives the percentage of time the device is congested.

In these experiments the data was aggregated together to produce large enough numbers to analyze. However, to deploy this technique in practice, a much larger set of data would be needed so that such wide ranges of Q values would not have to be aggregated together to determine the distribution. Ideally, there would be enough data available for devices with the same Q value and a specific Q value could be looked up, instead of a range of Q values.

A practical implementation of mapping Q factors to their traffic distributions is described in Chapter 6.

4.6 Relationship Between Q Factors and Sampling Intervals

It is necessary to know the appropriate sampling interval to sample a device at, since predicted rates of congestion for a device would be less accurate if made at intervals very different from its buffer processing time [4]. In order to use the Q factor as an indicator of traffic distribution, it is necessary to know the relationship between Q and the sampling interval used to measure the traffic counts. If the Q value varies drastically depending on the length of the sampling interval, then this implies that the Q value is not as stable an indicator of the traffic. However, if Q varies slowly with the change in sample length, then Q is a stable traffic indicator.

If the relationship between Q and the sampling interval is known, then predictions on traffic distributions can be made for a sampling interval different from the measured sampling interval. This section describes the relationship between Q and the sampling interval used for self similar traffic. Since self similar traffic accurately represents LAN and variable bit rate traffic and is the type of data running on the Bell ATM network, it is appropriate to study this class of traffic. Distributions other than self similar will have different relationships between the Q factor and the sampling interval than self similar distributions do.

It has been shown in [19] that the variance of self similar traffic at a desired sampling

interval can be determined by extrapolating or interpolating from the variance calculated at a different sampling interval. This is done using an exponential fit to the points. This is demonstrated mathematically by the equation

$$\text{variance}(X) \sim t^{-B} \dots\dots\dots(4.2)$$

where t is the sampling interval and B is the auto correlation function.

Since Q is the ratio of the mean to the square root of the variance, then Q values measured at given sampling intervals in self similar distributions can be extrapolated or interpolated in the same exponential way as Equation 4.2 to find the Q value at a different sampling interval.

4.6.1 Q Factors for Various Sampling Intervals on Bellcore Data

The Bellcore Ethernet data that was described in Section 4.1 and used in [2] to show that the variance varies slowly with time was used to confirm the concept that Q varies slowly with time, for self similar traffic. The Q values for different traces of the Bellcore data were calculated for five different sampling intervals: 0.01, 0.1, 1.0, 10.0 and 100.0 seconds. Table 4.10 shows the Q values computed using the five different sampling intervals on the four traces of data. It is evident that for all four traces, Q increases slowly as the sampling interval lengthens. This demonstrates that Q values for self similar traffic are stable with time. As a result, the more stable Q is with time, the more accurate the predictions will be for sampling intervals which are far from the measured sampling interval [4].

Sampling Interval	pOct.TL Trace Q Value	pAug.TL Trace Q Value	OctExt.TL Trace Q Value	OctExt4.TL Trace Q Value
0.01 seconds	0.979	0.621	0.139	0.164
0.1 seconds	1.83	1.06	0.278	0.187
1.0 seconds	2.36	1.61	0.403	0.193
10 seconds	2.83	2.40	0.509	0.198
100 seconds	3.08	3.46	0.642	0.214

Table 4.10 Q Values for Different Ethernet Traces using Different Sampling Intervals

4.7 Relationship Between Q Factors and Sampling Length

In order to make accurate congestion predictions using the Q value, the number of samples required to calculate Q accurately must be determined. This can be done by sampling the same data for different lengths of time. In this project, the sampling length was experimented with using both the Bellcore Ethernet data and the Bell ATM Network data.

4.7.1 Bellcore Data Results

In the Bellcore data, two of the traces last for 24 and 34 hours respectively. The Q values for these traces were calculated at one second sampling intervals at different sampling lengths, which ranged from one minute (60 samples) to ten hours (36000 samples). For the 34 hour long trace, when sixty samples were collected, approximately 2000 Q values could be calculated over the 34 hours. Similarly, when 36000 samples were collected, three Q values could be calculated over the 34 hours. Figures 4.37 to 4.42 show the calculated Q values for the 24 hour trace OctExt4 at different sampling lengths. All samples were collected at one second sampling intervals. Figure 4.37 shows the smallest sampling length used, one minute. Figure 4.42 shows the longest sampling interval used, ten hours.

It is apparent from looking at these graphs that the value of Q changes at different periods of the day. This implies that you need to sample for longer than one day to get an accurate estimate of Q. If you were to use a small sampling length, such as one minute, it is clear from Figure 4.37 that you would get a different value for Q depending on which minute in the day was chosen to sample.

Figures 4.43 to 4.48 shows the Q values calculated at different sampling lengths for the second Bellcore trace, OctExt, which lasted for 34 hours. The same conclusions can be made from these graphs as was made from the first Bellcore trace. Q values vary over a day, hence the required sampling length is longer than one day. Since the longest trace available to us was the 36 hour trace OctExt, the exact sampling length required for this data could not be calculated.

Figure 4.37 Q Values for OctExt4 Trace, 60 Samples per Q Value

Figure 4.38 Q Values for OctExt4 Trace, 300 Samples per Q Value

Figure 4.39 Q Values for OctExt4 Trace, 600 Samples per Q Value

Figure 4.40 Q Values for OctExt4 Trace, 1800 Samples per Q Value

Figure 4.41 Q Values for OctExt4 Trace, 7200 Samples per Q Value

Figure 4.42 Q Values for OctExt4 Trace, 36000 Samples per Q Value

Figure 4.43 Q Values for OctExt Trace, 60 Samples per Q Value

Figure 4.44 Q Values for OctExt Trace, 300 Samples per Q Value

Figure 4.45 Q Values for OctExt Trace, 600 Samples per Q Value

Figure 4.46 Q Values for OctExt Trace, 1800 Samples per Q Value

Figure 4.47 Q Values for OctExt4 Trace, 7200 Samples per Q Value

Figure 4.48 Q Values for OctExt Trace, 36000 Samples per Q Value

4.7.2 Bell ATM Network Data Results

Since the Bell ATM network data spans a ten week period, there is enough data available to calculate the required sampling length for this data. The same process that was applied to the Bellcore data to analyze the relationship between Q and the sampling length was applied to the Bell data. The Q values for fifteen different lines in the network were calculated at one hour sampling

intervals for three different sampling lengths. These sampling lengths are 1 day, 2 days and 4 days, which correspond to 24, 48 and 96 samples respectively. If the sampling length is too short then graphs of Q values calculated for that sampling length at different intervals in the trace will fluctuate. That is, the Q value is not stable. If the Q value is not stable, then the value of Q will be different depending on the time of day that the sample is taken. However, if the sampling interval is long enough, then Q values should be stable across the trace. This means that the value that is calculated for Q will be independent of what time in the trace the sample was taken at.

Figures 4.49 to 4.51 show the Q values calculated for sampling lengths of one day, two days and 4 days for line 1 in the ATM network. The x-axis represents the day the Q value was calculated for. In Figure 4.49, which has 24 samples per Q value, the Q values are scattered between 1.0 and 2.7. From this it is concluded that one day is too short a sampling length, since the Q values are not stable. Figure 4.50, which corresponds to 48 samples per Q value shows a graph that is flatter than the one in Figure 4.49, but still has some fluctuation in it. Figure 4.51, which has 96 samples per Q value, shows that a sampling length of 4 days is enough to get an accurate estimated of Q. Since the graph is relatively constant, regardless of which four days over the twenty five day period the Q factor is measured for, the result will be similar.

Figures 4.52 to 4.59 repeat the process on three other lines in the network and strengthen the premise that four days is an appropriate sampling length (96 samples) for the Bell ATM network. Each of these sets of graphs show scattered values of Q when only twenty four samples are used, a flatter graph when forty eight samples are used, and a stable graph when ninety six samples are used.

This process of experimenting with sampling lengths in order to determine an accurate Q value can be applied on any data set, to ensure that the sampling length being used to measure Q is not too short.

Figure 4.49 Q Values for Line 1, Bell Network, 24 Samples per Q Value

Figure 4.50 Q Values for Line 1, Bell Network, 48 Samples per Q Value

Figure 4.51 Q Values for Line 1, Bell Network, 96 Samples per Q Value

Figure 4.52 Q Values for Line 2, Bell Network, 24 Samples per Q Value

Figure 4.53 Q Values for Line 2, Bell Network, 48 Samples per Q Value

Figure 4.54 Q Values for Line 2, Bell Network, 96 Samples per Q Value

Figure 4.55 Q Values for Line 3, Bell Network, 24 Samples per Q Value

Figure 4.56 Q Values for Line 3, Bell Network, 48 Samples per Q Value

Figure 4.57 Q Values for Line 3, Bell Network, 96 Samples per Q Value

Figure 4.58 Q Values for Line 4, Bell Network, 24 Samples per Q Value

Figure 4.59 Q Values for Line 4, Bell Network, 48 Samples per Q Value

It might seem that a sampling interval of one hour, which was the sampling interval used on the Bell ATM network, is too large a sampling interval to use. However, consider the graphs of self-similar traffic taken from the Bellcore study [19] each with a sampling interval 10 times larger than the previous. These graphs are shown in Figure 4.61a-e. The sampling intervals are in the range 0.01 seconds to 100 seconds. For the sampling interval of 0.01 seconds (graph e), the sample length is 10 seconds. For the sampling interval of 1 second (graph c), the sample length is 16.7 minutes. For the sampling interval of 100 seconds, (graph a) the sample length is 27 hours. Figure 4.60 is a graph from the Bell ATM network which also plots time vs packets/time unit and

uses a sampling interval of one hour. All of these graphs exhibit self similar properties and imply that the longer the sampling interval used, the longer the sample length needs to be. So for a sampling interval of 1 hour (3600 seconds), it is not surprising that a sample length of 4 days, which was the value calculated in this thesis for the Bell network, is needed to get a stable Q value.

Figure 4.60 Time vs Cell Counts for a Bell ATM Network line - Sampling Interval of 3600 Secs

Figure 4.60 Bellcore Self Similar Ethernet Data Using Different Sampling Intervals [19]

Chapter 5

Combining Q Values at a Device

5.1 Determining the Q Value for any Network Device

Chapter four presented evidence that the Q value can be used as a potential indicator of congestion in a network. The data that was used in this study from the Bell ATM Network consisted of traffic counts on each transmission line in the network. Thus, for each transmission line, a Q value can be calculated and associated with a traffic distribution, as described in section 4.3. Then, an estimate of the amount of time the traffic level is above the congestion point can be made from the traffic distribution. If data was available for the traffic counts within the switch, then the Q value for the switch can be calculated and the entire process repeated to determine how often the switch is expected to be congested. Similarly, this process could be used on individual PVCs or on any other network device such as a bridge or router. As long as the traffic counts for each device can be obtained over a period of time, then the mean, variance and Q factor can be calculated and the congestion rate determined. This is a very useful technique since the same process can be used on a small scale device such as a PVC or on a large scale ATM switch. It also means that congestion can be predicted for any device in a network, as long as the mean and variance of the traffic can be obtained.

In most cases these traffic counts are obtained by a central management station who requests the information from the device using the SNMP protocol. In many cases however, it is not possible to obtain the traffic counts on every device in the network. For example, the device might not have an SNMP agent and therefore the statistics are not obtainable by the management station. In other situations the management traffic required to obtain the statistics may be too high or expensive. In many vendors network equipment, the required traffic count objects may not be supported by their MIBs. In these situations, the Q value for the device can still be calculated as

long as the traffic counts on all of the inputs into or outputs out of the device can be obtained. For example, in the Bell ATM Network, the traffic counts on the transmission lines connecting the switches were available but the actual traffic counts on the switch were not available. This chapter describes a mathematical formula that can be used to calculate the Q value for a device from the means and variances of either the input or output lines to that device.

This method can be used if statistics on individual PVCs are available but transmission line statistics are not. Then, the traffic counts for each PVC on a transmission line can be used to calculate the Q value for the line. Similarly if the traffic counts for all of the transmission lines entering or leaving a switch, bridge or router are available then the Q value for the switch, bridge or router can be calculated.

Unless there is congestion present in the device at the time of monitoring, the amount of traffic entering the device should be the same as the amount of traffic leaving the device. For instance, switches, routers and bridges all take cells or packets from one port and send them out to a different location, either to another network device or to an end user device. Therefore either the input or output traffic counts can be used to calculate the Q value of the device. The only other reason for the input traffic being different from the output traffic, besides packets being dropped due to congestion, is if there is problems with the actual collection of the statistics. This can occur if a device goes down for a period of time or if it does not respond to the management request.

Section 5.2 describes the Q combination formula that can be used to calculate the Q value for a device using the traffic counts on either the input or output connections to that device. This section shows mathematically that the Q combination formula is valid for different traffic distributions and is independent of network topology. Section 5.3 describes two heuristical Q combination formulas that provide reasonably good results for calculating the Q value. Section 5.4 provides the results of applying these formulas to the Bell ATM Network.

5.2 Theoretical Q Combination Rule

The goal of this Section is to provide a rule that can be used to determine the Q value of a device given the means and variances of all the inputs or outputs to that device. Since the Q value is derived from the mean and variance, some basic statistical principles can be applied to develop a formula.

The inputs to the device that the Q value is being calculated for can be regarded as random variables. Each input, whether it corresponds to a PVC, a transmission line or a network device is a random variable, with the mean and variance of the random variable corresponding to the mean traffic and variance in the traffic on that device.

If the random variables are independent, then the traffic on one input line is not influenced by the traffic on another input line. If this hypothesis holds, then the Q value for the device can be calculated by taking the sum of the independent random variables. The mean of the sum of independent random variables is equal to the sum of the means of the random variable. This can be found in most introductory statistics textbooks, such as [19]. Mathematically, if there are n random variables then adding these variables together would result in a variable with mean M,

$$M = \sum M_i \quad i = 1.. n \dots\dots\dots(5.1)$$

Similarly, the variance of the sum of independent random variables is equal to the sum of the variances of the random variable [19]. Mathematically,

$$V = \sum V_i \quad i = 1.. n \dots\dots\dots(5.2)$$

Recalling from Section 4.2 the definition for the Q value of a device d,

$$Q = M / (V)^{1/2} \dots\dots\dots(5.3)$$

Therefore, the summation of Q values yields the following Q combination rule, assuming independence of each input/output device.

$$Q = \sum M_i / (\sum V_i)^{1/2} \quad i = 1.. n \dots\dots\dots(5.4)$$

If the traffic on the input or output lines is not independent, meaning that the traffic on

some input is influenced by the traffic on another input line, then equation 5.4 is not completely accurate. It is assumed in this thesis that the devices in the network have independent inputs. For this to occur, the traffic on each line must be randomly distributed.

It is not necessary for the traffic on the input/output lines to belong to a specific distribution for the traffic to be randomly distributed. The traffic must only be random in nature. However, if the traffic on all of the lines represents the Poisson distribution or the Normal distribution, then equation 5.4 can be proven for that distribution. Refer to [19] for the proof that the mean of the sum of n independent Poisson or Normal random variables is equal to the sum of the n means. The same is true for the variance of the sum of n independent Poisson or Normal random variables. The resulting summation of the Poisson random variables is also from the Poisson distribution. The resulting summation of the Normal random variables is also from the Normal distribution.

The same deductions can be used on other distributions. For example, the sum of n Geometric random variables yields a variable with a Negative Binomial distribution. The sum of n independent Exponential random variables yields a variable with a Gamma distribution. The sum of n independent Gamma random variables yields a variable with a Gamma distribution.

The Q combination rule can be verified on a network by taking the Q value calculated using Equation 5.4 for the input lines and comparing it against the Q value calculated using the output lines. If the means and the variances of the input and outputs are the same, or similar, then the same Q value should be derived using either calculation. If the device is congested, or is discarding packets, then the Q values will be different. Section 5.4 demonstrates the use of the Q combination rule on the Bell ATM network. Section 5.3 provides alternative heuristical Q combination rules that were experimented with before the theoretical rule was realized.

5.3 Heuristical Q Combination Rules

Not all networks will have traffic that the theoretical Q combination rule derived in Equation 5.4 can be applied to. In this case, a different predictor could be derived through experimental measurements on the network. In this thesis, two alternate Q combination rules were experimented with on the Bell ATM network. The first one uses the sum of the mean traffic on the inputs or outputs divided by the sum of the standard deviations of the mean on the inputs or outputs. It is described mathematically in Equation 5.5. The second heuristic utilizes the Q value for each input or output along with the mean. It is described mathematically in Equation 5.6.

$$\text{Heuristic 1: } Q = \sum M_i / \sum (V_i^{1/2}) \dots \dots \dots (5.5)$$

$$\text{Heuristic 2: } Q = \sum (Q_i * M_i) / \sum (M_i) \dots \dots \dots (5.6)$$

These heuristical Q Combination rules are used on the Bell ATM Network in the next section along with the theoretical Q Combination rule described in Section 5.4

5.4 Results of Combining Q Values on the Bell ATM Network

The theoretical and heuristical Q combination rules described in the previous two sections were tested on the Bell ATM Network. Several switches were selected at random and the Q combination rules were applied to both the input and the output lines to the switch. This experiment was applied to two different sets of data, one in the summer and the other in the fall. The accuracy of the combination rules can be observed by taking the Q value calculated using the incoming lines and comparing with the Q value calculated for the same switch using the outgoing lines. The two values should yield the same result.

Table 5.1 lists the results obtained from the summer data while Table 5.2 lists the results obtained from the fall data. The first column in the table indicates the switch number. These were chosen at random and do not correspond to the same switch numbers described in other tables in this thesis. The second column lists the number of lines attached to each switch. This number corresponds to the number of input transmission lines for the In row and the number of output

transmission lines from the switch for the Out row. The third column in the table indicates whether the Q value is calculated for the incoming lines (In) or the outgoing lines (Out). The next column lists the Q value calculated using the theoretical Q combination rule (Equation 5.4). The column headed Heuristic 1 lists the Q value calculated for the switch using the first heuristic Q combination rule (Equation 5.5) and the Heuristic 2 column uses Equation 5.6 to calculate the Q value. The last column in the table shows the sum of the mean level of traffic on the switch.

In the cases where the sum of the mean traffic level is different for the incoming and outgoing lines on the switch, the Q In value is different from the Q Out value. This is to be expected, since it means that not all of the cells coming into the switch are sent out. This can be due to congestion, although that is not the case in this fairly lightly loaded network. The reason for the inconsistent mean values in this network is attributed to the device going down due to switch problems during the sampling period. It can be seen from the tables, that on the switches that have consistent mean values, the theoretical Q combination rule produces Q values that are very close to each other for the incoming and outgoing summations.

The accuracy of the theoretical Q combination rule can be better seen by plotting the Q value for the incoming lines versus the Q value for the outgoing lines. The result should yield a straight line intercept at the origin, with a slope of 1. Figure 5.1 plots the incoming Q value versus the outgoing Q value for the fall data corresponding to Table 5.1. Figure 5.2 plots the same graph for the summer data corresponding to Table 5.2. It can be seen from the graphs that the heuristical Q combination rule (Equation 5.4) provides a good estimation to the Q value for the switch. The points on the graph that are a distance from the line going through the origin correspond to the switches which have inconsistent mean values.

The results from the heuristical Q combination rules are included in Tables 5.1 and 5.2 for completeness. In the situations where the mean traffic levels are consistent, Heuristic 1 (Equation 5.5) yields a closer approximation to the theoretical value than Heuristic 2 does.

Switch Number	Attached Lines	In / Out	Theory (Q Value)	Heuristic 1 (Q Value)	Heuristic 2 (Q Value)	Mean (Cells)
1	2	Out	11.01	9.09	8.88	75267
1	2	In	3.22	2.99	2.84	62467
2	2	Out	8.68	7.56	6.69	121804
2	2	In	6.64	5.34	4.70	72235
3	2	Out	6.62	6.17	6.10	127890
3	2	In	4.56	4.62	4.08	67710
4	2	Out	6.54	6.96	6.13	272355
4	2	In	9.43	9.39	8.59	272093
5	2	Out	5.28	4.75	4.70	289430
5	2	In	5.27	4.79	4.67	289055
6	4	Out	40.0	28.3	28.1	28673887
6	4	In	36.6	25.9	25.7	28671952
7	4	Out	14.5	10.3	10.2	28409437
7	4	In	14.5	10.3	10.2	28429503
8	4	Out	4.24	6.44	2.90	44617
8	4	In	4.30	5.97	2.93	45325
9	5	Out	4.17	4.24	2.45	57100
9	5	In	4.14	4.05	2.35	57354
10	5	Out	2.88	2.03	1.81	85905
10	5	In	3.20	2.05	1.83	84188
11	4	Out	9.56	5.60	5.54	305647
11	4	In	6.13	4.57	4.44	580226
12	4	Out	12.0	11.1	7.81	435948
12	4	In	3.44	8.16	2.67	629966
13	6	Out	2.24	2.17	1.75	546111
13	6	In	0.95	1.92	0.90	967297
14	7	Out	1.92	2.29	1.47	710440
14	7	In	1.31	2.30	1.10	816014
15	6	Out	1.25	1.71	0.85	239266
15	6	In	1.41	5.30	0.94	270388
16	6	Out	0.58	0.69	0.56	2407974
16	6	In	0.58	1.11	0.56	2438402
17	6	Out	0.95	1.24	0.66	415288
17	6	In	1.02	2.85	0.70	446186
18	11	Out	0.84	0.70	0.42	4594332
18	11	In	0.90	0.59	0.45	4944299
19	11	Out	0.81	1.33	0.46	3018711
19	11	In	0.91	3.93	0.51	3394803
20	9	Out	0.96	1.07	0.50	891708
20	9	In	0.96	1.16	0.49	893918
21	9	Out	0.58	0.61	0.28	2611766
21	9	In	0.58	0.63	0.28	2614192
22	9	Out	0.80	0.49	0.39	4431461
22	9	In	0.80	0.50	0.39	4434534

Table 5.1 Q Combination Formula Values for Summer Data Set

Switch Number	Attached Lines	In / Out	Theory	Heuristic 1	Heuristic 2	Mean
1	2	Out	4.28	3.74	3.63	103248
1	2	In	4.11	3.50	3.39	88020
2	2	Out	3.14	9.61	2.97	24958
2	2	In	2.77	11.9	2.66	24058
3	2	Out	3.06	14.1	2.92	22097
3	2	In	2.87	15.0	2.74	20423
4	3	Out	3.79	2.79	2.35	41644
4	3	In	3.68	2.67	2.47	42259
5	2	Out	0.89	0.64	0.63	909372
5	2	In	0.89	0.64	0.63	909502
6	4	Out	2.29	1.63	1.49	184048
6	4	In	1.99	1.89	1.51	184089
7	4	Out	15.6	12.0	9.99	64526
7	4	In	18.6	14.9	9.66	64720
8	3	Out	12.2	9.23	9.23	401752
8	3	In	4.85	6.27	4.50	340818
9	8	Out	2.77	3.26	1.79	1113720
9	8	In	2.79	3.79	1.77	1114231
10	8	Out	1.37	1.60	0.90	1531368
10	8	In	1.40	1.38	0.87	1528063
11	6	Out	1.70	2.15	1.13	235934
11	6	In	1.92	5.24	1.30	266562
12	2	Out	3.88	8.68	3.65	12887
12	2	In	3.78	8.75	3.55	12443

Table 5.2 Q Combination Formula Values for Fall Data Set

Figure 5.1 Q incoming vs Q outgoing for switches in the Summer data set using the Theoretical Q Combination rule

Figure 5.2 Q incoming vs Q outgoing for switches in the Fall data set using the Theoretical Q Combination rule

5.5 Discussion

This chapter has shown that Q values can be calculated for devices in a network for which their Q value can not be directly measured. This is useful in situations where access to a devices' traffic characteristics is not possible. The device for which the Q value is calculated can be any network device, such as a PVC, a transmission line or a router. The function of the device is irrelevant, and all that is required is statistical information on the inputs or outputs to the device. The theoretical Q combination rule can be used to calculate a Q value given the means and variances for either all of the input or output lines to a device. The theoretical Q combination rule is derived using the principles of the sum of independent random variables. The Q combination rule is tested on the Bell ATM network, by comparing the calculated Q value from the incoming lines against the calculated Q value from the outgoing lines. The closer the two values are to each other, the more accurate the combination rule. The data from the Bell network yielded good results in switches which had consistent mean traffic levels for the incoming and outgoing traffic.

Chapter 6

Exploitation of Results

Chapters 4 and 5 have provided evidence that for the Bell Sygma ATM network, the Q value for a device appears to characterize that devices' traffic distribution. This observation can be used to estimate congestion in a high speed network where statistics on the traffic are available. This chapter describes a possible implementation of the results of this thesis, allowing network service providers to provision a reliable, cost effective network.

The proposed implementation is divided into two sections. The first section describes a technique that associates Q values with their distributions. This technique builds a library of traffic probability distributions that occur on a given network, and stores each distribution with its corresponding Q value. In order to do this, statistics must be available regarding the traffic on the network. The traffic counts are used to calculate the mean, variance and Q factor for each device. The implementation of this library requires an intelligent pattern matching engine, which is capable of comparing distributions together for similarities. It also requires an efficient storage and retrieval method. The details of this library implementation are described in Section 6.1.

The other component of the application can be implemented using a spreadsheet. The spreadsheet contains the current topology and configuration of the network. Changes to the network can be made and the spreadsheet would calculate the resulting Q values of the affected devices. The new Q values can then be used to lookup in the distribution library to find the corresponding probability distribution. Then, given the congestion point of the device, the distribution can be used to determine how often the device is expected to have traffic levels higher than the congestion point. The spreadsheet application is described in Section 6.2. Section 6.3 contains an example of how the library and spreadsheet applications function.

6.1 Developing a Library of Traffic Distributions

In order to determine the amount of time a device is expected to be congested the corresponding probability distribution for that device is used. Using the graph of the probability distribution, the area above the congestion point of the device corresponds to the expected congestion time of the device. These probability distributions have to be stored in such a way that lookups and comparisons can be made quickly, along with additions of new distributions when they are discovered. This section describes a method that was proposed in [4], which uses a library to store the distributions. It is likely that different companies networks will require a different library corresponding to the individual characteristics of that network, but ideally one library would be created that could be used by any network.

There are several issues that must be resolved in order to implement the distribution library. These include the methods to initially fill the library with distributions, to add a new distribution to the library, to search the library, and to compare two distributions together. It is anticipated that the library would become quite large, so efficient searching techniques are needed.

Two methods can be used to initially fill the library with distributions. One method is to build the library based on observed distributions on the network. The traffic counts on all lines in the network must be recorded and then the distributions determined using the bin method described in section 4.5. This was the method used in this thesis, however it is laborious and an automated method is more desirable. Another method is to combine the first method with simulation. The simulation could generate many different distributions to fill the library, using different self similar traffic models.

The library can be indexed based on the Q value. That is, when a change is made to the network and a new Q value calculated for a particular device, that Q value can be used as the search index in the library. Alternatively, if a statistical indicator is found that is a better than the Q value to classify distributions, then that indicator can be used as the library index.

The main work in the implementation of the library is in the area of comparing distributions. When adding a distribution to the library, it must be checked whether it exists already. Similarly, when searching the library, the distributions being looked for has to be

compared to the distributions in the library. In order to compare the distributions and decide on whether the distribution is close enough to an existing one or whether it is different, a pattern matching technique must be used. Using a statistical comparison test such as the Chi-Squared test, which determines whether two distributions are the same within a specified accuracy, is not specific enough for the library. The pattern matching has to be smart enough to know that even if distributions are almost the same, if the differences come in critical areas, then they should be considered different.

Section 6.3 gives an example of how the distribution library would be used, in combination with the spreadsheet application of Section 6.2, to predict congestion in a network device.

6.2 Using a Spreadsheet Program to Predict Network Performance

A main goal of this thesis was to propose a method that could be used by network service providers to predict the impact that changes in network topology and configuration could have on the network and its customers. The previous section described a method that generates and stores traffic probability distributions that occur on a given network. Traffic cell counts on the network are collected and from these statistics the library is built. As the network grows and changes, the traffic on the network must be continuously monitored to detect changes in traffic patterns on individual lines. As these patterns change, the library is updated with new probability distributions. This library can be used by the spreadsheet application described in this section.

The spreadsheet application would be used by network designers to plan and provision the network. The spreadsheet contains the means, variances and Q values for each line in the network. Depending on the type of network, the lines could be PVCs or entire virtual path segments or transmission lines. The spreadsheet also contains the value at which congestion occurs in each device. Using the current Q value of a device, its estimated congestion time can be read by looking up the probability distribution for that particular Q value in the distribution library. The designer can then make a change to the network. This change could be to add a new PVC to a path in the network, increase the traffic load on an existing line or reroute an existing PVC, among other

things. The designer makes these changes by altering the mean traffic and the variance for a new or existing device. The spreadsheet then recalculates the Q values for the devices that are affected by the change using the Q combination rule described in Section 5.2. The new Q values are used to lookup in the distribution library the new expected distributions of the devices affected by the change. The congestion rate and performance under the changed network are then compared with the original and a decision made on the appropriateness of the change.

In order to create the spreadsheet, statistics on the current performance of the network must be known. These include the mean traffic and the variance of the traffic, which are calculated based on the cell counts measured at regular intervals over a period of time. The congestion point of each device in the network must also be known in order to determine how often the device is expected to be congested.

The following is a list of definitions of values that are either used or calculated by the spreadsheet application.

d: Any device in the network for which congestion is being estimated. This could be a switch, router, bridge, or transmission line.

N_d : The number of lines entering device d.

M_i : The mean traffic on each line i, entering or leaving device d. ($i = 1..N_d$)

V_i : The variance in traffic on each line i, entering or leaving device d. ($i = 1..N_d$)

Q_i : The Q value on each line i entering or leaving device d. ($i = 1..N_d$)
 $Q_i = M_i / (V_i)^{1/2}$

M_d : The predicted mean traffic on device d.
 $M_d = \sum M_i \quad (i = 1..N_d)$

V_d : The predicted variance of traffic on device d.
 $V_d = \sum V_i \quad (i = 1..N_d)$

Q_d : The estimated Q value for device d.
 $Q_d = M_d / (V_d)^{1/2}$

$d_{\max}$: The maximum capacity of device d , before congestion occurs. It is measured in the same units as M_d and V_d .

H_d : The predicted standard deviation of the mean traffic, M_d , on device d .
 $H_d = M_d / Q_d$

$H_{d\max}$: The number of standard deviations that the maximum capacity of device d , $d_{\max}$, exceeds the mean predicted traffic, M_d .
 $H_{d\max} = (d_{\max} - M_d) / H_d$

$\text{Prob}(Q_d, H_{d\max})$: The probability that device d , with predicted Q value Q_d , has traffic that exceeds or equals $H_{d\max}$ standard deviations about the mean.

T_d : The fraction of time device d is estimated to be congested.
 $T_d = \text{Prob}(Q_d, H_{d\max})$

The M_i and V_i values of each line must be entered into the spreadsheet cells for each device in the network. The maximum capacity $d_{\max}$ also needs to be entered for each device. The spreadsheet then calculates all required values and ultimately generates the Q_d and $H_{d\max}$ values for all devices that are affected by the change. These values are then entered into the distribution library, and the estimated fraction of time that each device will be congested, T_d , is determined. Section 6.3 gives a numerical example of this process.

By using this method to estimate network congestion, designers can make changes to the network on the spreadsheet and be provided almost immediately with a performance estimate. As a result, designers can determine the best route to place a new customer on. They can anticipate when a higher speed transmission line is required and also see when they have over provisioned resources. This spreadsheet provides a first estimation of the performance of a high speed network. The process requires little execution time, compared to executing a simulation. The spreadsheet is straightforward to set up and only requires continual monitoring of traffic counts in the network to maintain the spreadsheet.

6.3 Spreadsheet and Distribution Library Example

This section provides a numerical example of how the spreadsheet application and library of distributions can be used by a network planner to predict congestion and performance of devices in a network.

Consider a switch, d , that has 4 input transmission lines, which carry traffic from attached LANs. These lines are denoted $L1$, $L2$, $L3$ and $L4$. Table 6.1 shows how the spreadsheet could be set up for this switch, and would be the same for each device in the network. The values in italics represent the calculations made by the spreadsheet. The means and variances for each of the inputs are entered into the spreadsheet, after being calculated from the traffic counts for that input line. Assume in this example that the units for the means and variances are in cells per second. The spreadsheet would then perform the calculation for the Q values for each input (Q_i). The spreadsheet would also calculate the M_d , V_d and Q_d , using the formulas described in Section 6.2.

i	M_i	V_i	Q_i	M_d	V_d	Q_d
1	600	$1.0 * 10^6$	<i>0.6</i>	<i>31600</i>	<i>$9.789 * 10^7$</i>	<i>3.19</i>
2	25000	$9.0 * 10^7$	<i>5.0</i>			
3	5000	$6.25 * 10^6$	<i>2.0</i>			
4	1000	$6.4 * 10^5$	<i>1.25</i>			

Table 6.1 Spreadsheet Application Example Before Changes

If the network planner anticipates that the traffic on transmission line $L2$ entering switch d is going to increase because of the addition of a number of workstations on the attached LAN, then the planner can make the changes to the spreadsheet and see the effect. Assume that it is expected that the mean traffic M_2 is going to double to 50000 cells per second and the variance is going to increase to $2.5 * 10^9$ cells per second. The network planner would change the mean and variance in the spreadsheet and the spreadsheet would recalculate Q_2 and the mean, variance and Q value for switch d . Table 6.2 shows how the spreadsheet would look after these changes, with the changes made in bold and the resulting calculations performed by the spreadsheet in italics.

i	M_i	V_i	Q_i	M_d	V_d	Q_d
1	600	$1.0 * 10^6$	0.6	56600	$2.50789 * 10^9$	1.13
2	50000	$2.5 * 10^9$	1.0			
3	5000	$6.25 * 10^6$	2.0			
4	1000	$6.4 * 10^5$	1.25			

Table 6.2 Spreadsheet Example After Increasing M_2 to 50000 and V_2 to $2.5 * 10^9$

As a result of the change in traffic on line L2, the Q value for switch d has decreased from 3.19 to 1.13. To determine the effect this has on congestion in the switch the value for Q_d is used as an index to the distribution library. Assume that the distribution library has been built, and that a distribution exists in the library that corresponds to this Q value. Table 6.3 shows the normalized distribution for this Q value.

Interval	Number of Occurrences	Interval	Number of Occurrences
[5,4]	2	(0.2,0.1]	2
(4,3]	4	(0.1,0]	4
(3,2]	12	(0,-0.1]	3
(2,1.5]	10	(-0.1,-0.2]	7
(1.5,1]	9	(-0.2,-0.3]	46
(1,0.9]	3	(-0.3,-0.4]	58
(0.9,0.8]	3	(-0.4,-0.5]	16
(0.8,0.7]	7	(-0.5,-0.6]	24
(0.7,0.6]	5	(-0.6,-0.7]	16
(0.6,0.5]	5	(-0.7,-0.8]	23
(0.5,0.4]	9	(-0.8,-0.9]	8
(0.4,0.3]	5	(-0.9,-1]	9
(0.3,0.2]	1	(-1,-1.1]	6

Table 6.3 Example Distribution for $Q = 1.13$

The network planner must know the congestion point for the switch, and in this case it is 131600 cells per second. Normalizing this value using the equation for $H_{dmax} = (d_{max} - M_d) / (V_d)^{1/2}$

$$(131600 - 56600) / (2.50789 * 10^9)^{1/2}$$

yields 1.5 standard deviations from the mean. This means that all traffic that is further than 1.5 standard deviation from the mean in either direction causes congestion. There are a total of 269 occurrences between -1.5 and +1.5 standard deviations from the mean in the expected distribution.

There are a total of 28 occurrences further than 1.5 deviations from the mean. Therefore the amount of time this switch is expected to be congested is

$$(28 / (28 + 269)) * 100 = 9.43\%$$

From this the network planner would conclude that by changing the traffic on line 2 to a mean of 50000 cells per second and a variance of $2.5 * 10^9$ would result in a switch that is expected to be congested 9.43 percent of the time.

As another example, assume that it is expected that the mean traffic M_2 is going to double again to 100000 cells per second and the variance is going to double to $5.0 * 10^9$ cells per second. The network planner would change the mean and variance again in the spreadsheet and the spreadsheet would recalculate Q_2 and the mean, variance and Q value for switch d. Table 6.4 shows how the spreadsheet would look after these changes, with the changes made in bold and the resulting calculations performed by the spreadsheet in italics.

i	M_i	V_i	Q_i	M_d	V_d	Q_d
1	600	$1.0 * 10^6$	0.6	<i>106600</i>	<i>$5.00789 * 10^9$</i>	<i>1.51</i>
2	100000	$5.0 * 10^9$	<i>1.41</i>			
3	5000	$6.25 * 10^6$	2.0			
4	1000	$6.4 * 10^5$	1.25			

Table 6.4 Spreadsheet Example After Increasing M_2 to 100000 and V_2 to $5.0 * 10^9$

As a result of the change in traffic on line L2, the Q value for switch d has decreased from 3.19 to 1.51. The value of 1.51 is looked up in the distribution library, and Table 6.5 shows the normalized distribution for this Q value.

Interval	Number of Occurrences	Interval	Number of Occurrences
[5,4]	5	(0.1,0]	3
(4,3]	4	(0,-0.1]	4
(3,2]	5	(-0.1,-0.2]	18
(2,1]	14	(-0.2,-0.3]	127
(1,0.9]	3	(-0.3,-0.4]	19
(0.9,0.8]	1	(-0.4,-0.5]	25
(0.8,0.7]	1	(-0.5,-0.6]	4
(0.7,0.6]	3	(-0.6,-0.7]	6
(0.6,0.5]	0	(-0.7,-0.8]	0
(0.5,0.4]	4	(-0.8,-0.9]	1
(0.4,0.3]	6	(-0.9,-1]	4
(0.3,0.2]	1	(-1,-1.1]	8
(0.2,0.1]	3	(-1.1,-1.2]	3

Table 6.5 Example Distribution for $Q = 1.51$

Now calculate the number of standard deviations that the maximum capacity of device d, exceeds the mean predicted traffic, M_d . Normalizing this value using the equation for $H_{dmax} = (d_{max} - M_d) / (V_d^{1/2})$

$$(131600 - 106600) / (5.00789 * 10^9)^{1/2}$$

yields 0.35 standard deviations from the mean. This means that all traffic that is further than 0.35 standard deviation from the mean in either direction causes congestion. There are a total of 68 occurrences between -0.35 and +0.35 standard deviations from the mean in the expected distribution. There are a total of 158 occurrences further than 0.35 deviations from the mean. Therefore the amount of time this switch is expected to be congested is

$$(158 / (158 + 68)) * 100 = 69.9\%$$

From this the network planner would conclude that by changing the traffic on line 2 to a mean of 100000 cells per second and a variance of $5.0 * 10^9$ would result in a switch that is expected to be congested 69.9 percent of the time.

6.4 Monitoring the Accuracy of Prediction Results

The accuracy of the congestion prediction technique described in this thesis is not known, although is estimated in [4] to be at most twenty percent. However, on an implemented system,

the accuracy can be continually monitored. The prediction method requires the traffic counts on each device to be continually monitored, so that the mean, variance and Q value of each device can be calculated. When a network operator is anticipating a change in the network, they can use the spreadsheet and distribution library application to predict the effect of the change. Then when the change actually occurs on the network, since the traffic counts on the devices are still being recorded, the actual result can be observed. The actual result can then be compared with the predicted result and the accuracy determined. This ability to monitor the accuracy of the results is an attractive feature of this congestion prediction method.

6.5 Fast Simulation Using the Q Value

One of the goals of this thesis was to provide a method to predict congestion in a network using simulation. Chapter 3 described problems that occurred while trying to simulate an ATM network due to the large amount of traffic that needs to be generated. From the observations made in Chapter 4 regarding the Q values, a different approach can be taken to simulating an ATM network.

Using the observation that the traffic distributions on a network are relatively stable with time and that Q values for self similar traffic are similar with different interval measurements, then the simulation could be run at a lower rate than normal. If the Q factor on a network are the same for one second as they are for a day then if the simulation was run for 1000 seconds and measurements made every second, then the measurements would provide probabilities of 1 part in 100, regardless of the time units [4]. This simulator could be run at much lower levels of traffic (using a self similar traffic generator) than the real network, since the Q values are affected by the distribution and not the level of traffic. The result is a simulation that used to take several orders of magnitude longer than real time now takes close to real time to complete.

Chapter 7

Conclusions, Contributions and Future Research

The following conclusions identify the observations made and results obtained in this thesis.

1. The problems associated with simulating large high speed networks were described in Section 3.4. An attempt was made to simulate a telco size ATM network, generating cells at a rate of 155 Mbps. The time required to simulate only a few seconds of real time traffic was found to take several months on a Sparc 10 workstation. This is as a result of each cell raising several events as it passed through the network, resulting in a large event processing overhead. Consequently, it was concluded that a commercial simulation package was not capable of modeling a large ATM network in sufficient detail as to provide timely results on details of cell level performance.
2. A technique to scale down a high speed network simulation was described in Section 6.5. This scaling method permits a commercial simulation package to provide results several orders of magnitude faster than without scaling.
3. Empirical evidence from the Bell ATM network was provided that shows traffic distributions on devices are relatively stable over time, notwithstanding a known increase or decrease in traffic. This was shown by comparing the distributions of lines in the summer with their distribution in the fall. Section 4.4 shows examples of lines that exhibit stable distributions over the time period.
4. A new statistical indicator, the Q value, was introduced in Chapter 4. The Q value represents the ratio of the mean traffic on a device to the standard deviation of the traffic. In Section 4.5, empirical evidence from the Bell ATM network supports the premise that devices with similar Q values have similar traffic distributions. Therefore, each Q value can be associated with a distribution. This distribution can then be used to predict performance using a library lookup technique, such as the one described in Section 6.1.
5. The Q values for the Bellcore aggregated Ethernet data were calculated over five different time scales. Section 4.6.1 shows that the Q values vary slowly over time for this self similar traffic. Therefore, Q values can be measured on one time interval and mapped to a

different time interval.

6. The Q value for a device varies with the number of samples taken and the length of time between samples. Sections 4.6 and 4.7 show that for the Bell ATM network, the Q value varies over a twenty four hour period, and that at least one hundred samples at one hour intervals are necessary in order to get a stable estimate of the Q value.
7. It might not always be possible to calculate the Q value for a switch or other network device due to an inability to measure traffic counts on the switch. However, if the Q values for all incoming or outgoing lines to the switch can be calculated, then the Q value for the switch can be calculated from the Q values of the entering lines. This is shown mathematically in Chapter 5 for different classes of distributions.
8. A practical application which uses the Q value to estimate congestion in a network is given in Section 6.3. A spreadsheet example is shown that takes the mean and variance of each input line into a switch and calculates the Q value for the switch. Then, the traffic on any of the input lines can be varied, by changing the mean and/or variance of that line. The spreadsheet program recalculates the Q value for the switch. This Q value is then looked up in a distribution library and the amount of time the device is expected to be congested can be determined, provided that the congestion point of the switch is known.

7.1 Contributions

The work in this thesis has resulted in the following contributions to the area of traffic analysis and performance prediction of ATM networks.

1. Provided empirical evidence supporting the claim that the traffic distribution of a network device can be identified by a statistical indicator referred to as the Q value. The Q value is based on the mean and the variance of the traffic on the device.
2. Demonstrated that in an operational ATM network, traffic distributions of lines are often stable over time.
3. Showed mathematically that Q values of incoming or outgoing lines to a device can be used to calculate the Q values for the device.
4. Showed empirically, using Bellcore aggregated Ethernet data, that Q values vary slowly

over time for self similar traffic.

5. Detailed a method to scale down the amount of traffic required to simulate a high speed network based on self similar traffic. The result is a fast simulation, which executes several orders of magnitude faster than without the scaling.

7.2 Future Research

The observations and results detailed in this thesis opens up a wide area of possibilities in performance prediction of high speed networks. The following list identifies tasks that would provide further confidence in the evidence presented in this thesis.

1. Repeat the process described in Chapter 4 on the ATM network when there is more traffic on the network.
2. Repeat the process described in Chapter 4 on a frame relay network, to demonstrate the belief that this method can be used on many different types of packet based networks.
3. On the ATM network, repeat the calculations of Q values over different sampling intervals, this time using smaller sampling intervals.
4. Investigate other statistical indicators, besides the Q value, as possible probability distribution identifiers.
5. Implement the spreadsheet application described in Section 6.2 for the ATM network. This will allow monitoring of how accurate the predictions made by the Q value process really are.
6. Determine how many bins are required to group distributions for a large number of devices into to provide accurate results.

References

- [1] R. Addie, M. Zukerman and T. Neame, "Fractal traffic: Measurements, modelling and performance evaluation," in *Proceedings of IEEE Infocom 1995*, pp. 977-983.
- [2] J. Beran, R. Sherman, M.S. Taqqu, and W. Willinger, "Long-range dependence in variable-bit-rate video traffic," accepted for publication in *IEEE Transactions on Communications*.
- [3] D. R. Cox, "Long-range dependence: A review", in *Statistics: An Appraisal*, H. A. David and H. T. David, eds. Ames, IA: The Iowa State University Press, 1984, pp. 55-74.
- [4] N. W. Dawes, S. Stark, and B. Pagurek, "Direct estimation of traffic distribution and congestion rates in communication networks," Technical Report SCE 95-14, Department of Systems and Computer Engineering, Carleton University, Nov. 1995.
- [5] N. Duffield, J. Lewis and N. O'Connell, "Predicting quality of service for traffic with long-range fluctuation," *Proceedings of ICC '95*, 1995, pp. 473-477.
- [6] A. Erramilli and J. Wang, "Monitoring Packet Traffic Levels," *Proceedings of IEEE GLOBECOM*, San Francisco, CA, 1994, pp. 274-180.
- [7] A. Erramilli and W. Willinger, "Fractal properties of packet traffic measurements," in *Proceedings of St. Petersburg Regional ITC Seminar*, St. Petersburg, Russia, 1993, pp. 144-158.
- [8] J. Evans, *Structures of Discrete Event Simulation*, England: Ellis Horwood Limited, 1988.
- [9] H. J. Fowler and W. E. Leland, "Local area network traffic characteristics, with implications for broadband network congestion management," *IEEE Journal of Selected Areas in Communication*, vol. 9, pp. 1139-1149, 1991.
- [10] V. Frost and B. Melamud, "Traffic modeling for telecommunications networks," *IEEE Communications Magazine*, vol. 33, pp. 70-80, Mar. 1994.

- [11] M. Garrett and W. Willinger, "Analysis, modeling and generation of self-similar VBR video traffic," in *Proceedings of SIGCOMM '94*, Sept. 1994, pp. 269-280.
- [12] W. Hays and R. Winkler, *Statistics: Probability, Inference, and Decision*, New York: Holt, Rinehart and Winston, Inc., 1971.
- [13] P. Heidelberger and S. Lavenberg, "Computer performance evaluation methodology," *IEEE Transactions of Computers*, Vol. C33, Dec. 1984, pp. 1195-1217.
- [14] R. Hogg and E. Tanis, *Probability and Statistical Inference*, New York: Macmillan Publishing Company, 1988.
- [15] J. Kleijnen and W. Van Groenendaal, *Simulation A Statistical Perspective*, England: John Wiley & Sons, 1992.
- [16] W. Lau, A. Erramilli, J. Wang, W. Willinger, "Self-Similar Traffic Parameter Estimation: A Semi-Parametric Periodogram Based Algorithm," *Proceedings of IEEE International Conference on Communications*, 1995, pp. 2225-2231.
- [17] A. Law and M. McComas, "Simulation software for communication networks: The state of the art," *IEEE Communications Magazine*, Mar. 1994, pp. 44-50.
- [18] G. Lazarou, V. Frost, J. Evans and D. Niehaus, "Using Measurements to Validate Simulation Models of TCP/IP over High Speed ATM Wide Area Networks," *Proceedings of IEEE International Conference on Communications*, 1996, pp. 309-314.
- [19] W. E. Leland, M. Taqqu, W. Willinger, and D. Wilson, "On the self-similar nature of Ethernet traffic (extended version)," *IEEE/ACM Transactions on Networking*, vol. 2, pp. 1-15, Feb. 1994.
- [20] B. Morgan, *Elements of Simulation*, New York: Chapman and Hall Ltd, 1984.
- [21] V. Paxson and S. Floyd, "Wide area traffic: The failure of Poisson Modelling," *IEEE/ACM Transaction on Networking*, vol. 3, pp. 226-244, June 1995.
- [22] U. Pooch and J. Wall, *Discrete Event Simulation*, Florida: CRC Press, Inc. 1993.

- [23] S. Wilks, *Mathematical Statistics*, New York: John Wiley & Sons, 1962.