Modeling of Network Traffic Variability with Applications to Performance Evaluation

18th July 2002

Modeling of Network Traffic Variability with Applications to Performance Evaluation

by

Sarat Chandrika Pothuri

B.Tech. (Electrical & Electronics Engineering),

Jawaharlal Nehru Technological University,

College of Engineering, Hyderabad, India, 2000.

Submitted to the Department of Electrical Engineering and Computer Science and the Faculty of the Graduate School of the University of Kansas in partial fulfillment of the requirements for the degree of Master of Science

_____

Professor in Charge

_____

_____

Committee Members

_____

Date Thesis Accepted

Acknowledgements

First and foremost, thanks to my parents, and Gopi, my husband. Thank you for the love, support, encouragement and inspiration you have continuously provided me with and for giving me room to follow my dreams. I am deeply grateful to my grandparents, brother and cousins for the love and valuable advice.

I would like to express gratitude to Dr. David W. Petr for being my advisor. He has offered constructive counsel on many issues related to my academic and research activities and his guidance in developing and writing the thesis was extremely helpful. Working with him has been a great learning experience.

I also want to thank Dr. Victor S. Frost and Dr. Swapan Chakrabarti for being in the committee. In addition, I thank Dr. Petr and Sprint Corporation for sponsoring my graduate program at KU.

At ITTC, there are also many thanks to go around. During the years in KU, I met some of my best friends. I thank them for the interesting, valuable and enlightening discussions and debates. Not only have you made my hard work more pleasant to endure, but also you have given numerous unforgettable moments to cherish. Thank you for you support and friendship.

Abstract

Network traffic analysis in modern, heterogeneous, high-speed networks poses new challenges to traffic engineers: recent measurements of cell streams in these networks reveal a number of characteristics that traditional network traffic models cannot emulate. Network traffic is intrinsically bursty, meaning that the rate of transmitted cells or packets is subject to severe fluctuations. This burstiness shows up even when averaging over large intervals of time, a phenomenon referred to as self-similarity. Self-similar burstiness is a ubiquitous phenomenon present in various packet network concepts. There has been focus on mathematical models for its description, and performance analysis based on queuing in the network.

This thesis mainly focuses on characterizing traffic at various time scales which includes measures of self-similarity (Index of Variability, IDV) and burstiness (Peak Rate Variability, PRV). We also discuss traffic models based on moment matching and performance analysis done on the traffic data. The variability of the network traffic over a wide range of time scales is shown through analysis of PRV and IDV along with performance evaluation using G/M/1 analysis. We investigate the variability in traffic analytically as well as by simulation and conclude that a lower order hyperexponential interarrival model could be used to model network traffic. Parameter optimization for the model should not involve curve-fitting alone, but should include an attempt to capture statistics that affect queuing behavior. Therefore, we propose a simple optimization technique to match the IDV curve, traffic peak rate behavior and queuing properties.

Contents

List of Figures

Chapter 1

Introduction

Network traffic measurements [2] have shown that network traffic is bursty on a wide range of time scales, which cannot be captured by traditional traffic models. This scale invariant burstiness has led to the description of new concepts like peak rate variability and the index of variability, both of which are discussed in this thesis. Our goal is to develop new methods for network traffic modeling and analysis mainly for the purpose of evaluating the performance of the network, where analysis encompasses a wide variety for problems. Self similarity [1] is the property where the aggregate network traffic variability remains the same over an extremely wide range of time scales or over all time scales. This is in contrast to classical models which smooth off at large time-scales (e.g., Poisson arrival processes, Markovian models of packet traffic, etc.). If a time series is bursty at all time-scales, it exhibits long-range dependence (LRD) [3, 2]. The concepts of self-similarity and LRD complicate various simple assumptions and makes solutions sometimes analytically intractable. But, the simplicity of self-similarity lends itself well to practical applications in network dimensioning and traffic analysis [19]. The goal of this thesis is to develop a network traffic model

that can capture essential characteristics of traffic such as self-similarity, LRD and queuing behavior.

1.1 Network Analysis

In constructing a model, various simplifying assumptions are made for analytical tractability, but some of these assumptions may be fundamentally inaccurate. It is the role of the modeler to ensure that these unrealistic assumptions do not affect the outcome of analysis. In this thesis we concentrate on one class of such assumptions. We assume that the packet arrival process is Markovian. In particular, packet inter-arrival times are assumed to be hyperexponentially distributed [4]. We will show that this assumption is realistic. Self-similarity has been observed in packet networks, yet Markovian assumptions approximately hold true for them [24] and this strengthens the assumption. The inter-arrival times cannot be simply modeled as exponential as it over-estimates the results in terms of performance of networks [6]. Based on the assumptions suggested by the models, one can develop mathematical tools for estimation of network performance related to Quality of Service (QoS) parameters. Accurate models of traffic streams help in understanding the maximization of the network utilization.

This thesis provides a new framework to characterize the variability in the traffic through peak rate variability analysis and index of variability analysis.

There are two main difficulties in self-similar network traffic analysis:

1. Wide uncertainty in choosing a mathematical model.

2. Queuing theory tools for treating both LRD traffic and finite buffer queues.

Depending on the type of traffic and desired mathematical tractability, a network

model can be chosen. For example, file sizes in the Web were shown to have heavy-tail distributions [35]. Validity of the mathematical model can be checked by comparing various statistical properties of synthetic traffic (generated directly from the model) and the real network traffic. Queuing tools imply queuing models for analysis of network traffic. Queuing analysis again depends on the traffic type. We cannot describe the traffic arrival distribution prior to its flow into the queue. This poses difficulty in fixing the buffer size. Also, LRD traffic is bursty and there would be more packet loss with a fixed buffer size compared to smooth arrival traffic (eg. Poisson) flow into the same buffer with same mean arrival rate. So, it is difficult to estimate the size of buffer given an arbitrary LRD trace. In this thesis, we assumed general arrivals and infinite buffer size because analytical expressions exist for infinite buffer size. G/M/1 is used for arbitrary arrivals and exponential service rates. But, it is analytically difficult to derive expressions for the queuing analysis if the arrival and service time distributions are general (G/G/1). Verification of analysis using tools like Extend needs simulation of long LRD traces, posing memory problems in storing the data. The work done on choosing mathematical models for self-similar traffic and related queuing models is explained in Chapters 4 and 5.

1.2 Theoretical Background

This research work is organized around the landscape of recent developments and previous accomplishments. The initial effort for this thesis focussed on becoming familiar with the traffic in current ATM networks. This led to a new measure of burstiness, peak rate variability (PRV). Another main idea behind the thesis was the concept of Index of Variability or IDV [4]. Information about

the self-similarity and LRD is the main foundation for understanding the Hurst parameter, the measure of self-similarity. Considering the ATM traffic measurements determined by simple models of uncorrelated arrivals of cells, some basic estimation problems related to renewal processes are studied, using the results of [8, 9]. Our analytical work has been directed towards modeling packet traffic motivated by the idea of IDV and hence required the study of the power-tailed distributions and their queuing performances [10, 11]. In brief, a literature search was conducted which included traffic analysis, self-similar network modeling, performance modeling, simulation and optimization techniques.

1.3 Traffic Measurement

Asynchronous Transfer Mode (ATM) [38] is a high-speed connection-oriented network technology that sends data through switched and permanent virtual circuits in fixed length packets called cells. Both optical carriers (OC), OC-3 rate (155 Mbps) and OC-12 rate (622.08 Mbps) links that are included in this study are bi-directional and the data is analyzed at each uni-directional port on the link. Sprint personnel collected cell-count data on a per-virtual channel circuit (VCC) basis for several switch ports. The data consists of ATM cell counts tracked every 5 milliseconds for over a 24-hour period on a single switch connected to an OC link. Therefore, there are more than 17 million cell counts in each data set. Also, the basic definitions used in this thesis are as follows:

Fundamental Time Interval: A particular non-overlapping time slot in seconds. Each VC or link data set is divided into fundamental time intervals that are 5 milliseconds in length (10 ms for some links).

Aggregation Interval: An interval that is an integer multiple of 5 ms. For ex-

ample, a 1-second aggregation interval consists of 200 consecutive 5 ms intervals. Aggregation intervals are non-overlapping.

Cell Count: Number of cells in a particular aggregation interval.

1.4 Organization of Thesis

The main objectives of the thesis are to:

1. Verify the self-similarity of empirical network traffic traces.
2. Develop a model for the self-similar traffic.
3. Investigate the effects of self-similarity and LRD on the performance (delay and loss probability) of the network traffic.
4. Compare the simulated traffic generated from the proposed model to the real network traces.

Chapter 2 discusses the basic definitions of self-similarity and the derivations of IDV. This chapter defines self-similarity and LRD in terms of autocorrelation and variance of the counting process. Traffic similarity at various aggregation levels and burstiness has been observed leading to the study of self-similarity and LRD. The real network traffic traces were tested for self-similarity and LRD using the variance-time plots. A new method to derive the IDV from counts is also presented. This was investigated on different traffic traces to confirm the self-similarity and LRD in ATM network traffic.

Chapter 3 of this thesis introduces Peak Rate Variability (PRV) and examines the problem of estimating the peak rate at the lowest time scale given the largest time scale of measurement of data. Analysis is done on ATM networks

to understand the variability of peak rates at various time scales. ATM traffic data is analyzed to determine its peak rate behavior as a function of aggregation time ranging from 5 milliseconds to an hour. A linear relation is developed to find the peak rate at the lowest aggregation level given the peak rate at the highest aggregation. We also illustrate the self-similar tendencies of this traffic data by comparing it with synthetic data that is independent and exponentially distributed.

Chapter 4 deals with the modeling of the traffic using different models like MMPP and hyperexponential models. Preliminary analysis is done to match the IDV using a two state Markov modulated Poisson process (MMPP) source model. Higher orders of hyperexponential distributions are found to be more appropriate in network analysis. We attempt to represent the inter-arrival times of measured traffic stream using hyperexponential distribution of order 3 (H_3). As is commonly the case, the mean is matched along with specific constraints related to hyperexponential. The analysis was used in generating synthetic traces approximately matching the real traffic traces in terms of performance. We found that, with properly chosen parameters, the H_3 distribution was sufficient to uniquely characterize the IDV and queuing behavior. The rest of the chapter describes the heuristic and optimization techniques used for estimating the H_3 parameters. The efficiency of the optimization technique in finding the suitable H_3 parameters is also discussed.

Chapter 5 describes the queuing behavior of the network traffic and attempts to match the queuing results with G/M/1 analysis [12]. Various simulation techniques are used for generation of hyperexponential data and queuing of the network traffic. Synthetic data is generated using parameters of matched H_3 (hyperexponential distribution, n^{th} order represented as H_n). We investigate the

performance of a simple queuing system (G/M/1) subject to hyperexponential arrival traffic, where the real traffic, synthetic traffic and theoretical analysis are compared. The results strengthen the assumption of modeling the inter-arrival distribution as hyperexponential distribution. We also show that IDV, along with PRV, is an effective measure for synthetic data generation with similar statistical properties as the real network trace. We conclude in chapter 6 with the discussion of results, contributions and future work.

1.5 Framework of Results

There are two areas of focus of network analysis in this thesis.

1. Source modeling and performance evaluation based on IDV [4]. In particular, characterization of IDV for H_3 over various time scales as a function of H_3 parameters.

2. Peak rate variability of network traffic over various time scales. Relationship between peak rates at different aggregation intervals on OC-link was obtained.

Issue (1) is studied in this thesis, although more work in this area is clearly necessary. Issue (2) has been explored as a part of analysis on the Sprint data network.

1.6 Lessons Learned

This thesis contributes two characterization techniques for self-similar network traffic. The first characterization (PRV) is based on finding peak rate at various

time scales and the second being a measure of self-similarity as a function of time scale (IDV). From a practical point of view the important issues are the estimation of peak rate at different time scales, LRD phenomena, and parameters, especially the estimation of IDV.

For practical applications, PRV can be used for various cell/packet counts in the network to calculate the maximum peak rate possible in the link. There are various disadvantages in measuring the network traffic at a very small granularity (milliseconds). One of them is the inefficiency of storing large amounts of data in the disks. If detailed data is not stored, processing speed would be required to find peak rate for the data collected on each link. Therefore, we introduced this PRV curve, which can be used to estimate the maximum peak rate at a smaller time scale given the peak rate at larger time scale. Since the coarse measurements (one-hour aggregation) of data underestimate the short-term peaks, the peak rate should be calculated at various time scales.

The main use of this calculation is in the area of capacity planning in the networks. In this case, a peak rate pattern on various VCCs can be observed. For example, the PRVs can be plotted for a month with the daily traffic measurements and these patterns can help us explain the behavior ofchanging peak rates on the link. Further, the traffic causing the peak rate on a particular VCC can be identified and can explain the policing violations. Also, the pattern can be used in load balancing the traffic in the network. Loads on the links can be distributed to avoid any congestion in the network and help prevent cell loses.

Another major consequence of the maximum peak rate, congestion, can be identified. Based on the peak rate, buffer sizing, cell loss identification and traffic shaping can be done by checking the policing parameters on a particular link.

This study also helps in the cases of link failure. The ideas is to pre-determine

(based on peak rate patterns observed for a long period of time on the links) a route that has less loaded links and splice the traffic on the occurrence of a link failure. If a link failure occurs, the traffic could be diverted based on the load balancing factors and the available pre-determined link to the destination.

Another practical application is in customer education. Coarse measurements result in lower peak rates leading to misconceptions of peak rates advertised to the customer. PRV would be helpful to educate the customer about the peak rate variations and dependence of PRV on time scales. Large time scale measurements average out the short-terms peaks with the idle periods (no cells/packet for a period of time).

This understanding of traffic characteristics is very important in network performance prediction, and the identification of these phenomena is a focus of this thesis.

Another focus of our investigation is the estimation and interpretation of the Index of Variability (IDV) in case of real traffic. It is shown that if we use the Hurst parameter in practice we are faced with various misleading affects that can deceive our self- similarity tests and Hurst parameter estimation methods. Finally, we conclude that the estimated value of the Hurst parameter may be distorted in many practical cases and it may have no information for practical usage. Index of variability is a varying parameter that could be used to generate a synthetic traffic, matching the peak rate characteristics of a real network trace. This would help in predicting the future rate of the traffic based on the IDV curve. Moreover it is quite a challenge to predict the nature of the traffic in future services.There was an attempt to match the IDV of the traffic and then generate data that could match the PRV characteristics of the real traffic trace. Hyper-exponential model of order 3 was used to match the IDV curve and the parameters

of the hyper-exponential model generated the simulated traffic trace. Simulated traffic traces help in various performance studies like queuing performance. It saves the disk space to store the real traffic traces. Queuing analysis can be used to check the cell loss in the link and traffic behavior.

Together the PRV curve and IDV analysis can provide insight into the traffic characteristics as a function of time scale.

Chapter 2

Estimating Measures of Self-Similarity

In recent years, large amounts of high-quality measurement data in communication networks have been used to examine the validity of the traditional statistical assumptions made when analyzing such networks. These traditional assumptions contain the premise that network traffic can be described by Markovian models. This implies that autocorrelations in network traffic decay exponentially fast and traffic behaves smoothly over long time scales. Recent studies have found that these traditional (Markovian) assumptions are not always satisfied. But, network arrivals continued to be modeled as Poisson process for analytical simplicity. The Poisson assumption model was first refuted by Paxson et al. [5], who investigated the error introduced by modeling TCP arrival processes as Poisson arrivals. Later, Ethernet LAN traffic at Bellcore was analyzed [1] to prove that network traffic exhibits properties like self-similarity and long-range dependence (LRD) and can be modeled by heavy-tailed distributions like Pareto. An extensive bibliographical guide with 420 references to publications on self-similar traffic and analysis

[14] provides ample evidence that network traffic possess self-similarity and LRD. Such traffic behaves extremely bursty on a wide range of time scales. Subsequent sections give the basic definitions of self-similarity, LRD and Index of Variability [4], a relatively new measure of self-similarity.

2.1 Definitions

2.1.1 Self-Similar Process

A self-similar stochastic process is one whose statistical distributions are essentially invariant to scaling of the time axis. More precisely, scaling by a factor $m > 0$ has the same effect as multiplying the process by a factor of m^H.

$$Y(mt) \doteq m^H Y(t)$$

where Y(t) is a cumulative discrete-time process (arrivals up to time 't'). The notation $\doteq$ indicates that the two processes have the same probability law, and H is the Hurst parameter, referred to as the self-similarity parameter of the process. The best known example of such a process is the Poisson process, which has parameter $H = 0.5$.

2.1.2 Classifications in Self-Similarity

Let $X(t)$ be a stationary incremental process of Y(t), where $X_t = Y(t+1) - Y(t)$. Define the aggregated process of X_t as

$$X_t^{(m)} = \frac{1}{m}[X_{tm-m+1} + X_{tm-m+2} + \ldots + X_{tm}]$$

where m is the size of the aggregating block. The above equation implies that X_t is partitioned into non overlapping blocks of size m, and their values are averaged, and t is used to index these blocks.

The first two moments are assumed to exist and be finite. We define mean: $\mu = E(X_t)$, variance: $\sigma^2 = E[(X_t - \mu)^2]$ and auto-covariance function:

$$\gamma(k) = E[(X_t - \mu)(X_{t+k} - \mu)]$$

Let $\gamma^{(m)}(k)$ denote the auto-covariance function of $X^{(m)}$. A process is called second-order self-similar if $X^{(m)}$ and $m^{1-H}X(t)$ have the same second-order statistics for any $m > 0$. Second-order self-similarity can be classified into two categories:

Exactly Self-Similar

A process X is *exactly second-order self-similar* with self-similarity parameter H if $\frac{X^{(m)}}{m^{1-H}}$ has the same variance and autocorrelation as X.

$$\gamma^{(m)}(k) = \gamma(k)$$

Asymptotically Self-Similar

X is an asymptotically second-order self-similar process if the second-order characteristics of X and $\frac{X^{(m)}}{m^{1-H}}$ are the same for m tending to infinity.

$$\lim_{m\to\infty} \gamma^{(m)}(k) = \gamma(k)$$

Another feature of asymptotically second-order self-similar process is that the variance converges to zero slower than the rate m^{-1}.

$$Var(X^{(m)}) = \sigma^2 m^{-\beta} \tag{2.1}$$

where the Hurst parameter, $H = 1 - \beta/2$. For $m > 0$, equation 2.1 holds true for exactly second-order self-similar process too.

Self-similarity also implies that traffic is not memoryless. Thus for example, the probability that a current burst will continue for N packets depends on the

number of packets that burst delivered so far i.e, it depends on burst history.

2.1.3 Long-range Dependence (LRD)

Long range dependence (LRD), i.e., correlation over wide range of time scales, is an important factor in performance evaluation and traffic modeling of networks. Long-range dependence is a characteristic associated with an infinite time series. Long range dependence is not synonymous with self similarity but for $0.5 < H < 1$, a process is both LRD and self-similar. A process is LRD if the autocorrelation decays slowly i.e., hyperbolically.

$$\sum_{k=-\infty}^{k=\infty} \gamma(k) \rightarrow \infty$$

If $0 < H \leq 0.5$, then X is short range dependent i.e., autocorrelation is summable (exponentially decaying autocorrelation).

$$\sum_{k=-\infty}^{k=\infty} \gamma(k) < \infty$$

2.1.4 Estimating Techniques for the Hurst parameter

The measure of self-similarity, the Hurst parameter, can be estimated using techniques such as variance-time, wavelet method, Pox diagram of R/S analysis and periodogram-based analysis (in frequency domain). Detailed descriptions can be found from references [16] [1]. Second order properties can be used in finding mathematically tractable models. Also, the derivation of IDV involves variance of the traffic and so variance-time would be more appropriate to compare the Hurst parameter and IDV and their effectiveness for finding the network traffic variability.

2.1.5 Variance-Time Method

The variance-time method is one of the statistical tests for self-similarity. This method is based on the property that a self-similar process has slowly decaying variances i.e., variance of aggregated process $X^{(m)}$ decreases more slowly than the reciprocal of m (Equation 2.1). Method to calculate Hurst from variance-time[1] plot:

1. Divide the trace data $X_1, X_2.....X_N$ into N/m blocks of size m. $m > 0$.

2. Average the series over each block.

3. Calculate sample variance given by:

$$Var[X^{(m)}] = \sum_{n=1}^{N/m}(X_n^{(m)} - \overline{X})^2/(N/m)$$

where

$$\overline{X} = (\sum_{n=1}^{N} X_n)/N$$

The variance-time plot is obtained by plotting $log(Var[X^{(m)}])$ against $log(m)$ and by fitting a simple least squares line discarding some values of 'm' (aggregation interval).

Taking log of Equation 2.1 and differentiating w.r.t log(m), we get the Hurst parameter.

$$log(Var(X^{(m)}) = -\beta log(m) + constant$$

$$slope_v = dlog(Var(X^{(m)})/dlog(m) = -\beta$$

$$H = 1 + slope_v/2 = 1 - \beta/2.$$

Figure 2.1 shows the estimation of the Hurst parameter using the least squares line fit. The linear fit can either be done by fitting straight line to the intermediate

[1]plotted on log-log scale. Variance-Time in this thesis refers to curve plotted in log-log scale.

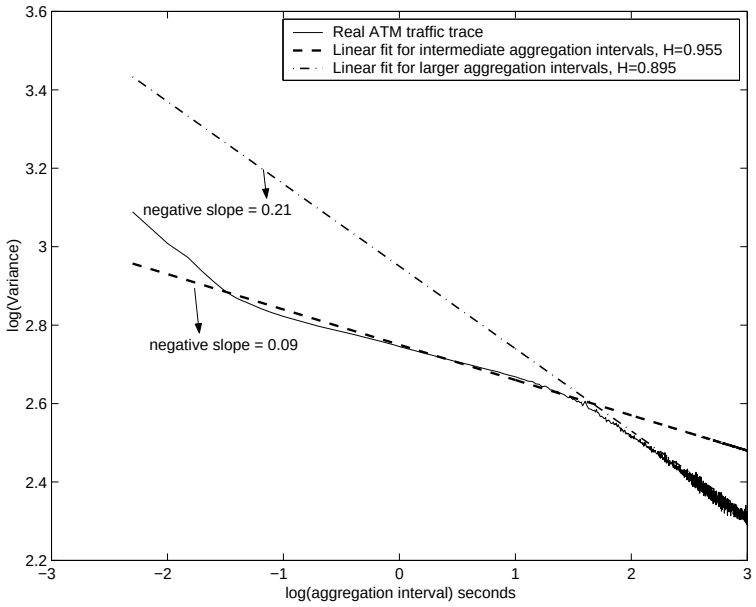


Figure 2.1: Variance-Time Plot, Hurst parameter Estimation

aggregation intervals (dashed line) or by fitting line to the higher aggregation levels (dotted-dashed line) (according to the variance-Hurst relation, equation 2.1). The slope (negative) of the linear line (dashed) is 0.09, therefore $H = 0.955$. Whereas $H = 0.895$ for the another linear fit (dotted-dash line).

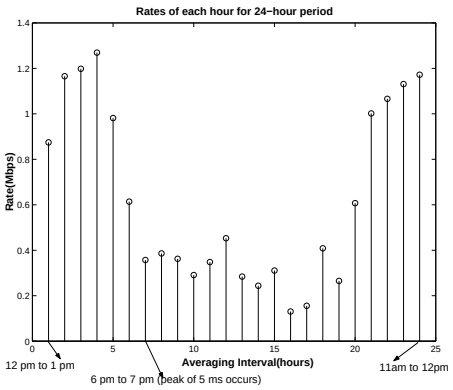
Hurst indicates the speed of decay of the autocorrelation function. H=0.5 implies that there is no variability (non-bursty i.e., smooth traffic as aggregation increases) as in the case of the Poisson process. As we know, $H < 0.5$ implies short range dependence (SRD) and $H > 0.5$ is long-range dependent (LRD), so it is confirmed that the traffic trace exhibits LRD from the values of the Hurst parameter (0.955 and 0.895). The Hurst parameter is claimed to be a good measure of variability [16] and is directly indicative of burstiness. But, the Hurst parameter does not consider the variability across all scales as the linear fit (dashed) is done discarding a few lower and higher values of 'm'. Another Hurst parameter (=0.895) for the same variance-time plot also does not capture

the variability. This implies that the Hurst parameter captures the burstiness correctly when there is linear decrease in variance for most or all of the time scales. The Hurst parameter has been used in performance evaluation (resource allocation) in recent studies [15, 17, 18, 19, 20] where the analysis just depends on the value of the Hurst parameter [21]. Since it is a single value derived from one of the methods of estimation of self-similarity, an estimation error may give incorrect results in performance analysis, leading to over/under estimation of utilization of network resources. There is every possibility that the linear fit may be imperfect or there may be numerous linear fits if the variance curve is extremely non-linear (Figure 2.1). Linear curve fitting in such cases results in error, and the Hurst parameter is influenced by the sample size and the technique to compute it [23]. Before discussing the other measure of self-similarity (IDV), we describe the analysis that led to the study of self-similarity and LRD.

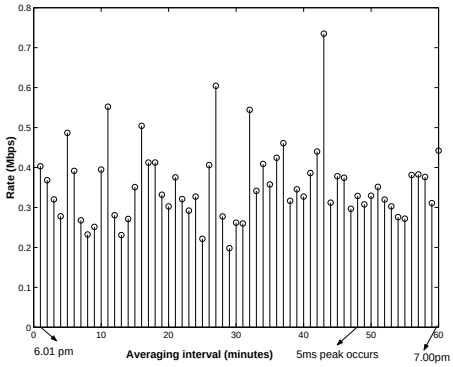
2.2 Validation of Self-Similarity

Traffic patterns in a network are predictable only in a statistical sense. The ATM data collected for this research project indicate that network traffic has similar statistical properties at a range of time scales: milliseconds, seconds, minutes, hours. This characteristic is referred to as self-similarity. Ethernet traffic was proved [16] to be self-similar and in this report, we show that ATM traffic also exhibits self-similarity using traffic rate properties.

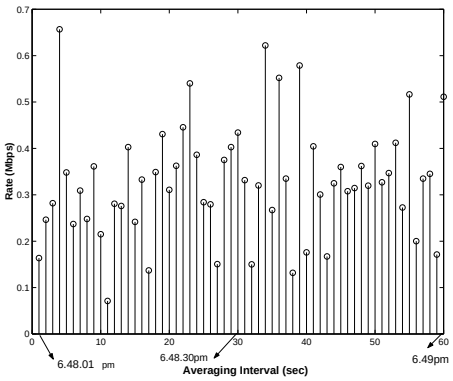
To illustrate the dynamic behavior and correlation of the structure of the process, a real network traffic trace is observed at different aggregation levels. Starting with the largest aggregation level (an hour), we successively examine smaller aggregation levels by zooming in on the part of the process where the



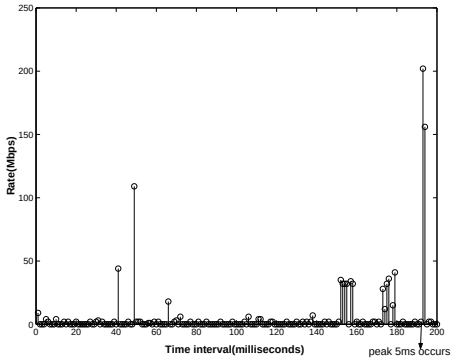
(a) Hourly Aggregation



(b) Expansion of the hour (6pm - 7pm)



(c) Expansion of minute (6.48pm - 6.49pm)

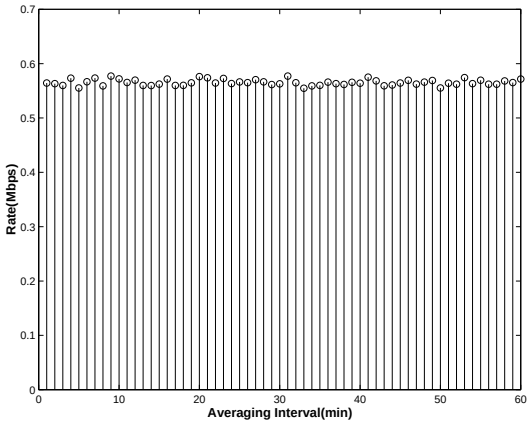


(d) Expansion of the second where maximum peak rate occurs

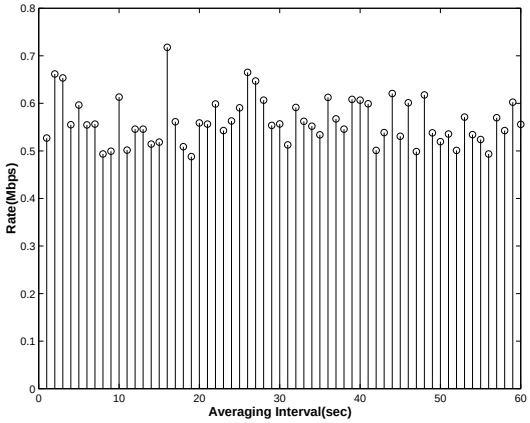
Figure 2.2: Rate Period Expansion for Traffic Trace.

5 ms peak rate occurs. Graphs with averaging time interval of one hour, one minute, one second and 5 ms (collected data) are plotted in Figure 2.2. Figure 2.2 shows a traffic trace, where average rate (Mbps) at the given aggregation level is plotted against time. In part (a), the time granularity is one hour. A single data point in this plot is the aggregated traffic volume over a 3600 sec period. The data point containing the maximum 5 millisecond peak rate is expanded to Figure 2.2 (b). As noted in Figure 2.2 (a), the 5 ms peak occurs between 6 p.m. and 7 p.m. but this is not the interval in which the peak of the hourly aggregated data occurs (4 p.m. - 5 p.m.). The aggregation level in Figure 2.2 (b) is one minute (60 s). Similarly, the other two Figures (fig. 2.2 (c) and fig. 2.2 (d)) are plotted with one-second and 5 ms granularity. By expanding the 6 p.m. to 7 p.m. interval window into a minute window (Figure 2.2 (b)) and then to seconds window (Figure 2.2 (c)), and finally to 5 ms window (Figure 2.2 (d)), the peak in each window occurs at a different point of time. This clearly indicates that the measuring interval is important factor for considering peak of a trace. Also, there is visual similarity among the plots, especially 2.2 (b) and 2.2 (c). Figure 2.2 clearly shows that the observed traffic trace is bursty on all time scales. This property is closely related to the notion of long-range dependence.

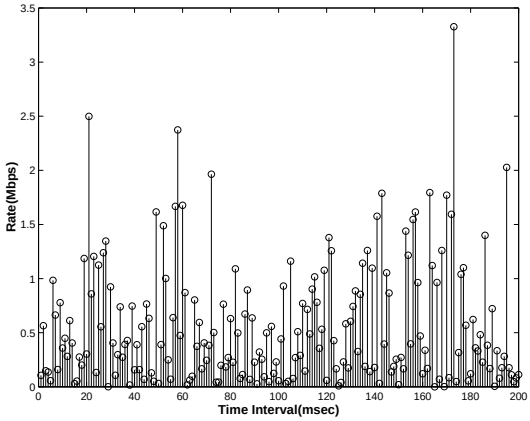
As observed in the previous paragraph, the ATM traffic was bursty on many or all time scales and looked similar at various aggregation levels. This is in stark contrast with traffic simulated from conventional traffic models. Figure 2.3 shows a trace obtained by generating independent counts, each of which is exponentially distributed, with the same average rate as the real ATM traffic trace used in plotting Figure 2.2. Starting with a time unit of one minute, each subsequent plot is obtained from the previous one by increasing the time resolution by a factor of 60 and by zooming in on a chosen subinterval. This traffic behaves smoothly



(a) Rate for Aggregation of a minute



(b) Rate for aggregation of a second



(c) Rates at 5 millisecond interval

Figure 2.3: Rate Expansion for Exponential Data

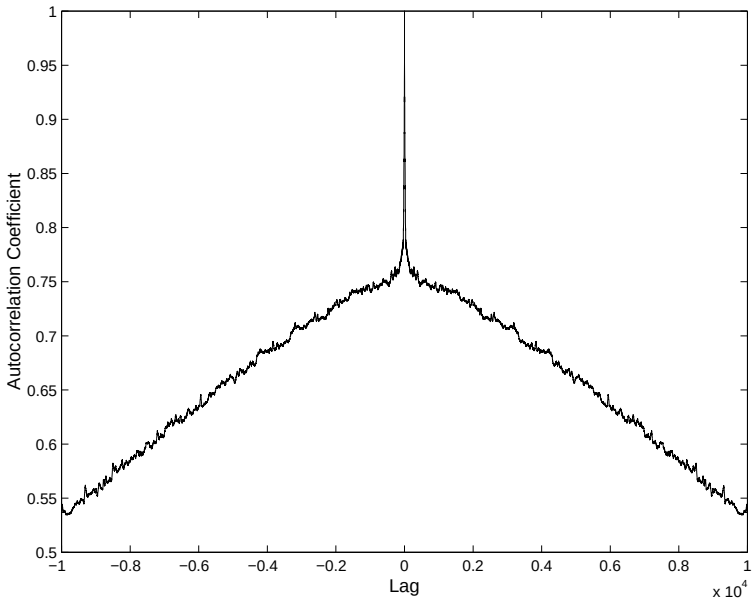


Figure 2.4: Autocorrelation of LRD data

on large time scales. Further statistical analysis of the correlation structure of measured network traffic shows that its autocorrelation function decays extremely slowly (Figure 2.4).

Burstiness

The critical feature in the plots of actual traffic is that the traffic is highly bursty over a wide range of time scales. Cell arrival rates over each hour for a period of 24 hours of ATM traffic can be noted from Figures 2.2 (a)-(d). The approximate similarity of widely varying peak rates is striking. There is high variation in burst lengths where highly busy (bursty) periods are separated by less busy periods. Another observation from Figure 2.2 is the resemblance of the plots with the magnitude suitably normalized, which indicates self-similarity.

A self-similar traffic stream will not "smooth out" over an extended aggregation of time as observed in the “rate expansion” plots (Figure 2.2). The effect of

self-similarity is to introduce long range correlation into the traffic stream which is a phenomenon that is observed in practice. This property is quite different from the data that is modeled by tradition Markovian model (Poisson process, Markovian arrival process (MAP), etc.).

2.3 Time Scale Dynamic Behavior : Index of Variability

The time scale dynamic behavior in traffic rates has already been examined in the previous section, where each aggregation window is expanded to view the variability in the process at each aggregation level. There are processes like Markovian modulated Poisson process (MMPP) that have $H = 0.5$ but still exhibit variability [4]. As mentioned in previous section, LRD is related to long time series. A time series can exhibit SRD or LRD tendencies depending on the aggregation levels that are considered. The Hurst parameter fails to give the complete variability at all time scales. A new measure of self-similarity has been proposed in [4]. This new measure is called Index of Variability (IDV or H_v) and is a better measure than the Hurst parameter since it captures variability at all time scales. Higher values of IDV imply high variability in traffic. IDV is calculated from Index of Dispersion of Counts (IDC) [24], which is the ratio of variance to mean of the process. Similar to variance-time plot (Figure 2.1), number of packets (counts) arriving in a given time slot have been examined at different aggregation levels (size of time slot) to find the IDC. IDV is a function of time scale describing the complete variability of traffic at all time scales as opposed to a single constant value of the Hurst parameter focusing on a few time scales. We must begin our discussion of IDV by reviewing the basics of point

processes.

2.3.1 Point Process

Any stochastic process in continuous time in which the sample paths are step functions, and therefore any process with a discrete state space, is associated with a point process, where a point is a time of transition, or a time of entry into pre-assigned state or set of states. Also, arrival processes can be described using point process [9].

A simple point process $\phi = \{t_n : n \geq 1\}$ *is sequence of points* $0 < t_1 < t_2 < \ldots$ *with* $t_n \to \infty$ *as* $n \to \infty$. *With* $N(0) = 0$, *let* $N(t)$ *denote the number of points that fall in the interval* $(0, t]$, *and* $\{N(t) : t \geq 0\}$ *is called the counting process for* ϕ. $N(t) = max\{n : t_n \leq t\}$.

ϕ becomes a random point process if the t_n are random variables. $Y_n = t_n - t_{n-1}$, $n \geq 1$ is called the n^{th} inter-arrival time and,

$$t_n = Y_1 + Y_2 + \ldots\ldots + Y_n \text{ with } t_0 = 0.$$

An important class of point process is the renewal process. A random point process $\phi = \{t_n\}$ for which the inter-arrival times $\{Y_n\}$ form an i.i.d. sequence is called a renewal process. In such a case, the subscript n may be dropped from the inter-arrival times Y_n. t_n is the n^{th} renewal epoch and $F(Y) = P(Y \leq y)$ denotes the inter-arrival distribution. The rate of the renewal process is defined as $\lambda = \frac{1}{E[Y]}$, where $E[Y]$ is the mean inter-arrival time. When the inter-arrival times are exponentially distributed, the renewal process is called Poisson process. Also, expected number of events that occurred during interval (0,t] is given by

$$E[N(t)] = \frac{t}{E[Y]}$$

A counting process is said to possess weakly stationary increments if the mean

and the variance of the number of the events that occur in any interval of time depends on the length of the time. We assume $N(t)$ to be weakly stationary. We refer τ_0 as fundamental time interval and τ as the time scale of the traffic trace, and τ ($=m\tau_0$, $m = 1, 2.....$) represents the measurement interval (i.e. 10ms, 1s, 1hr etc.). For each time interval $\tau > 0$, an event (packet) count sequence

$X = \{X_n(\tau), \tau > 0, n = 1, 2.....\}$ can be constructed from each point process, where the increment process

$$X_n(\tau) = N[n\tau] - N[(n-1)\tau]$$

denotes number of events that occurred during the n^{th} interval of duration τ. X is weakly stationary since the underlying point process $N(t)$ is weakly stationary. In this thesis, X represents a traffic trace and the underlying point processes have finite variances.

2.3.2 Derivation of IDV

The derivation of IDV according to [4] is presented first. Let $N(\tau)$ denote the number of events (packet counts, counting process) in the time interval (0,τ]. Using the notations of the previous section, the mean of the counting process can be described as

$$E[N(\tau)] = \lambda\tau \tag{2.2}$$

Variability of traffic was characterized [24] through the Index of Dispersion of Counts (IDC) defined as,

$$IDC(\tau) = \frac{Var[N(\tau)]}{E[N(\tau)]} = \frac{Var[N(\tau)]}{\lambda\tau} \tag{2.3}$$

Here τ is the time scale corresponding to one sample X_n. Network traffic

is constructed by one or more point processes. Since the arrivals are assumed independent, in a given traffic trace, we can consider the increments $X_n(\tau)$ to be sample functions of the counting process $N(\tau)$ for a given τ. The variance of $N(\tau)$ can thus be estimated as the variance of $X_n(\tau)$.

$$IDC(m\tau_0) = \frac{Var[N(m\tau_0)]}{E[N(m\tau_0)]} = \frac{Var[X_1(\tau_0)+X_2(\tau_0)+.....+X_m(\tau_0)]}{m\tau_0\lambda}$$

$$= \frac{mVar[mX^{(m)}]}{m\tau_0\lambda}$$

$$= \frac{mVar[X^m]}{\lambda\tau_0} \tag{2.4}$$

$X^{(m)}$ is the aggregated packet count process as defined in Section 2.1.2 and Section 2.3.1. Taking logarithm of equation 2.4 and replacing variance by its definition mentioned in equation 2.1, we get

$$IDC(m\tau_0) = \frac{m\sigma^2 m^{-\beta}}{\lambda\tau_0} = \frac{\sigma^2 m^{-\beta+1}}{\lambda\tau_0}$$

$$log(IDC(m\tau_0)) = log(\sigma^2) + (1-\beta)log(m) - log(\lambda\tau_0)$$

Taking derivative ($slope_I$) of $log(IDC(\tau))$ w.r.t $log(m)$, we get

$$slope_I = 1 - \beta$$

For a self-similar process, plotting $log(IDC(\tau))$ against $log(m)$ (IDC curve) results in an asymptotic straight line with $slope_I$ (slope of IDV curve) $2H - 1$, and thus

$$H = (slope_I + 1)/2$$

When X is a long-range dependent process, the slowly decaying variance property (equation 2.1) with parameter $0 < \beta < 1$ is equivalent to IDC curve with an asymptotic straight line with $slope_I$ $1 - \beta$, implying $0 < slope_I < 1$. When $slope_I$ is zero, then X is a short-range dependent process.

Relation Between Slope of IDC curve ($slope_I$) **and Slope of Variance-Time Plot** ($slope_v$)

We derive the relation between the slopes of IDV curve[2] and variance-time curve which is used to explain our method of deriving IDV from variance-time plot.

Refering to section 2.1.5, we derived that $slope_v = -\beta$

Therefore,

$slope_I = 1 + slope_v$ and $H = (1 - \beta + 1)/2 = 1 - (\beta/2) = 1 + (slope_v/2)$.

Different expressions of IDV

Index of variabilty is the Hurst parameter as a function of time. As defined in [4], index of variability ($H_v(\tau)$) of X for time scale τ is given by:

$$H_v(\tau) = \frac{1}{2}(\frac{d(log(IDC(\tau)))}{d(log(\tau))} + 1) \tag{2.5}$$

where $\frac{d(log(IDC(\tau)))}{d(log(\tau))}$ is the derivative ($slope_I$) of the IDC curve. Also, $\tau = m\tau_0$, $log(\tau) = log(m) + constant$. Therefore, $dlog(\tau) = dlog(m)$.

A simple expression is now derived relating IDV to the variance of the counting process.

Taking log of equation 2.4 and dividing through out by $log(\tau)$,

$$\frac{log(IDC(\tau))}{log(\tau)} = \frac{log(Var[N(\tau)]}{log(\tau)} - \frac{log(\lambda)}{log(\tau)} - 1$$

Taking derivative,

$$\frac{d(log(IDC(\tau))}{d(log(\tau))} = \frac{d(log(Var[N(\tau)])}{d(log(\tau))} - 1$$

[2]plotted on log-log scale. IDC curve in this thesis refers to curve plotted on log-log scale

Expanding just the variance term in the above equation and taking derivative of it, we get

$$\frac{d(log(Var[N(\tau)])}{d(log(\tau))} = \frac{log(e)dVar[N(\tau)]/Var[N(\tau)]}{log(e)\frac{d\tau}{\tau}} = \frac{\tau dVar[N(\tau)]}{Var[N(\tau)]d\tau}$$

Similarly,

$$\frac{d(log(IDC(\tau))}{d(log(\tau))} = \frac{log(e)dIDC(\tau)/IDC(\tau)}{log(e)\frac{d\tau}{\tau}} = \frac{\tau dIDC(\tau)}{IDC(\tau)d\tau}$$

Therefore,

$$\frac{d(log(IDC(\tau))}{d(log(\tau))} = \frac{\tau dIDC(\tau)}{IDC(\tau)d\tau} = \frac{\tau dVar[N(\tau)]}{Var[N(\tau)]d\tau} - 1$$

We can now use the above equations in expressing IDV in another form.

$$H_v(\tau) = 0.5\tau(\frac{dIDC(\tau)/d(\tau)}{IDC(\tau)} + 1) = 0.5\tau(\frac{dVar[N(\tau)]/d\tau}{Var[N(\tau)]}) \tag{2.6}$$

Given the variance of the process analytically, equation 2.6 is used in computing IDV theoretically (see Section 4.3.2).

IDV Calculation using Variance of Counts

Our approach in calculating IDV involved only variance of the counting process. Note that the factor of mean in IDC would not change the slope of the variance-time curve. So, our approach utilizes the second order properties of the packet traffic to derive IDV. We have already shown the relationship between $slope_I$ and $slope_v(-\beta(\tau))$.

From equation 2.5,

$$H_v(\tau) = 0.5(slope_I + 1) = 0.5(2 + slope_v)$$

Therefore, IDV can be expressed as

$$H_v(\tau) = 1 - \beta(\tau)/2 \tag{2.7}$$

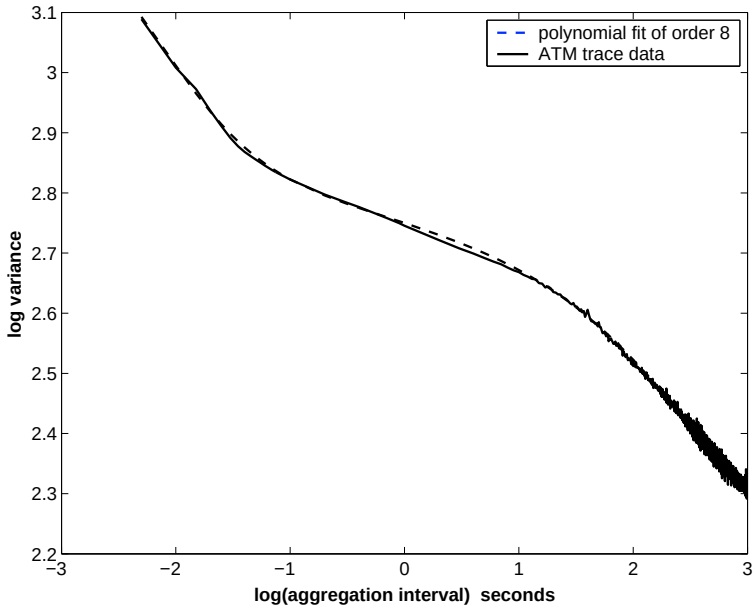


Figure 2.5: Variance-time Plot with Polynomial fit (order 8)

The negative slope $(1-\beta(\tau))$ of the log-log plot of IDC curve is a function of time scale. Since the slope is varying, $\beta(\tau)$ also varies with τ.

We calculate IDV from the real traffic traces by finding the variance of the data for every aggregation and fitting a polynomial to the non-linear variance-time plot (Figure 2.5). The derivative of the non-linear polynomial fit gives the slope $(dlogVar(N[\tau]/dlog(\tau))$ of the curve at each time scale τ $(-\beta(\tau))$. IDV is computed using equation 2.7. A polynomial of order 8 is chosen as the 'norm of residuals' value is low (Figure 2.6). The norm of residuals is a measure of the goodness of fit, where a smaller value indicates a better fit than a larger value. Order 8 is chosen since the norm of residuals is approximately same for orders greater than 8. After fitting the polynomial, Index of Variability, H_v is calculated by differentiating the polynomial w.r.t to log(aggregation interval) (equation 2.7). Figure 2.7 gives the IDV for the trace data whose variance is fitted by the polynomial (Figure 2.5).

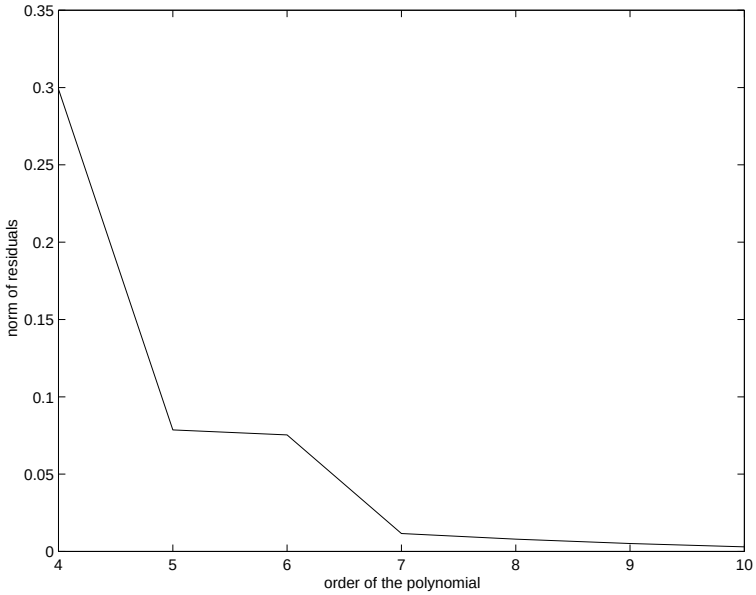


Figure 2.6: Norm of residuals vs. Order of polynomial

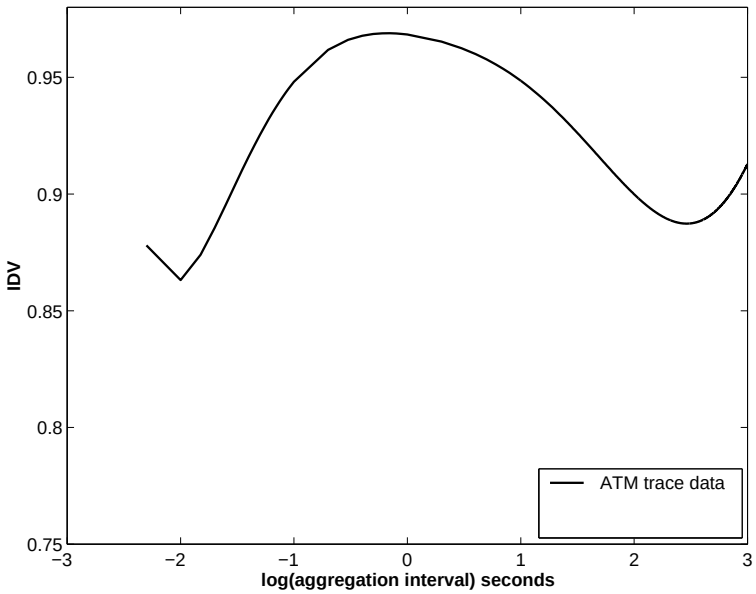


Figure 2.7: IDV Curve for ATM traffic trace

A purely second order self-similar process, $H_v(t)$ is a constant for all $'t'$. For an asymptotically second order self-similar process $H_v(t)$ approaches a constant value as $t \rightarrow \infty$.

Chapter 3

Peak Rate Variability

Network bandwidth is an important factor of data transfer over the virtual channels of the present day's information highway. Despite the high speed fiber optic links ranging to gigabits per second, there is congestion in the virtual channels creating higher rates (peak rates) in the channels. This may be due to network topology consisting of slow intermediate switches/routers, failures in the network and sudden routing changes in the network. Peak rate is an important factor to consider in the network to study the traffic variation on the links. Different aspects of peak rates are studied in this chapter.

3.1 Data Analysis

The data collected in the ATM (for ATM terminology, refer [38]) network was analyzed to study peak rate characteristics on the virtual circuits as well as on the links. There are two basic terms that recur in this section. They are defined as follows:

Peak Count: The maximum cell count among the cell counts in the similar

aggregation intervals. For example, the maximum cell-count among all 1-second intervals.

Peak Rate: The peak count for a particular interval converted into Mbps rate.

Each peak count is transformed into peak rate by the conversion factor 53∗8/ (length of aggregation interval), where 53 is the length of the cell (in bytes) and 8 is the number of bits per byte.

It was observed (Figure 2.2) that peak rate on the links varied as the time scale varied. So, this chapter discusses various characteristics and aspects of peak rate by doing the following studies:

1. Generating independent exponential data for a single VCC to compare with the actual data of that VCC to study its characteristics.

2. Finding equations describing the peak rate of the optical carrier (OC) links and individual VCC traffic.

3. Finding relationships between ratio of maximum to minimum peak rate and hourly (minimum) peak rate for VCCs on all the links and for the links themselves.

The variability of peak rate with the time was analyzed by studying the following plots:

1. Peak rate versus log of averaging time interval for the real traffic trace data on a single VC as compared to peak rate of exponentially distributed data.

2. Peak rate on an OC link consisting of multiple VCs.

3. Ratio of maximum peak rate to minimum peak rate in both OC-links and VCs.

3.2 Variability over Multiple Scales

Cell-counts aggregated over non-overlapping blocks of time are used for calculating peak rates at different aggregation time scales (fig 3.1). The 5 ms cell counts are aggregated over different time intervals and the peak rate (in Mbps) for each time interval is plotted versus the aggregating time interval (the latter being in log scale facilitating better visualization of the features in smaller aggregation intervals). We refer to such a curve as Peak Rate Variability (PRV) curve. It can be observed that the peak rate tends to fall as the aggregation interval increases and the minimum peak rate is obtained for the maximum aggregation of one hour, indicating the variability of peak rate over multiple time scales.

Figure 3.1 shows a few PRV plots chosen from set of PRV plots (about 300 PRV plots for VCs on OC-3 and OC-12 links). [13] is a detailed report of PRV plots on VCs and OC-links. From Figure 3.1(a), it can be seen that the peak rate is maximum for no aggregation (5 ms), remains high throughout the aggregation intervals up to about 25 seconds, then falls off when aggregated further. Note that the peak rate remains more or less constant for some aggregation intervals (0.1 sec-25 sec). The peak rate for 5 ms and 1-hour aggregation interval is approximately 11.5 Mbps and 3.95 Mbps respectively. Note also that the peak rate falls abruptly for aggregation intervals beyond approximately 75 seconds. The magnitude difference ($\tilde{}$ 8 Mbps) between maximum and minimum peak rates is much greater than the average arrival rate of the traffic ($\tilde{}$ 2 Mbps). It is also significant that the 1-hour peak rate (4 Mbps) is greater than the average rate (1.74 Mbps), which indicates that the traffic is still bursty at very large aggregation time interval. Peak to mean ratio evaluation is common in characterizing traffic but cannot provide much statistical information about the data. PRV plots were

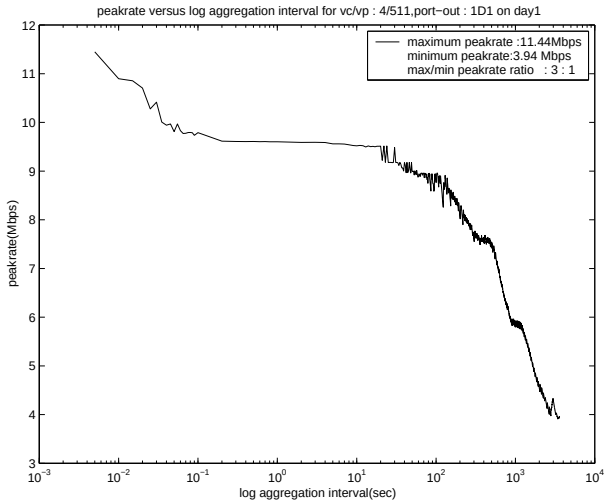


Figure 3.1 (a)

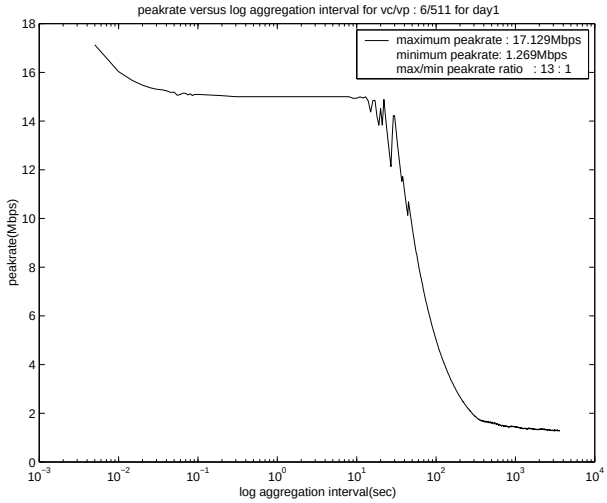


Figure 3.1(b)

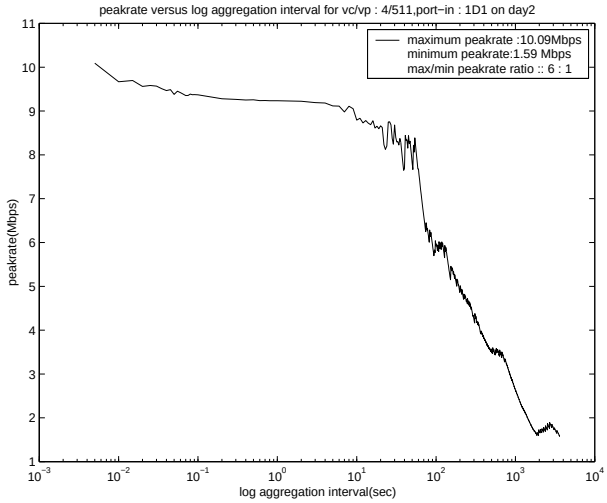


Figure 3.1 (c)

Figure 3.1: Peak Rate Variability of Network Traffic Traces.

generated for various VCs on an ATM link and a similar type of behavior was observed on all VCs. Some of the common similarities observed were as follows:

1. Abrupt fall at higher aggregation intervals (at approximately 10 seconds).
2. Constant peak rate for a range (100 milliseconds-5 seconds) of intervals.

It may seem to the reader that the peak rate should be monotonically decreasing as aggregation interval increases, but Figure 2.1 shows several jumps. The occurrence of these jumps in Figure 3.1 can be explained by this example:

Consider a set of points (cell-counts) which is a subset of the actual data with a measurement period of 5 ms between each cell count. A 50 ms duration implies 10 points in the set with a basic measurement period of 5 ms.

Set: {0, 0, 2, 20, 1, 16, 0, 6, 0, 0}.

It is important to note that cells are aggregated over non-overlapping blocks of time for calculating peak rate. When the 5 ms interval is considered, the peak rate would be 1.696 Mbps as the 4th element accounts for the highest cell count in that interval.

(20*53*8)/5ms = 1.696 Mbps.

Summing pairs of cell counts for an aggregation of 10 ms, the peak occurs in the second 10 ms slot (3rd & 4th element in the set), which would be:

(20+2)*53*8/10 ms = 0.933 Mbps.

This exhibits the expected decrease in the peak rate. However, by summing the cells for an aggregation of 15 ms, the peak rate can be obtained by adding the second 15 ms slot (4th, 5th & 6th element in the set) which would be:

(20+1+16)*53*8/15 ms = 1.14 Mbps.

Therefore, depending on the traffic, the peak rate might increase as aggregation increases, and this behavior is exhibited in Figure 2.1.

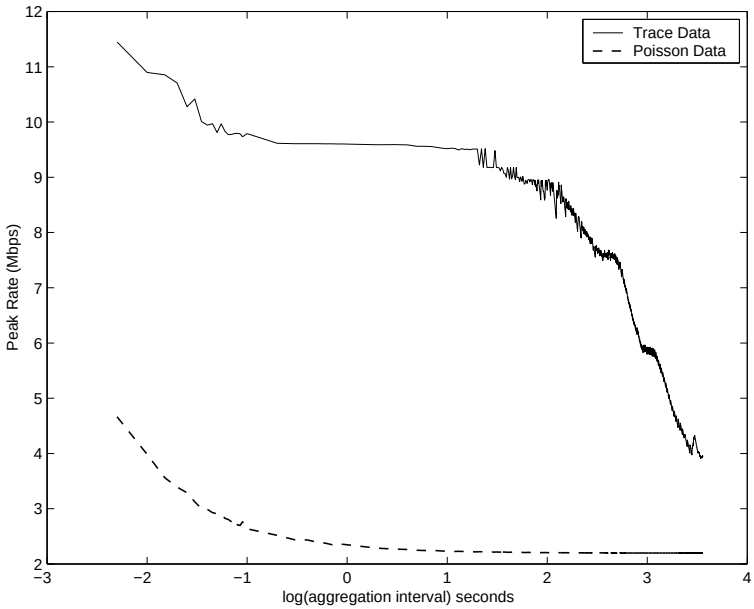


Figure 3.2: Peak of Poisson Data with exponentially distributed Inter-arrivals

Comparison with Poisson Arrivals

A sample function (synthetic trace) of a Poisson arrival process is generated with the same length and mean arrival rate as that of the ATM trace. This data was generated by transforming uniformly distributed data and the mean was equated to that of the collected network data.

The dashed line in Figure 3.2 shows that the peak rate in the generated Poisson data falls off with increasing aggregation time in an exponential-like fashion and remains essentially constant after the 10 second aggregation interval, whereas for the collected data (solid line in Figure 3.2) the peak rate remains constant for aggregation intervals ranging from 0.25 seconds to 25 seconds and falls off abruptly after that. The peak rate at all time scales of Poisson traffic is much smaller than that of the real traffic and this adds to the set of reasons for not choosing a traditional Poisson model for modeling real traffic.

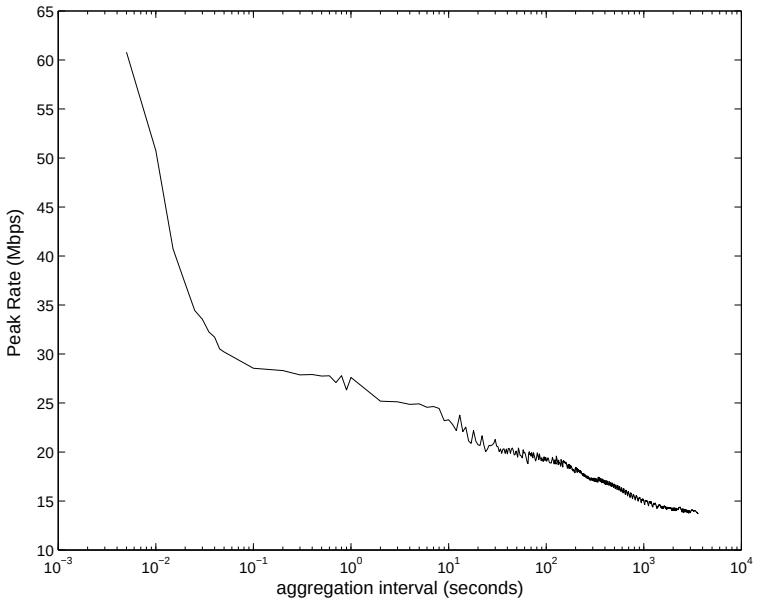


Figure 3.3: Peak Rate on an OC-3 link

3.3 Peak Rate on a Link

We now turn our attention to traffic on an entire OC link. Each day's cell-count data on different links at different switches is analyzed to obtain the peak rate over different aggregation time scales (PRV curve). Link data can be obtained by summing VCC cell counts for all VC/VP pairs on a given OC-x link (where x can be either 3 or 12).

Analysis is done on OC-3 links consisting of tens of VCs. The PRV curve behavior remained approximately the same in all the OC links with a regular pattern indicating similar traffic flow in them [Appendix-PRV of OC-links, Figure 6.1]. Here we consider a particular OC-3 link with 37 VCs. Figure 3.3 shows an expected decay in the peak rate as the aggregation time interval increases on the OC-3 link. The sharpest drop occurs in the region of 10ms-100ms aggregation. Note the difference between peak rate at finer time scales versus peak rate at one-hour time scale. As with VC data, one-hour averaged data does not give an

accurate indication of possible link congestion, which can occur in much smaller time scales.

The maximum peak rate of 60.8 Mbps (aggregated at a lower time scale of 5 ms) and a minimum of 13.7 Mbps for aggregation over 3600 seconds (1 hour) indicating approximately 75% drop in the peak rate. The ratio of maximum to minimum peak rate is 5 : 1, which indicates a rapid decay of peak rate in this particular link. The maximum expected peak rate on any link would be equal to capacity of that link. It is interesting to note that the peak rate is just three-eights of OC -3 link capacity. None of the links (Figure 6.1) reach even half the OC-3 link capacity at the 5 ms aggregation interval. It is assumed that at finer time scales (microseconds), the link might achieve the OC-3 link capacity. The ratio of 5 ms peak rate to 100-second peak rate is equal to the ratio of 100-second (aggregation interval) peak rate to the 1-hour peak rate (Figure 3.3). This indicates a rapid decrease in peak rate from 100-second aggregation and the traffic is much smoother than that at 5 millisecond interval.

Similar Analysis for OC-12 Link

The data on OC-12 links was collected with measurement period of 5 millisecond similar to OC -3 link measurements. One of the OC-12 (Figure 3.4) links consists of 37 VCs. Similar (in shape) peak rate curves were plotted for different OC-12 links. The peak rates (at 5 ms interval) for all OC-12 links were noted [13] to be approximately 175 Mbps and the ratio of maximum to minimum peak rate is also approximately the same. There is a more gradual decrease in peak rate of a link (at aggregation interval of 100 ms) as opposed to the rapid decrease in peak rate in its VCCs (at 10 ms). Similar to the OC-3 links, these links do not reach the OC-12 link capacity of 622 Mbps for 5 ms aggregation but may

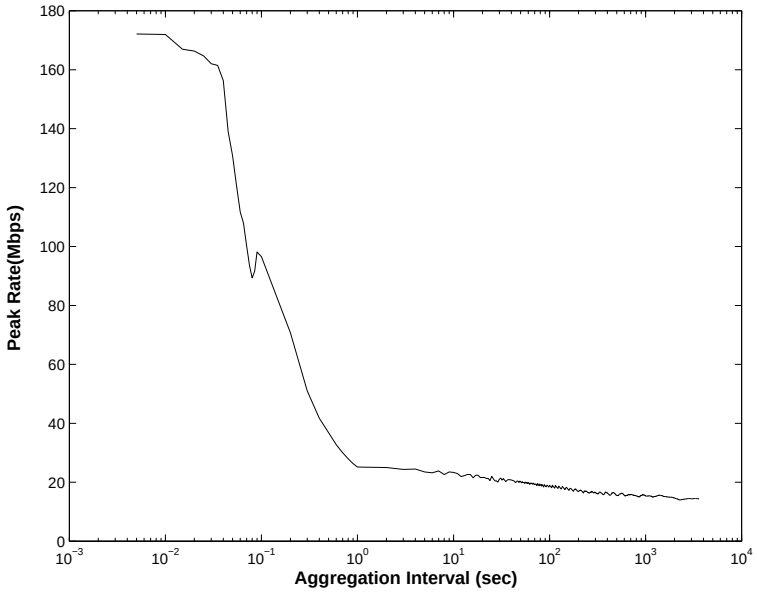


Figure 3.4: Peak Rate on OC-12 Link

reach the maximum link capacity at a much finer time scale. Had all VCs been highly loaded, the peak rate at the 5 ms would have been much higher than that observed in Figure 3.4. There is a rapid fall in the peak rate in VCC traffic at around 100 second aggregation interval and at 100 millisecond interval in OC link traffic. The rate at which peak rate drops from 5 milliseconds to 100 seconds is much less than the peak rate fall from 100 seconds to one-hour aggregation adding another point of difference between OC and VCC curves. Similar traffic pattern was observed in all OC-12 links [Appendix-PRV of OC-links, Figures 6.2-6.3] [13], indicating consistency in the traffic on all the VCCs in OC-12. The peak rates for all the OC-12 links can be noted from Figures 6.2 and 6.3. The peak rates of all OC-12 links are approximately 175 Mbps and the ratio of maximum to minimum peak rate is also approximately the same. For both OC-3 and OC-12 links, the sharpest drop occurs in the region of 10ms-100ms aggregation.

3.4 Ratio of Maximum to Minimum Peak Rates

It is always not possible to store the traffic cell counts at 5 ms granularity due to memory constraints in a switch. For most data networks, network traffic statistics are recorded on one-hour time scale to analyze the traffic characteristics. As we saw in the previous section, the peak rate at the lowest scale is incomparable to the value at the hourly peak rate. So, the time scale is a major factor in determining peak rate. This section discusses the relationship between maximum and minimum peak rates. The previous two sections showed that the coarse measurements significantly underestimate the actual short-term peaks in the traffic. The natural question is therefore, how can we use these coarse statistics to estimate the traffic behavior at a 5 ms time scale. Analysis of the ratio of maximum to minimum peak rate is done in order to show that there is a linear relationship between this ratio and peak rate at one-hour aggregation. Analysis was done on the data derived from the set of OC-3 links and OC-12 links.

Calculation of peak rate based on an hour aggregation for setting parameters for various algorithms used in traffic shaping, performance, etc., would lead to incorrect results. This analysis would help in reducing the percentage error in calculation of various results.

OC Links

Given the one-hour peak rate, the maximum peak rate of OC-links can be estimated by the equations given below:

The linear regression yields the following relationship with a coefficient of determination of $r^2 = 0.853$. Equation for OC-3 link (Figure 3.5 (a)),

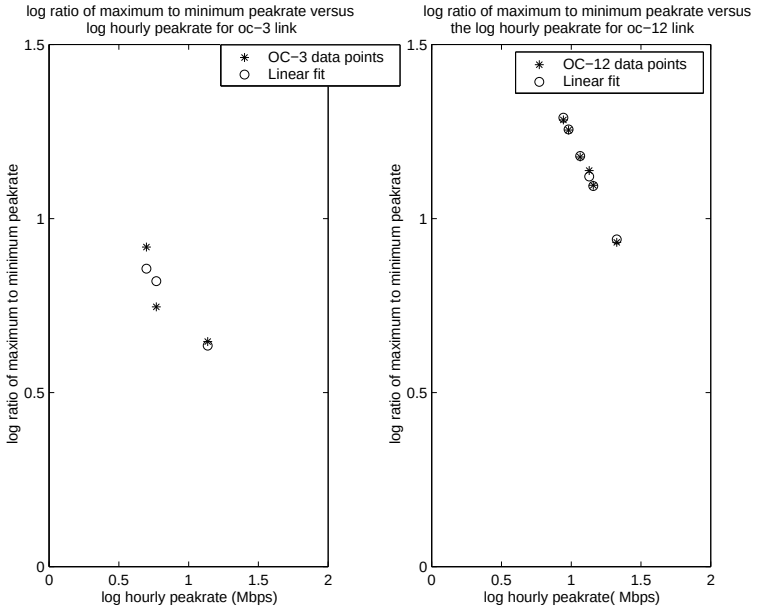


Figure 3.5: Ratio Plots of OC-links.
Sub-plot on the left is 3.5 (a) and on the right is 3.5 (b)

$$Rlog = -0.5032 * oc3log + 1.207 \tag{3.1}$$

Where *Rlog* is the logarithm of ratio of maximum to minimum peak rate and *oc3log* is the logarithm of hourly peak rate (Mbps) of OC-3 link.

Equation for OC-12 link (Figure 3.5 (b)),

$$Rlog = -0.9136 * oc12log + 2.1516 \tag{3.2}$$

Rlog is the logarithm of ratio of maximum to minimum peak rate and *oc12log* is the logarithm of hourly peak rate (Mbps) in equation 3.2. $r^2 = 0.94$.

Given the peak rate at one-hour aggregation of a link, the ratio can be calculated and multiplied by the one-hour peak rate to obtain approximate 5 millisecond peak rate.

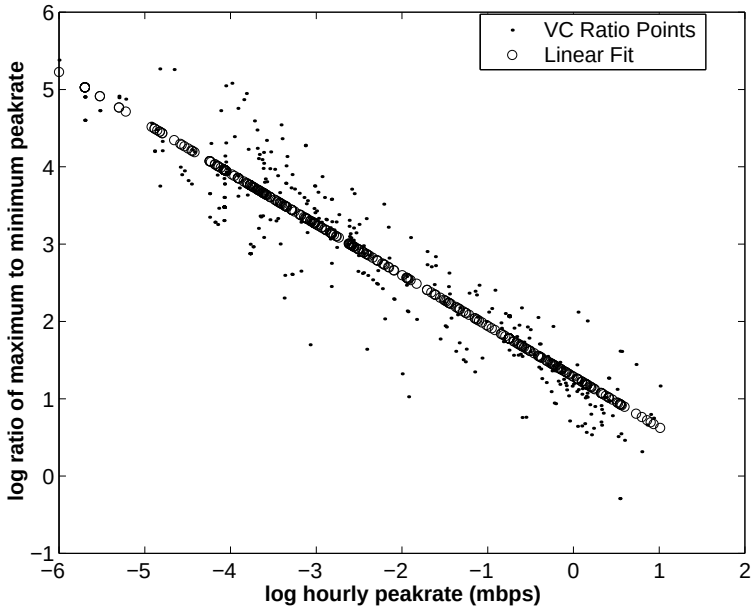


Figure 3.6: Ratio Plot of VCs

VCCs on OC Link

We now investigate the ratio of the maximum to minimum peak rate of the VCCs, in particular as a function of hourly (minimum) peak rate. A specific pattern is typical of all the VCCs in all the links. Certain VC's carried heavy traffic and it resulted in the high peak rate in its link. All the links have a regular pattern indicating similar traffic flow in them [13].

Finally, an equation can be obtained for the peak rate of a VCC on an OC-link. Figure 3.6 plots the logarithm of ratio of maximum to minimum peak rate of all VCCs combined sketched on a single plot. The resulting linear fit (equation 3.3) can be used to characterize the behavior of VCCs in the OC-links. The equation of the straight-line fitted to the cluster of ratio points with coefficient of determination, r^2=0.85, is,

$$rlog = -0.6565 hourpeaklog + 1.287 \tag{3.3}$$

Where *rlog* is the logarithm of ratio of maximum to minimum peak rate and *hourpeaklog* is the logarithm of hourly peak rate (Mbps) of all the VCCs.

Chapter 4

Approach, Models and Analysis

4.1 Traffic Characterization

A traffic model is a stochastic process that includes a set of parameters. Given a realistic traffic trace, a traffic model can be considered to be accurate if its queuing performance is similar to that of the trace. Using inaccurate models may result in over-engineering (low efficiency) or under-engineering (poor performance). Considering modeling and analysis of traffic loads in high-speed networks, a huge set of arrival processes with different short- and long-term correlation structures have been developed and numerous light- and heavy-tailed distributions describing the underlying random variables of the load models have been identified. Recently, there were attempts to estimate the parameters of such models from real data [29]. This complex correlation structure that spans across wide range of time scales usually called long-range dependence is not taken into consideration in traditional Markovian models. Various studies have indicated [1, 7, 15, 19] that LRD has significant impact on resource management and network performance evaluation. Current network traffic trace phenomenon suggests that self-similar

modeling is better than Poisson modeling. The central idea of traffic modeling is to construct analytical models that capture perhaps not all statistics, but ones that are important for performance analysis.

Understanding the nature of traffic, identifying its characteristics and building practical models are vital for the tele-traffic engineering of today's packet switched networks. New observations of measured traffic call for new approaches (e.g., multiple time scale characterization) and the ever changing services and protocols of the Internet trigger particular models (e.g. Ethernet models, WWW models). This chapter overviews the traffic characterization and modeling activities of this thesis, presenting different models developed for packet traffic.

4.2 Previous Research

4.2.1 Markov Models

The long held paradigm was that network traffic could be adequately described by Markovian models (e.g., Poisson) with sufficient accuracy. This modeling can be used if the network traffic shows a little or no auto-correlation. The memoryless property in Markov models demand that the time spent in a state is distributed exponentially with a particular mean arrival rate corresponding to that state. The models available in literature for modeling traffic (like MMPP) can be considered traditional. These models assume arrival rate process to have finite mean and finite variance unlike that expected for LRD traffic. The basic assumption of arrival process being Markovian ignores the significant correlation present in the network traffic. A very large state space is needed for capturing the complex network behavior, and the use of Markov models seems problematic.

The Poisson process has traditionally been used to model traffic. The inter-

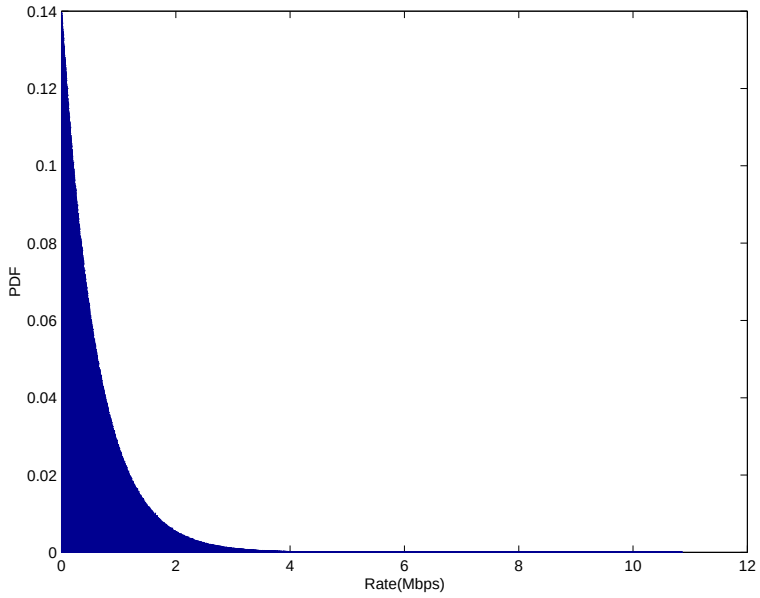


Figure 4.1: PDF of Exponential Distribution

arrival time distribution is exponential for a Poisson count process. The pdf for an exponential distribution is shown in Figure 4.1. This results in a data where the bursts smooth out as the aggregation interval increases (Figure 2.3).

If traffic were to follow a Poisson arrival process, it would have a smooth traffic characteristic when considered at large time scales. Poisson traffic aggregates well over time, implying that peaks in the arrival process tend to be canceled out rapidly by the succeeding dips. Also, the burstiness is restricted due to high degree of multiplexing as aggregation of traffic from multiple number of Poisson sources causes a smooth Poisson traffic stream. Non-Poisson traffic fails to aggregate like the well-behaved Poisson traffic (peak rate of the link, Figure 3.3), where the resulting aggregated non-Poisson traffic is still highly bursty. A property that is worth noting for Poisson process, is that the values of the random variable representing arrival process at different time scales are uncorrelated. This is explained by the fundamental memoryless property of Poisson. Therefore, it does not fit into the modeling of self-similar traffic, where the random variables

are correlated. The main purpose of next section is to investigate theoretical distributions that approximate the empirical distribution of the measured cell inter-arrival distribution.

4.3 Models for Packet Traffic

4.3.1 Markov Modulated Poisson Process (MMPP)

A Markov-modulated Poisson process (MMPP) is a doubly stochastic process, that is, a Poisson process with an intensity changing in time in accordance with another Markovian process. An MMPP can be modeled as a continuous time Markov chain, with state space $\{1, ..., k\}$. We say that the MMPP is of order k, and each of the k states corresponds to an arrival rate λ_i, when the chain is in state i. We consider a two-state MMPP where the mean sojourn times in state 1 and state 2 are α^{-1} and β^{-1} respectively. Using the notations used in previous sections for the cumulative point processes $(N(t))$, mean of the counting process is defined as,

$$E[N(t)] = \frac{(\lambda_1\beta + \lambda_2\alpha)t}{\alpha + \beta} \tag{4.1}$$

From [4],

$$H_v(t) = 0.5\frac{1 + A[1 - (1 + \rho t)e^{-\rho t}]}{(1 + \rho A)t - A(1 - e^{-\rho t})} \tag{4.2}$$

Where $\rho = \alpha + \beta$ and $A = 2\alpha\beta(\lambda_1 - \lambda_2)^2/(\rho^3(\lambda_1\beta + \lambda_2\alpha))$.

This model captures randomness of arrivals across sources, but it fails to capture complete variability though a closed form analysis [4]. There are just

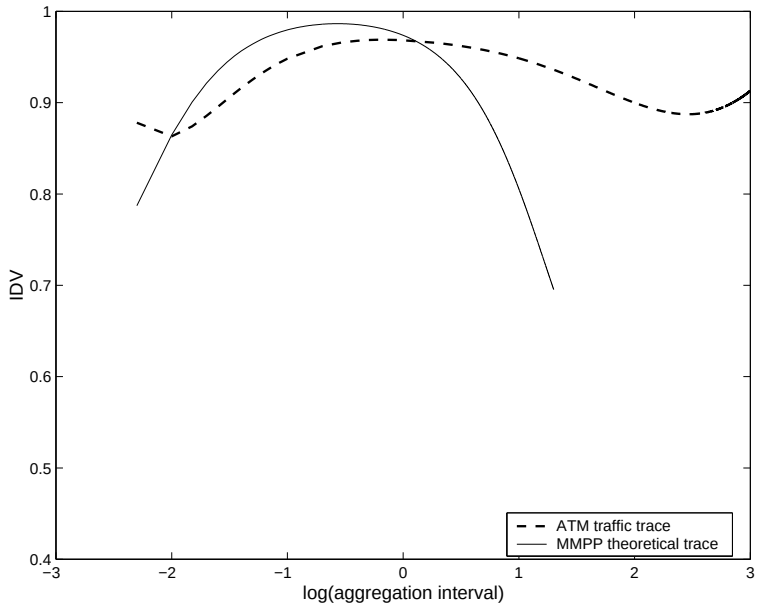


Figure 4.2: Comparison of ATM Traffic Trace IDV and MMPP Theoretical IDV

four parameters involved in a two-state MMPP and three degrees of freedom if the mean arrival rate is matched. It was observed during the analysis that three parameters were not sufficient to match the IDV curve as it could not match the bi-modal nature of a typical ATM traffic trace. This two-state model captures variability with a small number of parameters but is incapable of matching the IDV of a real traffic trace (Figure 4.2). Just by heuristics, the parameters were varied to find an IDV curve that is closest to ATM traffic trace IDV. One of the arrival rates should be less than the mean arrival rate and the other arrival rate should be greater than the mean. The space to search for one of the arrival parameters is constrained, leaving the difficulty to search the other arrival rate. A Matlab program was used to generate various plots by varying the parameters. The closest match was visually observed to find the matching IDV plot. The theoretical trace (solid line in Figure 4.2) was the closest that could be obtained using the MMPP model.

Multiplexing of several MMPP sources could be done but it involves the process of setting parameters for the states. Though it is easy to build a simulation model, it would be tedious to manipulate parameters and observe the variation in the IDV trace. There was an effort to generate synthetic traces using the parameters from the two-state MMPP analysis by building a model in Extend, a simulation tool. Long traces (order of gigabit) are needed to achieve a close match to the real traffic trace. With short length traces (a few Mega Bytes of data), there will be less number of points to calculate variance and it is erratic to estimate the correct IDV of the trace. Another reason for not attempting to continue this model is a minor problem in the simulation model. In simulation, the transition from one state to another has to synchronize with the change of arrival rates. There was some delay in change of arrival rates. The percentage error in the final mean rate of the synthetic trace will be less if a long trace is generated. The major constraint, however, is the memory to store the synthetic trace data, so short traces had to be generated, leading to mismatch of IDV due to insufficient data points. Also, while arrival processes based on such models can be described using a few parameters, it is analytically very difficult to analyze a queuing system for such a process as input.

4.3.2 Renewal Process Model for Inter-arrival Distribution : Hyperexponential Distribution

A renewal process involves recurrent patterns connected with repeated trials. This best suits to describe self-similar models as self-similarity is the the invariance of an intrinsic pattern under scaling. Hyperexponential can be categorized as one such renewal process [8]. Previous studies in earlier sections revealed

that network traffic exhibits burstiness, that is variability, over multiple time scales. In many circumstances, heavy-tailed distributions have been appropriate for capturing variability because of the slow decaying property of those probability distributions compared to the ones belonging to the exponential family and the heavy-tailed property of the bursts. This implies that the length of the burst is highly variable i.e, exhibits variability over a wide range of time scales. We have shown that the aggregate process (counting) of network traffic data is long-range dependent and hence self-similar. Collected ATM data is in the form of events in successive intervals of fixed length whereas the estimation of the renewal function to calculate the IDV is based on the observation of the inter-arrival times between the events of interest. Cox's construction by a renewal structure [8] for inter-arrival time distribution is straightforward, since it requires only that inter-arrival times are i.i.d and variance can be obtained from the counting process.

A random variable X has a heavy tailed distribution if its complementary cumulative distribution (ccdf) $F'(t)$ satisfies

$F'(t) = Pr\{X(t) > t\} \sim ct^{-\alpha}$ as $t \to \infty$ ($f(t) \tilde{} g(t)$ means $(f(t)/g(t)) = 1$ as $t \to \infty$)

where α and c are positive constants. A common situation is $1 < \alpha < 2$ for which the random variable X has finite mean and infinite variance. This gives rise to long-range dependence, i.e, non-summable autocorrelation function. Heavy tailed distributions have high or even infinite variance and therefore show extreme variability on all time scales. Distributions with infinite variance lead to self-similarity. Recent studies have shown evidence indicating that the aspects of communication and computer systems can show heavy-tailed distribution [35, 36]. They also highlighted the predominance of heavy tails in arrivals. The analysis

of existing measurements of high speed network traffic by statistical methods has shown that the characteristic random variables are often heavy-tail distributed or even follow mixtures of heavy-tailed distributions [30]. There is a class of sub-exponential distributions like Pareto and Weibull that could also be used for modeling [10] but their Laplace transforms are not tractable for queueing analysis. The Laplace transform makes it possible to analyze the performance models by numerical inversion [33]. Derivation of IDV involves solving the Laplace transforms and their integrals. There is no convenient Laplace expression for Pareto and Weibull distributions and hence it is difficult to use queuing models like G/G/1. For any number of phases of hyperexponential, Laplace expression is easy to calculate. This is also another reason for the choice of hyperexponential distribution for modeling. Also, since we have finite variance, hyperexponential models are best suited for tractable models. Hyper-exponential could be used to match the heavy-tailed distributions [17]. But, matching Pareto or Weibull inter-arrival probability density function (pdf) with hyperexponential pdf requires various approximations as pdf needs to be calculated from continuous process. We can calculate probability mass function from the data and approximate it to pdf. The following approximations result in failure of fitting a hyperexponential to a heavy-tailed distribution:

1. Calculating real traffic data pdf from pmf of the traffic data.
2. Fitting Pareto/Weibull pdf to the traffic data pdf.
3. Fitting hyperexponential pdf to the Pareto/Weibull pdf to find the parameters of hyperexponential distribution.

Every data set cannot be fit using Pareto/Weibull distribution. Also, these numerous approximations involved in calculating the parameters of hyperexponen-

tial lead to inaccuracy in parameter values [17]. We tried to fit hyperexponential pdf to the data pdf but were unsuccessful due to inaccuracy involved in parameter values due to above mentioned approximations.

Highly bursty traffic can be generated using the hyperexponential inter-arrival time distribution, thus providing a basis for modeling traffic varying over multiple time scales. A parametric approach is discussed in later sections to estimate the IDV of hyperexponential distributions and to cope with the data analysis in a highly variable environment such as the ATM.

One of our goals was to show that the hyperexponential distribution inter-arrival time can be used to emulate the data in its peak rate characteristics and self-similarity through IDV curve. This requires search procedures and optimizations to find the parameters of hyperexponential (discussed in section 4.4).

4.3.3 Derivation of IDV for H_n

Let E_i,$i = 1, 2, ..., n$ be n independent exponential random variables each with parameter λ_i, $i = 1, 2, ..., n$, where $\lambda_i \neq \lambda_j$ for $i \neq j$. Suppose that there are n positive constants w_i for $i = 1, 2, \ldots, n$ such that

$$\sum w_i = 1 \tag{4.3}$$

If the random variable $H_n = E_i$ with probability w_i, then X is a hyperexponential random variable with n exponential stages (or order n) and parameters w_i,λ_i, $i = 1, 2, ...n$. The probability density function of H_n is:

$$f_n(X) = \sum_{i=1}^{n} w_i \lambda_i e^{-\lambda_i x} \tag{4.4}$$

$$E[H_n] = \sum_{i=1}^{n} \frac{w_i}{\lambda_i} \tag{4.5}$$

$$\lambda_{H_n} = \frac{1}{E[H_n]} \tag{4.6}$$

For example, an order 3 hyperexponential pdf is:

$$f_3(x) = w_1\lambda_1 e^{-\lambda_1 x} + w_2\lambda_2 e^{-\lambda_2 x} + w_3\lambda_3 e^{-\lambda_3 x} \tag{4.7}$$

$$E[H_3] = \frac{w_1}{\lambda_1} + \frac{w_2}{\lambda_2} + \frac{w_3}{\lambda_3} \tag{4.8}$$

For an n^{th} order hyperexponential, there are $2n-2$ degrees of freedom as two parameters are determined using the equations (4.3) and (4.6). Therefore, there are four degrees of freedom for an H_3 distribution.

An analytical expression for IDV was derived for hyperexponential distribution in [4]. The derivation involves finding the variance of the counting process since the underlying process is assumed to be a point process. The result from [4] for the H_3 is:

$$Var[N(t)] = 2\lambda \int_0^t \phi(u)du + \lambda t - \lambda^2 t^2 \tag{4.9}$$

where $\int \phi(t) = L^{-1}[\frac{f_3^*(s)}{s(1-f_3^*(s))}]$.

L^{-1} implies inverse Laplace transform and $f_3^*(s)$ is the Laplace transform of $f_3(x)$.

$$f_3^*(s) = L[f_3(x)] = w_1(\frac{\lambda_1}{\lambda_1 + s}) + w_2(\frac{\lambda_2}{\lambda_2 + s}) + w_3(\frac{\lambda_3}{\lambda_3 + s}) \tag{4.10}$$

Solving equation we get,

$$\nu(t) = L^{-1}[\frac{f_3^*(s)}{(1 - f_3^*(s)}] \tag{4.11}$$

and hence we can get $\phi(t)$ as

$$\phi(t) = \int_0^t \nu(u)du \tag{4.12}$$

Now IDV can be calculated using the equation 2.6. An analytical expression was derived for H_2 in [4] and we derived IDV for H_3 distribution [Appendix]. After this basic step of deriving IDV, we observe the different IDV curves using the H_3 distribution. It was observed that the IDV spans across a few orders of time scales proving the high variability. We classify the hyperexponential into two categories where the degrees of freedom are reduced.

1. Balanced hyperexponential.

2. Doubly balanced hyperexponential.

These different hyperexponential distributions demonstrate convincingly that multiple time scale traffic can be modeled by appropriate choice of H_n parameters.

4.3.4 Balanced Hyperexponential

When the ratio of weights to arrival rates are equal i.e., $\frac{w_i}{\lambda_i} = \frac{1}{n\lambda}$, the distribution is called a balanced hyperexponential distribution. For the H_3 distribution, this reduces another two degrees of freedom leaving us with two degrees of freedom. By fixing one variable and varying the other, we get a highly variable traffic (Figure 4.3), over a broad range of time scales. Figure 4.3 is a set of unimodal H_3

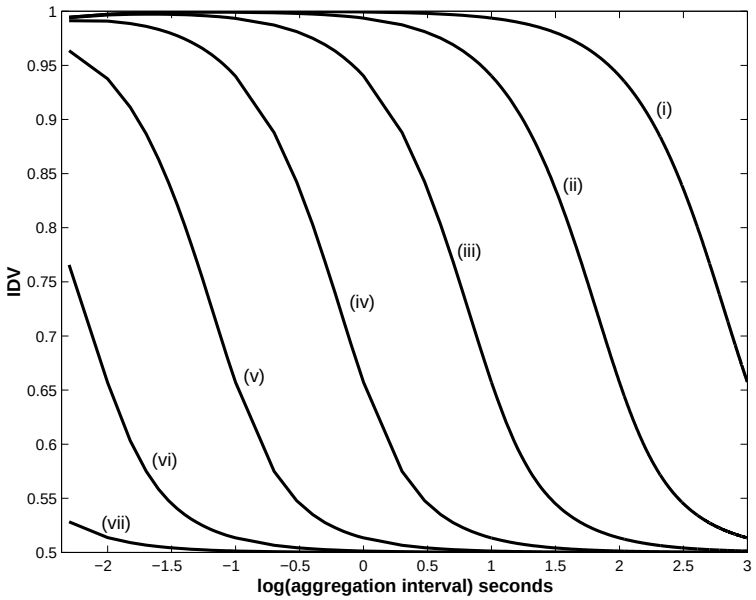


Figure 4.3: Balanced H3

IDV curves where $w_2 = 0.75$, $1/\lambda = 1/5183$ and w_1 is varied from 0.1 to $10e^{-8}$ (moving towards left, i.e., from (i) to (vii)). By decreasing w_1, we decrease λ_1, implying the increase in mean inter-arrival time of the first term in the f_3. As a consequence, IDV being nearly unity spans across several orders of magnitude. Notice that the curves decay to 0.5 eventually.

4.3.5 Doubly Balanced Hyperexponential

Now, in addition to the requirements of the balanced hyperexponential, suppose we fix the ratios of weights i.e, $\frac{w_i}{w_{i+1}} = k$, where $0 < k < 1$ for $i = 1, 2, \ldots n-1$. We call this a double-balanced H_n. Now for the H_3 distribution there is only a single degree of freedom. A bimodal IDV can be generated by using this model. Value of k controls the variability at all time scales. When k is decreased (see k=$1e^{-04}$ in Figure 4.4), two of the weights are increased leading to higher values of their corresponding $\lambda's$. This implies that short inter-arrival times occur frequently

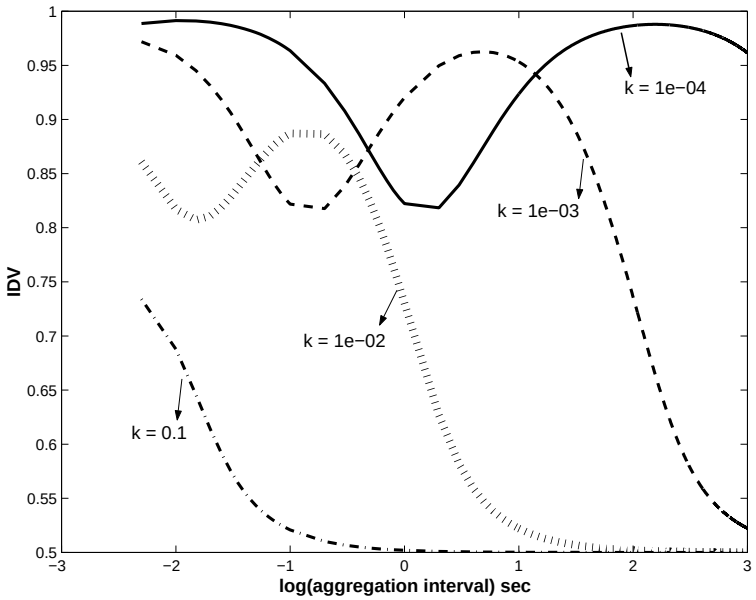


Figure 4.4: Doubly Balanced H_3

leading to bursts between large inter-arrival times. This increases variance in the "counting process" with large values of IDV.

4.3.6 Matching Real Network traces

H_3 can be used for modeling self-similar traffic by matching the mean and the IDV of the real traffic trace data. There are infinite solutions for the parameter values by just matching the mean value of the trace data. Here we attempt to match the trace IDV reasonably by heuristic method and optimization technique.

Heuristic Method

The heuristic method is a tedious and inefficient method of finding the parameter values. This method uses the general information of the relation between parameters to find "good" approximate values for the parameters. The generation of IDV involves the use of a heuristic, or a combination of several equations proceeded

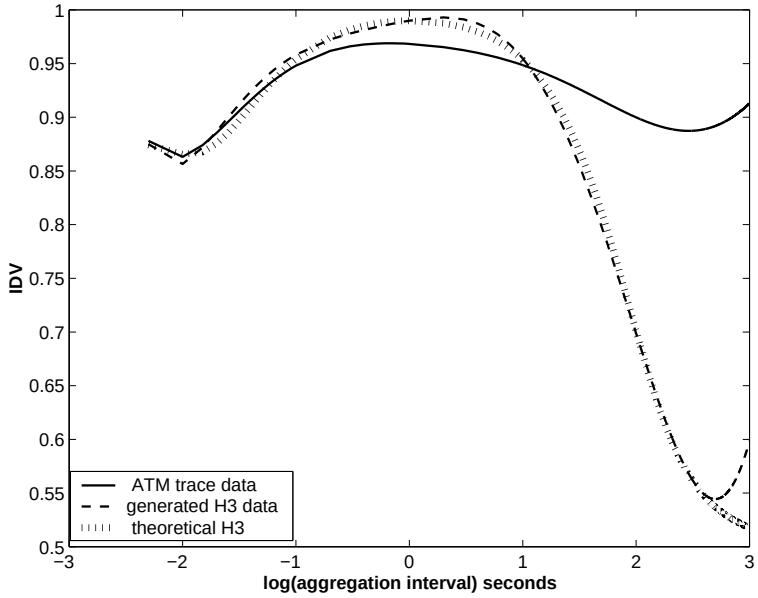


Figure 4.5: IDV of Real traffic trace, Synthetic trace and Theoretical H_3 Parameters for generating synthetic trace: w_1 = 0.72e-06, w_2 = 0.0063, w_3 = 0.9937, λ_1 = 0.0246, λ_2 = 221.21 and λ_3 =7351.

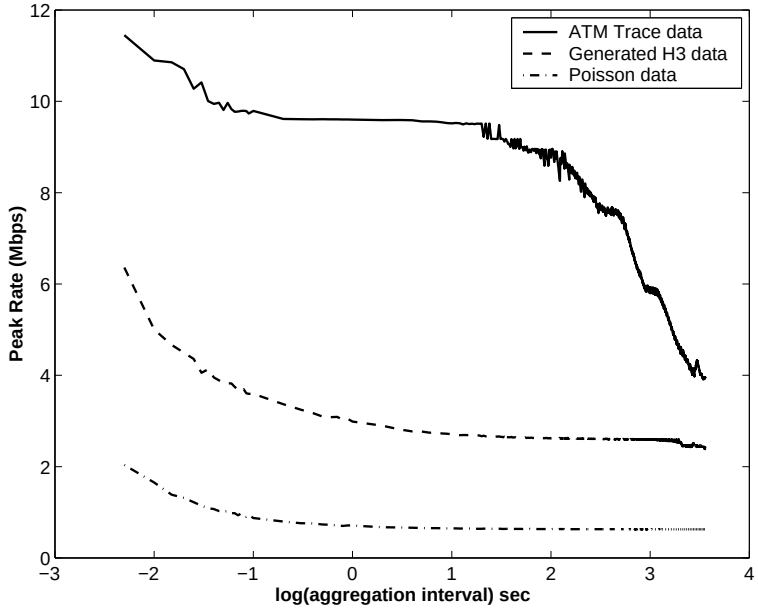


Figure 4.6: PRV of Real traffic trace, Synthetic trace and Poisson traffic

sequentially to calculate the final result. Our approach uses pattern matching where the pattern is an IDV of a real network trace. In a doubly balanced H_3, there is only one independent variable (assume it to be λ_1) and the other H_3 parameters are constrained. Equating the mean of the H_3 to the mean of the real network traffic trace, the independent variable (λ_1) is changed. The change in the variable gives numerous curves and a curve nearest to the trace IDV curve is chosen. Now this independent variable (λ_1) attains a constant value. All the parameters of H_3 are now fixed. Now, each of the parameters (other than λ_1) is varied to check if we can get any better match to the real traffic trace IDV. The basis for this decision is to minimize the maximum magnitude difference between the trace IDV and generated H_3 IDV (observed visually). A smaller magnitude implies a better match.

Using the heuristics, the IDV was matched for most of the time scales (Figure 4.5). Though we did not succeed in matching the queuing performance analysis of the synthetic trace generated using the parameters obtained from heuristic method to the performance of the real data trace, the synthetic trace IDV matched the real trace IDV at various time scales (further discussed in chapter 5). Another drawback of this method was the mismatch of the PRV curve (Figure 4.6). The synthetic trace (dashed line) is not as bursty as the real traffic trace as the peak rate at 5 ms is approximately half the peak rate of the real trace 5 ms-peak rate (solid line). An advantage of the heuristic approach is that the sequential classification of the equations are explicitly defined implying that the parameters can be controlled without using any random approach. Therefore, software problems in implementing this approach can be easily found out.

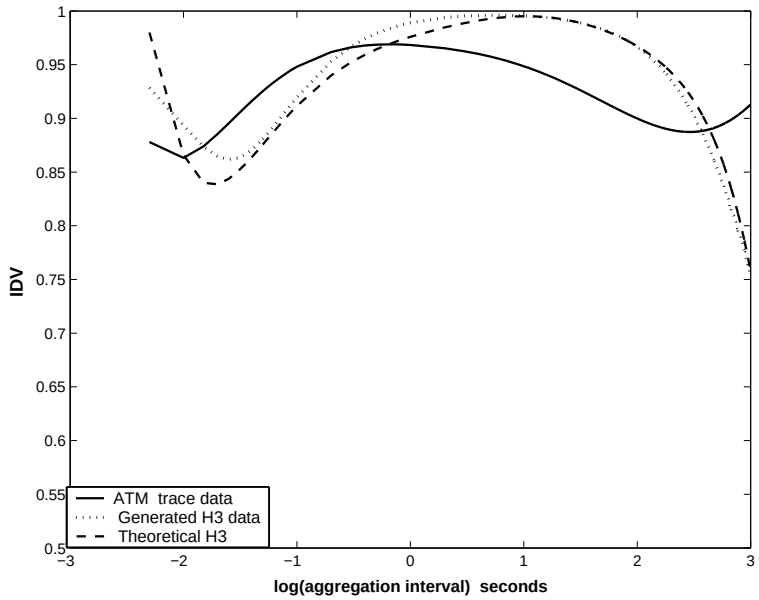


Figure 4.7: IDV curve comparing Real trace, H_3 generated data,Theoretical H_3 Parameters for generating synthetic trace: w_1 =0.003701, w_2 = 0.9962989, w_3 = 1e-07, λ_1 = 82.821, λ_2 = 19242 and λ_3 = 0.01037.

Optimization Technique

The IDV expression is non-linear with multiple local minima. There are very few solvers dealing with non-linear constrained optimization problems. One of such solvers, AMPL [26] is used for matching the IDV of the trace data to the IDV of H_3 generated data. The objective function was to minimize the maximum magnitude difference between the two curves. There was a reasonably good match between the two IDV curves and the performance analysis results (mentioned in chapter 5).

Figure 4.7 shows the IDV match between ATM traffic data, synthetic H_3 data and the theoretical H_3 calculated from H_3 IDV expression. Though the match between real ATM trace and synthetic trace is not exact, the latter and the theoretical H_3 IDV curves match very well. The peak rate curve (Figure 4.8) implies that a highly bursty data can be generated using H_3 where the peak rate

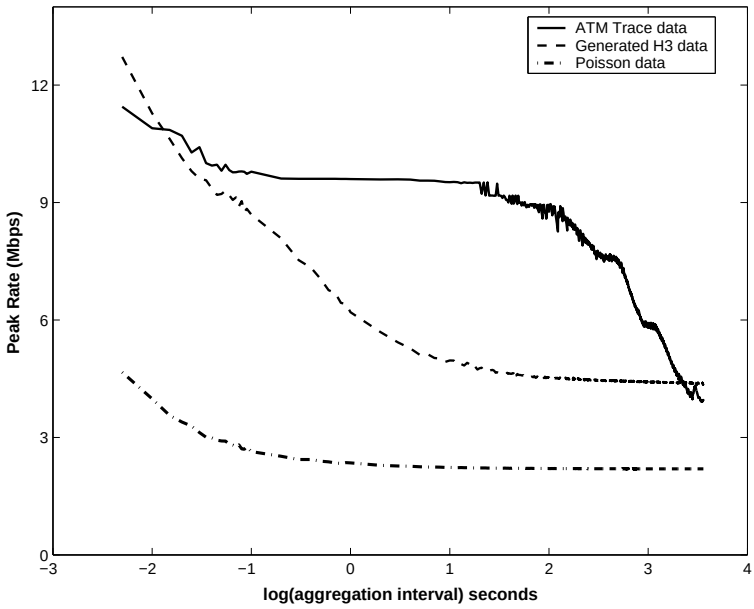


Figure 4.8: Peak rate curve comparing real trace, H_3 generated data, Poisson traffic

at the lowest time scale matched pretty well. This also shows that a higher order hyperexponential can be used in modeling to get good results to imitate the real data in terms of performance.

4.4 Optimization Technique

AMPL optimization package is used for finding the 2n parameter values in the n^{th} order hyperexponential model. The parameters in the hyperexponential are the arrival rates and the weights associated with the arrival rates. One of the linear constraints is the summation of weights to unity. The other constraint is to satisfy the mean arrival rate which is the reciprocal of summation of ratio of weight to corresponding arrival rate. The package uses a reduced gradient approach where the algorithm manipulates the initial parameter values satisfying the linear constraints. An iteration attempts to reduce the objective function

within the subspace of the variables. Variables here are the results of non-linear math equation involving the variable parameters. When no further progress can be made with the current variable, AMPL displays the result.

Outline of the Problem

We describe an attempt to match the IDV curve obtained from a trace with known mean arrival rate. The IDV values are calculated at multiples of the lowest aggregation interval (5 milliseconds) or basic data measurement interval.

Constraints

1. The mean arrival rate (λ_{mean}) should be matched. $\sum_i \frac{w_i}{\lambda_i} = \frac{1}{\lambda_{mean}}$.

2. Weight sum should be unity. $\sum w_i = 1.0$.

3. The weight should not be less than millionth of a unit. $w_i > 1e^{-07}$, $\forall i$.

4. The parameter should be greater than the thousand of a unit. $\lambda_i > 1e^{-03}$, $\forall i$.

Model

Input of the AMPL requires the initial parameter values and IDV values to be matched. All the IDV values calculated from trace are entered as an input to the AMPL. There are around 1000 points which is fairly a large data set to be matched. The objective function f is to minimize the maximum magnitude between the IDV generated using the AMPL parameter values and the IDV from trace.

$$f(x) = max(|IDV_{trace}(1) - IDV_{H_3}(1)|, |IDV_{trace}(2) - IDV_{H_3}(2)| \ldots |IDV_{trace}(m) - IDV_{H_3}(m)|)$$

Preprocessing and Data Handling

The preprocessing of an optimization problem can not result in a substantial reduction of the computation but is necessary for solving the problem.The initial values for the 2n-1 parameters must be entered. The AMPL is set-up to choose the initial parameter values using random method, but this allows the possibility of generating initial parameters that might lead to non-convergent solution. So, depending on the mean arrival rate, the initial values are scaled and certain bounds are applied. There are '2n' parameters in H_n distribution (n $>$ 0). If 'n' is odd, set bounds on $(\frac{n+1}{2})$ arrival rate parameters such that each of them is less than the mean arrival rate. The weights corresponding to arrival rates lesser than mean, should have weights greater than 1/2. The rest of the arrival rate parameters $(\frac{n-1}{2})$ should be set such that they are more than mean arrival rate with their corresponding weights set to weight less than 1/2. If 'n' is even, half of the arrival rates should be set greater than mean and rest should be set lesser than mean. This allows AMPL to chose meaningful values and get a good starting point for optimization.

Solving the model

A large number of iterations with different random seeds used in setting the parameter values gives a better match with the real traffic IDV trace data values. The output of the AMPL optimization writes data into a file in AMPL format (a specific pattern for storing parameter values). The output file is parsed using the *awk* program to get the H_3 parameter values in a 'matlab input format'. The values are used in generating IDV plots to compare the best match with the original trace IDV.

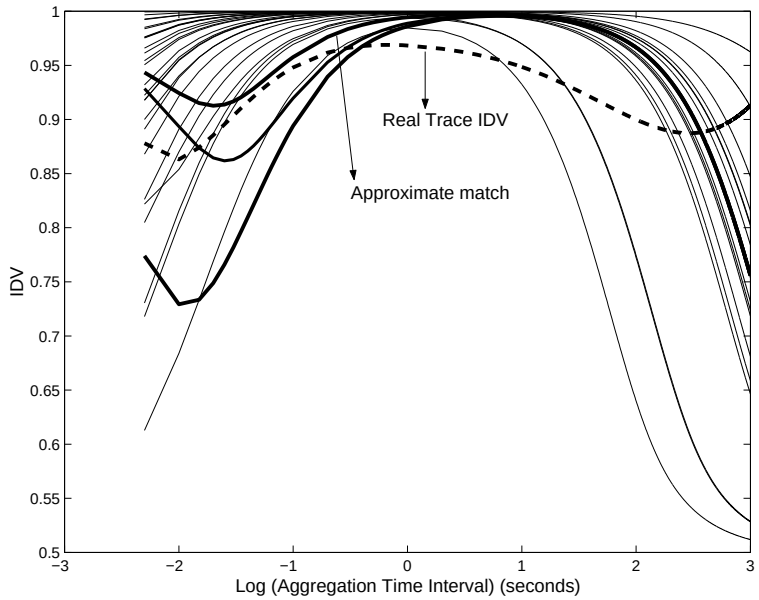


Figure 4.9: Curves for Matching the Real Trace IDV using AMPL

Model analysis

The optimization problem has practically non-unique solution (i.e there are many very different solutions having almost the same value of the goal function). Therefore a graphical technique has been used to provide a unique solution having some additional properties. We need to somehow measure how good the estimates are. One possibility is to compare the IDV curve of real traffic with that of the generated synthetic traffic trace. The highlighted curves in Figure 4.8 are quite close in shape at lower aggregation intervals. One of the highlighted curves with the least objective function is chosen to be the closest match. Matching the values at the smaller aggregation levels makes sense as the IDV values are based on the calculation of variance. Variance estimates are better when there are more number of points (smaller aggregation levels have large set of points). Second possibility is to consider the queuing behavior and compare the queue with model as input. The queuing behavior is discussed in chapter 5.

4.4.1 IDV Derivation for Hyperexponential of Higher Orders

IDV derivation involves the calculation of Inverse Laplace transform and integral of the result. There is a potential difficulty in calculating the inversion of Laplace transform in case of IDV derivation. Not all Laplace transforms have inverses, especially in the derivation of IDV. The derivation involves polynomials in both denominator and numerator which need to be factorized partially to apply inverse Laplace transforms. Though not impossible, an analytical expression is difficult to derive for H_n with n $>$ 3. There are a number of numerical inversion methods but they require pre-assignment of values to parameters. This restricts evaluation of the correct IDV as the numerical inversions embed assumptions [39] that could affect the IDV value drastically. This was the reason that we did not attempt to analyze higher orders of hyperexponential.

Chapter 5

Performance Evaluation of Self-Similar Networks

High speed networks will be required to carry a broad range of traffic classes ranging from bursty, variable bit rate traffic to smooth, constant rate traffic, while satisfying the QoS requirements. Multiple time scale traffic is characterized by considerable fluctuations in the traffic rate, well above or under the average rate, over several time scales. Characterizing is important in dimensioning and design for such traffic. Design requires determining link capacities, buffer sizing and processing capacity of switches. It has been shown [34] that self-similar traffic heavily impacts the queuing behavior of the system, and ignoring this aspect leads to underestimation of loss probability and buffer sizes. In this chapter, we try to provide analytical tools and techniques for characterizing the properties.

5.1 Effect of High Variability Traffic on a Queue

ATM is connection oriented packet-switched mode of transfer using 53 byte cells. All cells belonging to the same connection follow the same path along the network. Available ATM data was easier to analyze as the cell size was fixed and inter-arrival time could be calculated easily. The performance for a queuing system with LRD input can be radically different from performance of a traditional Markovian system [37]. As the number of LRD traffic sources increases, the aggregate traffic becomes burstier than individual traffic streams. Traditional analytical approaches towards performance evaluation cannot be applied to such networks. The main performance metrics of interest are delay in the network and loss probability. Little is known about the finite buffer and packet loss rate except for observations like the relevance of time scales and correlation structure at larger time scales [20].

Queuing analysis of Poisson traffic has been observed where the queuing behavior seemed insensitive to marginal properties of traffic [20]. As mentioned earlier, there are various models described for packet traffic where arrival processes are based on models like Chaotic maps or Fractional Brownian motion, but it is difficult to analyze the queuing system. So, a tractable model, the hyperexponential inter-arrival model is chosen in this thesis. We next undertake a queuing analysis using the G/M/1 queuing model.

5.1.1 G/M/1 Analysis

G/M/1 is a queuing situation in which the arrival pattern is unconstrained but service times are exponentially distributed. The state of the G/M/1 queue consists of two parts, a continuous and a discrete part, i.e, the state is given by

the number of customers in the system and the time until the next arrival. An embedded Markov chain is formed for the continuous part [27]. The standard G/M/1 queue analysis applies where new arrivals find a system containing exponentially distributed amount of work. This implies that the mean and standard deviation of the packet delay are identical. In [28], a geometric parameter β is calculated from which the queuing delay can be calculated. Let $f_3(t)$ be the pdf of the order 3 hyperexponential inter-arrival times. L(s) is the Laplace transform of the interarrival time distribution. μ is the mean service rate and β is a geometric factor calculated from equation 5.3. We apply the technique [28] to H_3 as follows:

The pdf of the interarrival times is given by:

$$f_3(x) = w_1\lambda_1 e^{-\lambda_1 x} + w_2\lambda_2 e^{-\lambda_2 x} + w_3\lambda_3 e^{-\lambda_3 x} \tag{5.1}$$

Taking the Laplace of the above,

$$L(s) = L[f_3(x)] = w_1(\frac{\lambda_1}{\lambda_1 + s}) + w_2(\frac{\lambda_2}{\lambda_2 + s}) + w_3(\frac{\lambda_3}{\lambda_3 + s}) \tag{5.2}$$

The geometric factor β can be calculated using this simple expression:

$$\beta = L(\mu(1 - \beta)) \tag{5.3}$$

Replace s by $\mu(1 - \beta)$.

$$\beta = w_1(\frac{\lambda_1}{\lambda_1 + \mu(1 - \beta)}) + w_2(\frac{\lambda_2}{\lambda_2 + \mu(1 - \beta)}) + w_3(\frac{\lambda_3}{\lambda_3 + \mu(1 - \beta)}) \tag{5.4}$$

By solving this polynomial equation in β, we get various solutions depending

on the degree of the polynomial. For H_3, we get four solutions for β. Choosing the value less than one, we can find the mean queuing delay ($E[Q]$)and the mean system delay ($E[T]$).

$$E[Q] = \frac{1}{\mu(1-\beta)} \tag{5.5}$$

$$E[T] = \beta E[Q] \tag{5.6}$$

In order to make comparisons with the analytic results, we associate exponentially distributed service times with the arrivals listed in the H_3 and the real traffic trace files, even though the real traffic trace was gathered from an ATM (fixed packet size) link. Also, the trace files list number of arrivals in each 5 ms interval, so the simulation spaces each set of arrivals evenly throughout the associated 5 ms interval.

5.1.2 Synthetic Hyperexponential Data Generation

Computer simulation is a standard tool for the verification process in network analysis. Simulation has been used to create artificial 'data traces'. There are valid reasons for using a simulated data rather than real data. The gigabit size data files render simulation ineffective for systems of realistic size. Mainly, it is less expensive in terms of storing the data (memory constraints), where a few parameters could be stored to generate a traffic imitating its behavior.

Artificial network traffic generation should simulate the stream of packets on several different levels of description. Such stream of of packets is characterized by sequence of observations,

..,$X(t_{n-1})$,$X(t_n)$, $X(t_{n+1})$, ...

at time points

$....., t_{n-1}, t_n, t_{n+1}, ...$

These observations can be described as the inter-arrival times between packets. $X(t_i)$ is described by a family of random variables with known probability distribution function and time index t.

Complementing the theme of traffic modeling is the issue of simulation, such as generation of synthetic traffic traces. For validating the source model, synthetic data with hyperexponential service algorithm times was generated with known arrival rate. Hyperexponential inter arrivals times for the cells is constructed based on the algorithm described in [32]. The method is based on the principle that a random variable with any arbitrary probability density function can be generated, by applying a simple transformation to a uniform random variable varying between zero and one. This random variable is then used to weight the exponentially distributed random variable with mean equal to one of the parameters of hyperexponential distribution that maps to the weight parameter.

Since we are comparing the second order properties (note that IDV is derived from variance) of the process of counts, we have to generate the number of packets arriving during a time slot. A composition method is used to generate the hyperexponential inter-arrival times and thus the counts in fixed interval of time are calculated. By definition, we know that a random variable X follows a hyperexponential distribution $H(n; \lambda_1, \lambda_2 \ldots\ldots \lambda_n; w_1, w_2, \ldots\ldots)$ if the p.d.f,

$f(x) = \sum_{j=1}^{n} w_j \lambda_j e^{-\lambda_j x}$, $x > 0$.

for $0 \leq w_j \leq 1$, $\sum_j w_j = 1$, $\lambda_j > 0$, $j = 1, 2, \ldots\ldots, n$.

We already know the values of the w_j and the λ_j from the optimization technique. The algorithm to generate a hyperexponential random variable is as follows:

1. Generate a sample of uniform random variable 'U_1', between 0 and 1.

2. Depending on the value of 'U_1', select the value of λ. Specifically, $p_k = \sum_{i=1}^{k} w_i$ for $k > 0$. $p_0 = 0$. If $p_k < U_1 \leq (p_k + p_{k+1})$, $n > k \geq 0$, then the arrival rate associated with the weight is λ_{k+1} for $n > k \geq 0$.

3. Generate X as an exponential random variable, $X = -Log(1 - U_2)/\lambda$, where U_2 is uniform random variable between 0 and 1.

A trace with length equal to real network trace length is generated. The correctness is checked by the mean of the generated data to the mean of the original trace data. This synthetic data is used in the same manner as the trace data.

5.2 Numerical Results

This section gives an exposition to queuing with self-similar input. We consider a continuous time queueing model with infinite buffer and FIFO server. Let time be divided into fixed length sampling intervals. The real traffic represents number of cell counts in a sampling interval and we assume them to arrive equally spaced in the time interval. This information is utilized in calculating the inter-arrival time and hence the queuing delay of the traffic. The simulation technique for the synthetic data is described in previous section. A queuing system (G/M/1) is subject to self-similar arrival traffic to observe the delay in the queue. The results are compared among the real traffic trace (simulation), synthetic trace (simulation) and the G/M/1 analysis (theoretical).

We have shown in chapter 4 that the H_3 model is able to match the IDV reasonably well. Now we have to determine whether H_3 is an appropriate substitute for the original trace with respect to performance measures. An infinite

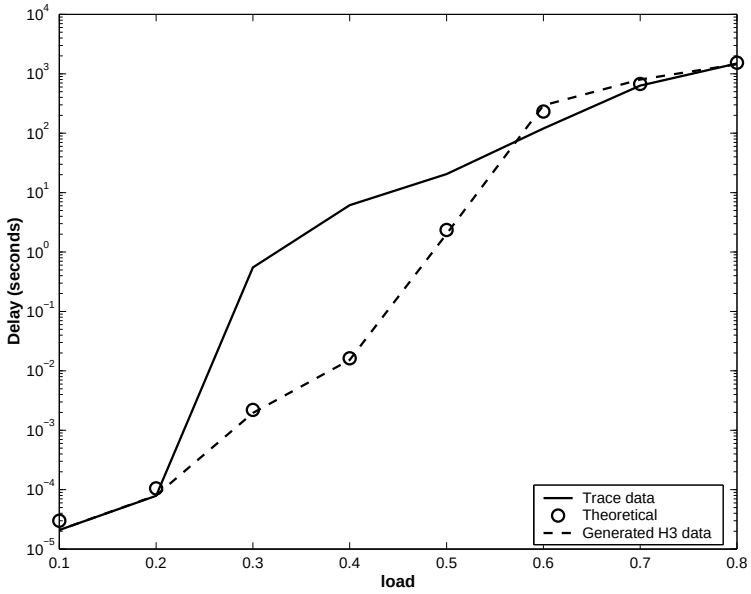


Figure 5.1: Delay Characteristics comparing Real network trace, Synthetic trace and Theoretical analysis

buffer, single server system with exponential service times and mean arrival rate is considered. Though the traffic trace used for comparison is an ATM trace, we assume an exponential service to match the analysis assumptions. We feed the server with an arrival stream modeled by a point process whose inter-arrival distribution is given by third order hyperexponential distribution. The queuing delays of the traffic trace and generated H_3 are calculated using the FIFO server and infinite buffer and compared to the theoretical results of G/M/1. The delay for different methods used in calculating H_3 parameters is presented.

5.2.1 H_3 with Parameters from Optimization

The simulated trace considered here is the data generated using parameters obtained from the optimization technique. The mean delay is calculated for various loads. The generated H_3 and the G/M/1 theoretical coincide well (Figure 4.8) at a few time scales. The real trace coincides well at very low and very high

loads which can be explained using the peak rate curve. The delays for the real network trace are high at intermediate loads compared to generated H_3 loads. In brief, G/M/1 yields optimistic results.

The delay increases steeply when the load is increased from 0.2 for the real traffic trace. The service rate at 0.2 is approximately 11 Mbps and this exceeds the peak rates even at small time scales for both real trace and H_3 trace. Hence, the mean delay is low as expected. Next, consider an intermediate load of 0.4. The service rate is 5.5 Mbps, which is lower than the peak rate of the real trace but higher than the peak rate of the H_3 at time scales less than 1 second. This is reflected in the greater difference between the two traces. For a load of 0.6, the service rate is approximately 3.7 Mbps, which is lower than the peak rates of both the traces even for relatively large time scales. Consequently, there is a queue build-up leading to high delays (hundreds of seconds) for both.

Obtaining parameters for the H_3 by matching IDV is advantageous as it can correctly estimate the queuing behavior of the real trace. IDV is an important factor for generating a self-similar traffic with predefined peak rate characteristics.

5.2.2 H_3 with Parameters from Heuristics

Though the IDV is matched pretty well using heuristics, we can conclude from queuing analysis that the generated trace cannot be a substitute for the real trace.

From the peak rate curve (Figure 4.6), we can estimate the queuing behavior of the generated H_3 data. A single value of the mean delay at 80% load is calculated for the generated H_3 trace and compared with the mean delay of the real trace. The delay characteristics are as follows,

Condition	Mean (s)	Std Dev (s)
H_3 Theoretical	0.0114	0.0114
H_3 Simulation	0.0114	0.0114
Real Trace Sim	1491	1597

Here there is good agreement between analysis and simulation results for the H_3 model, but the mean and standard deviation of the real traces are several orders of magnitude larger than the corresponding values of H_3 model. There are infinite solutions to match the IDV but a correct solution is that which correctly matches the performance characteristics. Given just the mean of the traffic trace and the cell counts per fixed interval, there is no accurate procedure to determine the solution that matches the performance characteristics because the generation of synthetic traces is a random process. Since we cannot guarantee that matching real trace IDV would match the performance results, performance tests should be done on all those synthetic traces that match the real trace IDV.

5.3 Relevant Time Scales

If self-similarity is not taken into consideration, it can lead to inaccurate conclusions in performance metrics. So, the Hurst parameter has been used in evaluating the performance of networks. It was believed that higher Hurst parameter values result in worse queuing performance. Recent approaches [31] showed that queuing performances are related to a few important time scales. The authors argue that a higher Hurst parameter may be associated with smaller queues. A function is described which relates buffer size, capacity of link and the standard deviation. For ease of use, we call the function as 'relevant time scale' (RTS) function. RTS is given by

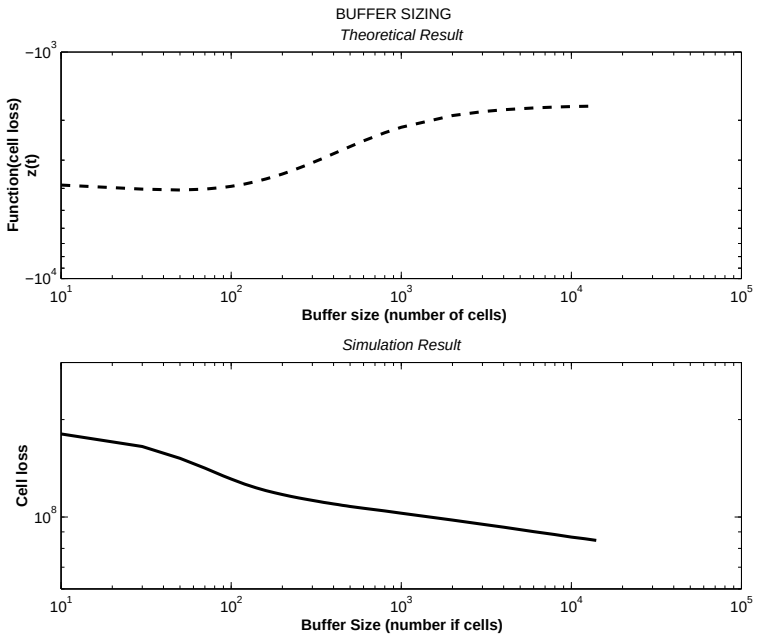


Figure 5.2: RTS function, z(t) and Cell Loss

$$z(t) = \frac{B + ct}{S(t)} \tag{5.7}$$

where B is the buffer size, m $(=\lambda_{mean} * 53 * 8)$ is the mean traffic rate, $S(t)$ is standard deviation of the data and c $(=C - m)$ is the excess capacity. C is the percentage of actual capacity of link (capacity shared by a VC) because we are considering the case of a VC here. Also, $\rho = m/C$, so we get:

$$c = \frac{m(1 - \rho)}{\rho} \tag{5.8}$$

$z(t)$ is minimized with respect to time scale to find the relevant time scale. As RTS function decreases with decrease in size of buffer, the cell loss increases with decreasing buffer size.

The z(t) function predicts the cell loss approximately. Observe that the sub plots in Figure 5.2 almost look like mirror images. z(t) gives the region where the

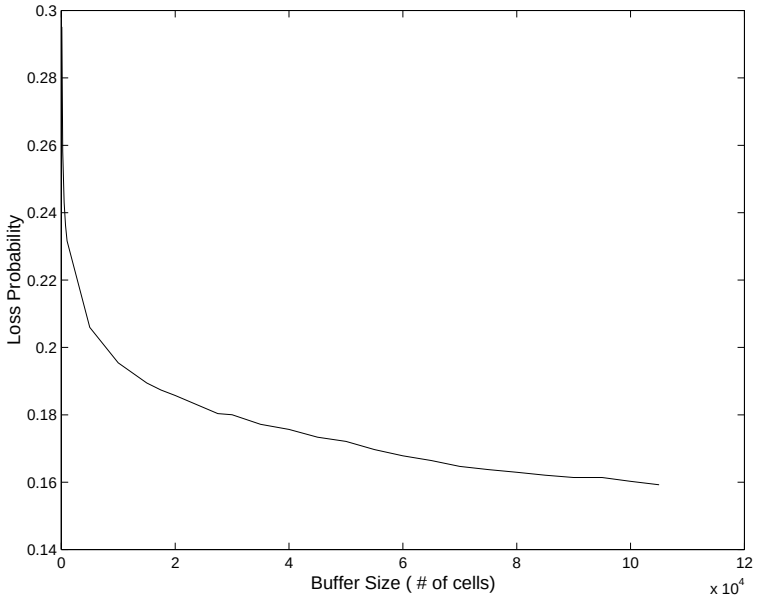


Figure 5.3: Loss Probability for Self-Similar Data

cell loss will rapidly decrease or will remain constant as the buffer size increases. If the cell loss is decreasing with a lesser gradient as the buffer size increases (1500-10000 cells), it would be a inefficient to increase buffer sizes in the switches to support bursty traffic.

5.3.1 Cell Loss

We have used ATM traffic traces and performed queuing analysis in order to investigate the effect of LRD on cell loss. As expected, the cell loss probability decreases as the buffer size increases. The loss probability decreases steeply though buffer has the capacity to accommodate thousands of cells. Figure 5.3 is the loss probability of a VC with mean arrival rate of 5183 cells/second and load of 70%. Probability is high (0.16, total number of cell counts being around 50 millions in a 24 hour long trace) with unrealistic buffer size of 100000 cells. For Poisson process with same arrival rate and load factor, it was observed that the

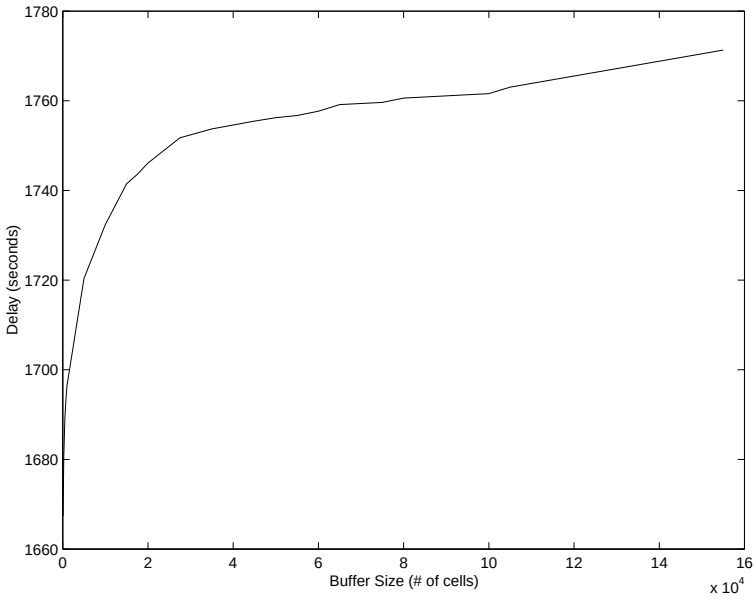


Figure 5.4: Delay of Self-Similar data with 40% Load

loss probability was zero for large buffers. The effect of LRD is that the buffers needed at the switches must be bigger than the predicted by traditional queuing analysis and simulations. The degree to which self-similarity effects performance is determined by the load on the link. We considered a moderate load situation where delays are still high for large buffer sizes (Figure 5.4). This situation is considered to demonstrate the realistic condition (40% load) where there can be greater loss.

Chapter 6

Conclusion and Future Work

6.1 Conclusions

This chapter concludes the work done for the thesis and summarizes the work for future extensions. Our initial step was to visually inspect the self-similarity and burstiness at various time scales. We then deviated to do some traffic engineering by analyzing the peak rates of the real network trace data. A useful relationship between peak rate at higher and lower aggregation levels was derived to calculate peak rate at smaller aggregation level. We concluded that it would be erroneous to consider the peak rate at higher aggregation levels for network management.

Continuing the work on self-similarity and LRD, the concept of IDV was extended to ATM traces to observe the variability in traffic. The possibility to model self-similar traffic by means of hyperexponential distribution was assessed. There was an attempt to match the IDV using heuristic and optimization methods. We showed that a hyperexponential model of order 3 can be used to match IDV for a few time scales and estimate delay characteristics approximately. We pointed that choice of parameters can have a drastic effect on queuing and peak

rate though IDV matches well at most of the time scales.

Assuming an exponential inter-arrival distribution defeats the purpose of performance evaluation in networks. Even assuming a hyperexponential distribution that matches the mean and the constraints is misleading sometimes. G/M/1 analysis is presented to compare the queuing characteristics of the model. The delay at higher loads of the real trace is successfully captured by the hyperexponential model. Load-delay characteristics capture the statistical properties of the real traffic at very high loads and very low loads. This has been correlated to the peak rate behavior of the traffic. Finally, we address that our model has complexity in calculating IDV and hyperexponential parameters for orders higher than 3.

Contribution of Thesis

This thesis considered the aspects of packet networks with modeling of packet arrivals and peak rate variability in the traffic. We analyzed the peak rate characteristics and self-similarity in the ATM traffic. We now present our accomplishments in the thesis.

1. ATM traffic traces were analyzed to show the self-similarity in them. An interesting subject that we applied to the ATM traffic trace was the concept of IDV, a relatively new measure of self-similarity. We introduced a new procedure to calculate IDV from the log-log variance time plot.

2. We also introduced a new measure for traffic analysis, peak rate variability (PRV). We have shown that peak rate at 5 ms can be estimated, given the peak rate at one-hour. Peak rate studies on the link as well as on VCs on the link were done. The traffic flow at various aggregation was discussed in

both VC and OC-links with the help of PRV curve.

3. We have proposed that a higher order hyperexponential distribution can be used for modeling the self-similar data.

4. We showed that hyperexponential of small order (H_3) is able to model the self-similarity behavior over several time scales. An equation to find the IDV for H_3 was derived using the analysis in [4].

5. We approached with a heuristic method to match the real traffic trace IDV curve but failed to match the peak rate and queuing properties. We then used optimization techniques to match the IDV. The synthetic traces generated using the optimization method matched the peak rate properties at a few time scales and matched the queuing properties quite well.

6. We evaluated our techniques to match IDV by using the queuing results for G/M/1 queue.

7. As the final work, we have shown that self-similar data has higher cell loss when compared to Poisson traffic, concluding that the analysis using traditional models can drastically affect the network performance.

6.2 Future Work

Further work is needed to analyze the shape of PRV curves. There are few time scales where the peak rate is constant and this is typical for all the ATM curves. Also, the analysis on the peak rate relationship should be extended to other types of traffic in the network. The proposed approach of H_3 looks promising, but a more rigorous algorithm to find the parameters of the hyperexponential of any

order is still needed. A new approach to solve IDV from the counting process has to be found without involving complex Laplace/Fourier calculations. Due to inherent bursty nature of the traffic, there is a significant impact of packet loss and network delay. Queuing incorporating the IDV should be focussed. To smooth the traffic, an optimal allocation of buffers in the network is needed and this resource allocation problem has to be solved using IDV. IDV expression is not in closed form for higher order of hyperexponential and computationally inefficient. Further research should also account for numerical tractability of the approach. Nevertheless, the implementation of efficient numerical procedures for estimating the parameters of a distribution remains an open problem that limits the use of various class of distributions in applications. A better match of IDV is possible if the queuing characteristics are also matched. Similar to the matching of IDV curve, the queuing curve can be matched using the optimization techniques. We hope this analysis will help in generating a trace data that matches in queuing as well as IDV characteristics.

Bibliography

[1] W. Willinger, M. Taqqu, R. Sherman, and D. Wilson, "Self-Similarity Through High-Variability: Statistical Analysis of Ethernet LAN Traffic at the Source Level," IEEE/ACM Transactions on Networking, 5(1), pp. 71-86, January 1997.

[2] J. Beran. Statistics for Long-Memory Processes. Chapman & Hall, 1994.

[3] J. Beran, R. Sherman, M. Taqqu, and W. Willinger, "Long-Range Dependence in Variable-Bit-Rate Video Traffic," IEEE Trans. on Communications, Vol.43, Nos.2-4, pp.1566-79, February/March/April 1995.

[4] G. Y. Lazarou, "On the variability of Internet traffic", Ph. D. dissertation, University of Kansas, 2000.

[5] V. Paxson and S. Floyd. "Wide-area traffic: the failure of Poisson modeling,". In Proc. ACM SIGCOMM '94, 1994.

[6] M.W. Garrett and W. Willinger, "Analysis, Modeling and Generation of Self-similar VBR Video Traffic,", Proc. Sigcomm'94, Sept. 1994.

[7] K. Park, G. Kim, and M. Crovella, "On the Effect of Traffic Self-Similarity on Network Performance," In Proc. SPIE International Conference on Performance and Control of Network Systems, November 1997.

[8] D.R.Cox. Renewal Theory. Methuen,London,1962.

[9] D.R.Cox and Valerie Isham, Point Process, London and New York, Chapman and Hall, 1980

[10] C. Goldie and C. Klüppelberg, "Sub-exponential Distributions, in A Practical Guide to Heavy Tails: Statistical Techniques for Analyzing Heavy Tailed Distributions," pp.435-459, R. Adler, R. Feldman, M. Taqqu (Eds), Birkhäuser, 1998.

[11] M. Greiner, M. Jobmann and L. Lipsky, "The Importance of Power-Tail Distributions for Telecommunication Traffic Models, Operations Research," Vol.47, No.2, pp.313-326, March 1999.

[12] R. Nelson, Probability, Stochastic Processes, and Queueing Theory, Springer-Verlag, 1995.

[13] S.C.Pothuri, L.Vijayan and D.W.Petr, "Analysis of Measured Traffic on OC Links," ITTC,Technical Report, ITTC-FY2001-TR-18838-01.

[14] W. Willinger, M.S. Taqqu, and A. Erramilli, "A Bibliographical Guide to Self-Similar Traffic and Performance Modeling for Modern High-Speed Networks Stochastic Networks: Theory and Applications," In Royal Statistical Society Lecture Notes Series, volume 4. Oxford University Press, 1996.

[15] Park, K., Willinger, W. (eds.) (2000). Self-Similar Network Traffic and Performance Evaluation. Wiley, New York.

[16] Will Leland, Murad Taqqu, Walter Willinger, and Daniel Wilson, "On the Self-Similar Nature of Ethernet Traffic (Extended Version)," IEEE/ACM Transactions on Networking, Vol. 2, No. 1, pp. 1-15, February 1994.

[17] A. Feldmann and W. Whitt, “Fitting Mixture of Exponentials to Long-Tail Distributions to Analyze Network Performance Models,” Performance Evaluation, Vol.31, pp.245-279, 1998.

[18] S. Robert and J.-Y. Le Boudec “New Models for Pseudo Self-Similar Traffic,” Performance Evaluation, Vol.30, pp.57-68, 1997.

[19] P. Morin, J. Neilson, "The Impact of Self-Similarity on Network Performance Analysis," Tech. Report No.95.495, Computer Science, Carleton University, December 1995.

[20] Grossglauser, M., Bolot, J.-C. (1999). “On the relevance of long-range dependence in network traffic,”. IEEE/ACM Trans. Netw. 7, 629–640.

[21] N. Likhanov, B. Tsybakov, and N. Georganas. “Analysis of an ATM buffer with self-similar fractal input traffic,”. In Proceedings of IEEE INFOCOM ’95, pages 985–992, Boston, MA, Apr. 1995.

[22] Y. H. Kim, S. -q. Li, "Time scale of Interest in Traffic Measurement for Link Bandwidth Allocation Design," IEEE Proc. INFOCOM’96, San Francisco, USA, March 1996.

[23] S.Molnar, A.Vidacs, A. Nilsson, Bottlenecks on the Way Towards Fractal Characterization of Network Traffic: Estimation and Interpretation of the Hurst Parameter , International Conference of the Performance and Management of Complex Communication Networks (PMCCN’97)

[24] Allan T.Anderson, “Modeling of Packet Traffic with Matrix Analytical Methods,” Ph.D. Dissertation, IMM-PHD-1995-18, Technical University of Denmark, Lyngby 1995.

[25] B. Ryu and S. Lowen. “Point process models for self-similar network traffic, with applications,”. Stochastic Models, 1997.

[26] R. Fourer, D. M. Gay, and B. W. Kernighan, AMPL: A Modeling Language for Mathematical Programming, Boyd and Fraser, 1993.

[27] R.Nelson,Probability, Stochastic Processes, and Queueing Theory, Springer-Verlag,1995

[28] M. Greiner, M. Jobmann, and L. Lipsky, "The Importance of Power-Tail Distributions for Modeling Queueing Systems," Operations Research, vol. 47, no. 2, March-April 1999.

[29] M. Greiner, M. Jobmann and C. Kluppelberg, "Telecommunication Traffic, Queueing Models, and Subexponential Distributions," Queueing Systems, 1999.

[30] Alma Riska, Evgenia Smirni, Gianfranco Ciardo, "Analytic Modeling of Load Balancing Policies with Heavy-tailed Distributions," in the Proceedings of the Second International Workshop on Software and Performance, pages: 147-157, Otawa, Canada, September 2000.

[31] Neidhardt, A., and Wang, J. “The Concept of Relevant Time Scales and Its Application to Queueing Analysis of Self-Similar Traffic (or Is Hurst Naughty or Nice?),”. In Proceedings ACM SIGMETRICS’98 (Madison, Wisconsin, USA, Jun. 1998), pp. 222–232.

[32] A. M. Law and W. D. Kelton. Simulation Modeling and Analysis. McGraw-Hill, New York, 2 nd ed. edition,1991.

[33] J. Abate,G. L. Choudhury and W. Whitt,"Waiting-time tail probabilities in queues with long-tail service-time distributions," Queueing Systems 16 (1994) 311- 338.

[34] Philippe Nain (1999)." Impact of unsmooth traffic on network performance," Stastistical Inference for Stochastic Process.

[35] M. E. Crovella and A. Bestavros, " Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes," In Proc. ACM/SIGMETRICS'96, May 1996.

[36] W. Willinger, V. Paxson, and M.S. Taqqu, "Self-similarity and Heavy Tails: Structural Modeling of Network Traffic, In: A Practical Guide to Heavy Tails," 27-53, Chapman & Hall 1998.

[37] A. Erramilli, O. Narayan and W. Wilinger, "Experimental Queuing Analysis with LRD Packet Traffic," IEEE/ACM Trans on Networking, vol 4, no 2, Apr 1996.

[38] ATM Forum. `http://www.atmforum.com`

[39] J. Abate and W. Whitt. "Computing Laplace transforms for numerical inversion via continued fractions," INFORMS Journal on Computing, 7(1):36–43, 1995.

Appendix

IDV for H_3

IDV for 3rd order hyper exponential distribution.

Probability density function of inter arrival time:

$f(x) = kae^{-ax} + mb^{-bx} + nce^{-cx}$

Mean arrival rate $= \lambda = 1\frac{1}{E(x)}$

Mean inter-arrival time $= E(x) = \frac{1}{a} + \frac{m}{b} + \frac{n}{c}$

Therefore, $\lambda = \frac{abc}{kbc+mca+nab}$

$Var[N(t)] = 2\lambda \int_0^t \phi(t)dt + \lambda t - (\lambda t)^2$

To find $\phi(t)$

$g(x) = \frac{f(x)}{1-f(x)} \& k + m + n = 1$

$f^*(s) = \frac{k}{s+a} + \frac{m}{s+b} + \frac{n}{s+c}$

$\phi(t) = L^{-1}[\frac{g^*(s)}{s}]$

$g*(s) = \frac{\frac{ak}{s+a} + \frac{bm}{s+b} + \frac{cn}{s+c}}{1-[\frac{ak}{s+a} + \frac{bm}{s+b} + \frac{cn}{s+c}]}$

$= \frac{ak(s+b)(s+c)+bm(s+a)(s+c)+cn(s+a)(s+b)}{(s+a)(s+b)(s+c)-[ak(s+b)(s+c)+bm(s+a)(s+c)+cn(s+a)(s+b)]}$

$= \frac{s^2(ak+bm+cn)+s[ak(b+c)+bm(a+c)+cn(a+b)]+abc(k+m+n)}{s^3+s^2(b+c+a)+s(ab+bc+ca)+abc-[(ak+bm+cn)s^2+s[ak(b+c)+bm(a+c)+cn(a+b)]+abc(k+m+n)]}$

$= \frac{s^2(ak+bm+cn)+s[ab(k+m)+ac(k+n)+bc(m+n)]+abc}{s^3+s^2(a+b+c)+s[ab+bc+ca]+abc-s^2(ak+bm+cn)-s[ab(k+m)+ac(k+n)+bc(m+n)]-abc}$

$= \frac{s^2(ak+bm+cn)+s[ab(k+m)+ac(k+n)+bc(m+n)]=abc}{s^3+s^2[a+b+c-ak-bm-cn]+s[ab(1-k-m)+bc(1-k-n)+ac(1-m-n)]}$

$= \frac{s^2(ak+bm+cn)+s[ab(k+m)+ac(k+n)+bc(m+n)]+abc}{s^3+s^2(a+b+c-ak-bm-cn)+s(abn+acm+bck)}$

Let,

$x = ak + bm + cn$

$z = ab(k + m) + ac(k + n) + bc(m + n)$

$F = a + b + c - (ak + bm + cn) = a + b + c - x$

$E = abn + acm + bck$

Therefore equation reduces to

$\frac{s^2x+sz+abc}{s^3+s^2F+sE}$

$= \frac{sx}{s^2+sF+E} + \frac{z}{s^2+SF+E} + \frac{abc}{s(s^2+sF+E)}(1)$

Let roots of equation $s^2 + sF + E$ be r_1 and r_2

Therefore

$r_1 = (-F + \sqrt{F^2 - 4E})/2$

$r_2 = (-F - \sqrt{F^2 - 4E})/2$

Also,

$r_1 - r_2 = j$

Reducing 1 and 2 & partial fractions of 2 gives

$\frac{sx}{(s-r_1)(s-r_2)} + \frac{z}{(s-r_1)(s-r_2)} + \frac{abc}{s(s-r_1)(s-r_2)}$ (2)

$\frac{x}{r_1-r_2}[\frac{r_1}{s-r_1} - \frac{r_2}{s-r_2}] + \frac{z}{r_1-r_2}[\frac{1}{s-r_1} - \frac{1}{s-r_2}] + abc[\frac{1}{r_1 r_2 - s} + \frac{1}{r_1(r_1-r_2)(s-r_1)} - \frac{1}{r_2(r_1-r_2)(s-r_2)}]$

$= \frac{1}{(r_1-r_2)}[\frac{xr_1}{s-r_2} - \frac{xr_2}{s-r_2} + \frac{z}{s-r_1} - \frac{z}{s-r_2} + \frac{abc(r_1-r_2)}{r_1 r_2 s} + \frac{abc}{r_1(s-r_1)} - \frac{abc}{r_2(s-r_2)}]$ (3)

Taking Inverse Laplace of 3(above equation)

$= \frac{1}{(r_1-r_2)}[(xr_1 + z + \frac{abc}{r_1})e^{r_1 t} - (xr_2 + z + \frac{abc}{r_2})e^{r_2 t} + \frac{abc(r_1-r_2)}{r_1 r_2}]$

Let $Q = xr_1 + z + \frac{abc}{r_1}$

$R = xr_2 + z + \frac{abc}{r_2}$

$M = \frac{abc}{r_1 r_2} \approx \lambda$

$M = \lambda \ (for \ t < 10^6)$

Therefore,

$g(t) = \frac{1}{(r_1-r_2)}[Qe^{r_1 t} - Re^{r_2 t} + M(r_1 - r_2)]$

From A

$g(t) = \frac{Q}{J}e^{r_1 t} - \frac{R}{J}e^{r_2 t} + M$

$\phi(t) = \int_0^t g(u)du$

$= \frac{Q}{Jr_1}e^{r_1 t} - \frac{Re^{r_2 t}}{Jr_2} + Mt - [\frac{Q}{Jr_1} - \frac{Q}{Jr_2}]$

Let

$P = \int_0^t \phi(u)du$

$= \frac{Q}{Jr_1^2}e^{r_2 t} - \frac{R}{Jr_2^2}e^{r_2 t} + \frac{Mt^2}{2} - [\frac{Q}{Jr_1} - \frac{R}{Jr_2}].t - [\frac{Q}{Jr_1^2} - \frac{Q}{Jr_2^2}]$

From variance expression

$V = Var[N(t)] = 2\lambda \int_0^t \phi(u)du + \lambda t - (\lambda t)^2$

$= 2\lambda p + \lambda t - (\lambda t)^2$

$D = \frac{dv}{dt} = 2\lambda[\frac{Q}{Jr_1}e^{r_1 t} - \frac{R}{Jr_2}e^{r_2 t} + Mt - [\frac{Q}{Jr_1} - \frac{R}{Jr_2}]] + \lambda - 2\lambda^2 t$

$since\ M = \lambda$

$D = \lambda + 2\lambda[\frac{Q}{Jr_1}e^{r_1 t} - \frac{R}{Jr_2}e^{r_2 t} - \frac{R}{Jr_2}e^{r_2 t} - (\frac{Q}{Jr_1} - \frac{Q}{Jr_2})]$

Therefore

$Idv = H_v(t) = 0.5t * \frac{D}{V}$

PRV Plots of OC-links

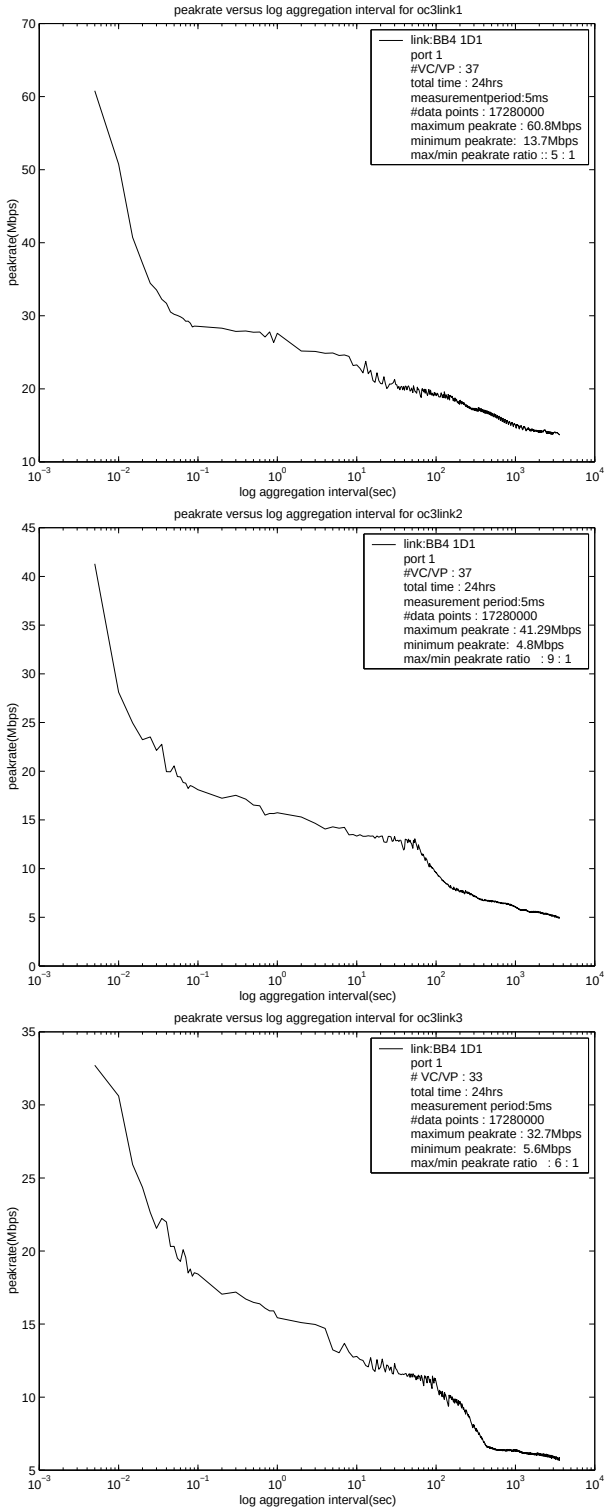
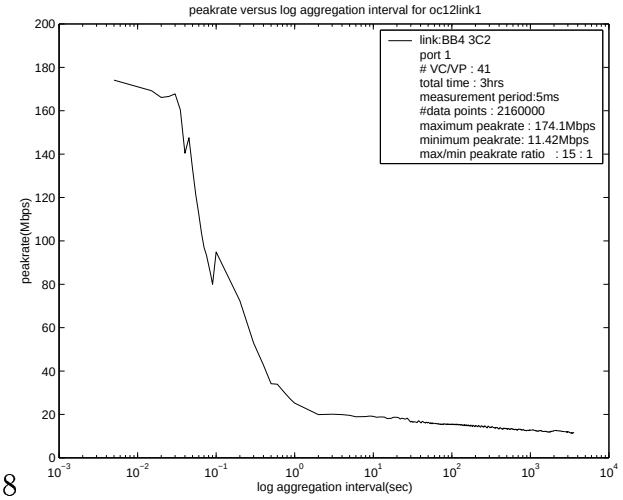
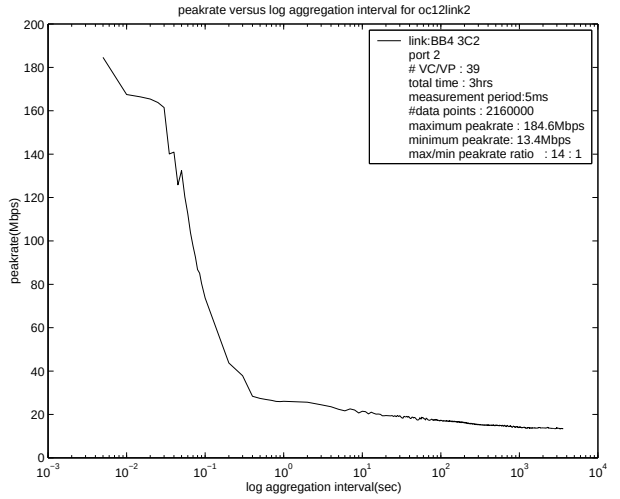


Figure 6.1: PRV plots of OC-3 links





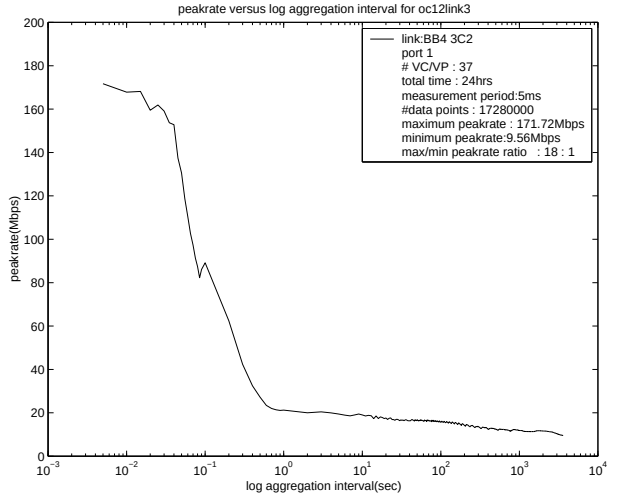


Figure 6.2: PRV plots of OC-12 links

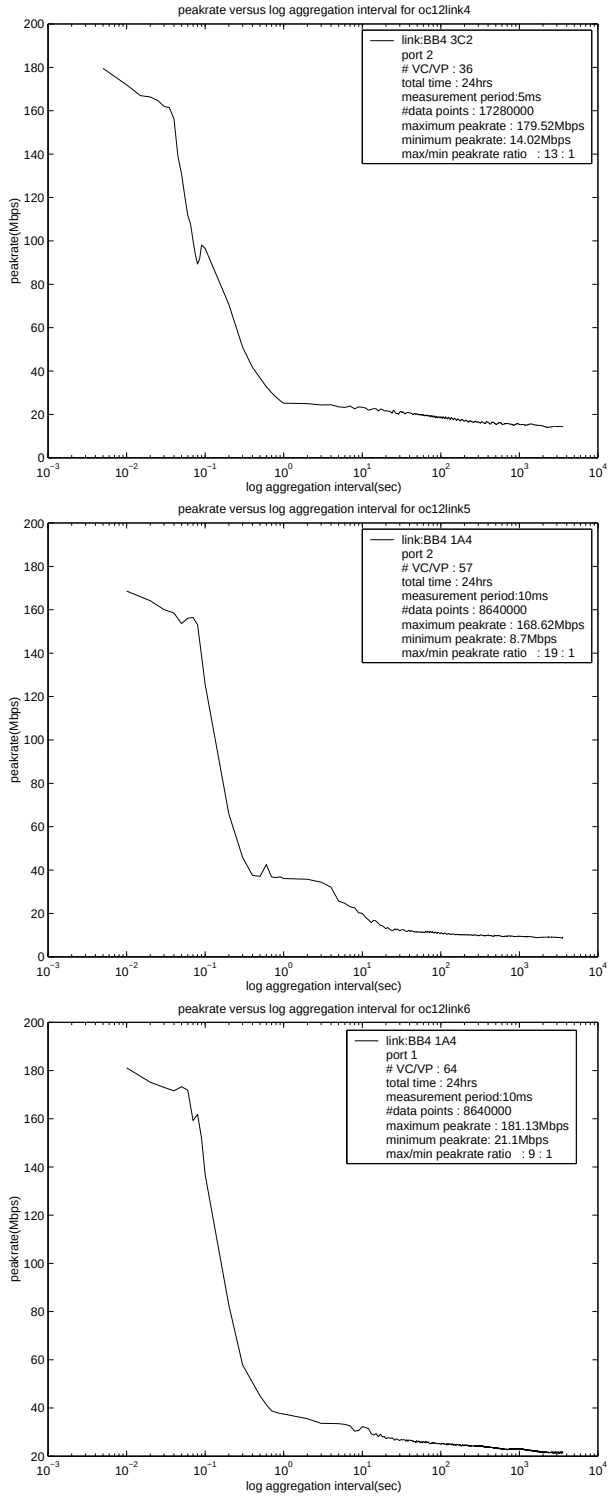


Figure 6.3: PRV plots of OC-12 link