

ANALYZING INTERNET PACKET TRACES USING LINDLEY'S RECURSION

Martin J. Fischer

Mitretek Systems, Inc.
3150 Fairview Park Drive South
Falls Church, VA 22042-4519, U.S.A.

Denise M. Bevilacqua Masi

Mitretek Systems, Inc.
3150 Fairview Park Drive South
Falls Church, VA 22042-4519, U.S.A.

ABSTRACT

Internet trace packet data for a given network link contains information on each packet's arrival time and size. An important problem is to model the congestion packets experienced over the collection period. Recent research has utilized a relationship from Queueing Theory known as Lindley's Recursion to model packet congestion. This relationship has existed for 50 years and has been quite beneficial in analyzing these traces. We report on our use of Lindley's Recursion to analyze publicly-available link data from the Abilene Network, an Internet2 backbone network. We extend the use of Lindley's Recursion and include a discussion of the computational problems, numerical evaluation of trace packet performance and potential modeling issues, and a statistical investigation of the independence of packet interarrival times. In addition, we show how Lindley's Recursion can be used to extend the baseline analysis to interject Voice over Internet Protocol (VoIP) packets into the trace.

1 INTRODUCTION

Internet trace packet data for a given link in the network contains information on each packet's arrival time and size. A very important problem is to model the congestion the trace packets experienced over the collection period. In this paper we discuss how to utilize a very famous 50-year-old equation called Lindley's Recursion from Queueing Theory to model the congestion the packets are experiencing. This discussion includes the computational problems we encountered, numerical evaluation of trace packet congestion performance and potential modeling issues, a statistical investigation of independence, and how Lindley's Recursion can be used to analyze performance if VoIP packets are interjected into the trace. This use of Lindley's Recursion to analyze a packet trace is a numerical simulation to study the system performance; in this case the trace data is used to compute arrival and service times, rather than generating random variables.

This paper has grown out of our modeling and simulation support to the National Communications System (NCS). The mission of NCS includes assisting the President, National Security Council, Office of Science and Technology Policy, and the Office of Management and Budget in the coordination of the planning for and provision of *national security and emergency preparedness communications* for the federal government under all circumstances, including *crisis or emergency, attack, recovery, and reconstitution*. In support of its mission, NCS runs several emergency telecommunications priority service programs for federal government users, including the Government Emergency Telecommunications Service (GETS) and the Wireless Priority Service (WPS). However, industry is moving from circuit switched to Internet Protocol (IP) technology for all telecommunications applications including voice, and NCS is initiating a new program called the Next Generation Priority Service (NGPS). This NCS program will investigate the need to evolve toward IP capability to ensure continuity of priority traffic during emergencies. As part of that investigation, we are conducting various modeling and simulation activities to support the NGPS. The results presented in this paper are just one step in gaining a better understanding of the modeling of such systems.

The modeling of IP networks depends on the nature of the packet interarrival and packet size distributions. These statistics have been extensively studied by many authors. For a sample of such studies, see (Paxson and Floyd 1995), (Resnick 1997), (Adler et al. 1998), (Cao et al. 2002), (Nogueira et al. 2004), (Heyman 2005), (Karagiannis et al. 2004), and (Fischer and Masi 2005). These studies started in the early 1990s and continue until the present. Self-similarity, long-range dependence, heavy-tailed distributions, convergence to Poisson processes, and Markov Modulated Poisson Processes are just some of the statistical properties used to characterize these traffic statistics.

In this paper we present a method to analyze Internet traces using a famous equation called Lindley's Recursion from Queueing Theory (Gross and Harris 1998). This simple equation does not depend on statistical assumptions

about the packet trace—it just takes the packet trace and generates required congestion results like packet latency, jitter, and loss. In fact, it can be used to generate the complete probability distribution of delay and buffer sizes.

One of the first uses of Lindley's Equation to analyze Internet packet traces was the 2004 report by Park et al. and subsequent paper in *Computer Networks*. That paper discusses a Trace-Driven Queueing Analysis using Lindley's Recursion as a method of simulating traces. They provide an analysis of long-range dependence in a continually evolving Internet traffic mix by employing a number of recently developed statistical methods. Heegaard et al. (2005) used a Lindley-type recursion and importance sampling to analyze packet traces of MPEG encoded video streams. They introduced the use of importance sampling to minimize the impact of large number of events that have to be processed.

In this paper we extend these uses of Lindley's Recursion to a much larger class of analysis problems. In Section 2 we present and discuss Lindley's Recursion and a related congestion measure called Kingman's Upper Bound, which can be used to determine packet arrival time independence. The Internet traces we used in this analysis are discussed in Section 3. That section also included our computational experiences in processing the data. In Section 4, we present some of the potential analyses by applying Lindley's Recursion to an Internet trace and the resulting congestion analysis. Those analyses include a numerical evaluation of trace packet performance (packet delay, jitter and loss) and a statistical investigation of independence. We also show Lindley's Recursion can be used to analyze a modified system if VoIP packets are interjected into the trace. We close the paper with some concluding remarks on future extensions of the use of Lindley's Recursion.

2 LINDLEY'S RECURSION FROM QUEUEING THEORY

Consider a G/G/1 queueing system—in such a queue, the customer interarrival distribution is general, the service distribution is general, and there is a single server. (For a complete description of this queueing system, see Gross and Harris, 1998.) For the n^{th} arrival, define $W_q(n)$ to be the waiting time in the buffer, $A(n)$ to be the arrival time, and $S(n)$ to be the service time. For the case of an infinite buffer and a First Come First Served (FCFS) system, Lindley's Recursion is:

$$W_q(n) = (W_q(n-1)(A(n) - A(n-1)) + S(n-1))^+,$$

where $x^+ = x$ if $x \geq 0$, and $x^+ = 0$ if $x < 0$.

Lindley's Recursion states that the waiting time in the buffer of the n^{th} arrival equals the waiting time in the

buffer of the $(n-1)^{\text{st}}$ arrival minus the time between the n^{th} arrival and the $(n-1)^{\text{st}}$ arrival plus the service time of the $(n-1)^{\text{st}}$ arrival. If that quantity is less than or equal to zero, then the waiting time in the buffer of the n^{th} customer is zero. The only assumption required for Lindley's Recursion to be valid is FCFS service discipline and an infinite buffer. We do not need any independence assumptions.

For the case of a finite buffer, Lindley's Recursion can be modified to give the loss probability of the n^{th} arrival as well as the buffer waiting time. At each arrival, all service completion events since the previous arrival time must be tracked to update the number of customers in the system, e.g., using event-oriented bookkeeping. This process requires arrays saving the event times and event type (service, arrival, blocked arrival). Since Internet trace data has each packet size, keeping track of the departures between arrivals is straightforward because of the FCFS discipline. When the number of customers in the system at the current arrival time is computed, then it is determined whether the current customer will be serviced or blocked. Thus, it is straightforward to extend Lindley's Recursion to the finite buffer case for FCFS systems.

Once $W_q(n)$ is computed for each arrival, moments of the buffer delay are easily found as well as the complete probability distribution of buffer delay for an arbitrary arrival. Since the system wait of the n^{th} arrival equals the buffer wait plus the service time, we also easily have that statistic. In addition, using a modification of the method described above for the finite buffer problem, one can determine the number of customers in the system at an arrival point. So, the generation of $W_q(n)$ and the availability of the data allow one to compute all congestion measures of interest; that is the expected buffer delay, jitter and loss. In addition, if interested one could generate the complete probability distribution of buffer delay.

There is also a famous upper bound—Kingman's Upper Bound—on the expected customer delay in the buffer for a FCFS infinite buffer G/G/1 queue. If $E\{A\}$, $\text{Var}\{A\}$ are the mean and variance of the interarrival times and $E\{S\}$ and $\text{Var}\{S\}$ are the similar statistics for the service time, then Kingman's Upper Bound for the expected delay in the buffer, $E\{W_q\}_{KUB}$, is given by

$$E\{W_q\}_{KUB} = \frac{\text{Var}\{A\} + \text{Var}\{S\}}{2(E\{A\} - E\{S\})}.$$

For this bound to hold, we must have the load to be less than one, that is $E\{A\} > E\{S\}$, the customer interarrival times are independent and service times are independent and the service discipline to be FCFS. For a development of $E\{W_q\}_{KUB}$, see (Gross and Harris 1998); it becomes exact as the load ($= E\{S\}/E\{A\}$) approaches one. Kingman's Upper Bound is a steady state result. Thus, one would expect if the data from the Internet obey the independence assumption, then Kingman's Upper Bound

would be a very important measure. In addition, if the upper bound was violated, then one would expect that is due to the independence assumptions being violated.

The Transmission Control Protocol (TCP) congestion control that would be implemented in a real network is not being modeled here. Model results from Lindley's Recursion and Kingman's Upper Bound are rather being used and compared to check the appropriateness of the model assumptions (i.e., the independence required by Kingman's).

3 INTERNET TRACES

In our analysis, we used the publicly-available data from Abilene Network, an Internet2 high-performance backbone network that enables the development of advanced Internet applications and the deployment of leading-edge network services to Internet2 universities and research labs across the country. The Abilene Network supports the development of applications such as virtual laboratories, digital libraries, distance education and teleimmersion, and the advanced networking capabilities that are the focus of Internet2. The current network is a primarily OC-192 (10 Gbps) backbone; at the time the data was collected it was OC-48; see <http://abilene.internet2.edu/>.

Packet traces were collected by the National Laboratory for Applied Network Research (NLNAR) on August 14, 2002, at the Indianapolis router node (IPLS), a CISCO GSR-12015 with four OC-48 uplinks. We downloaded data from the Cleveland to Indianapolis (westbound) link; the opposite direction is also available, plus both directions of the IPLS—Kansas City link. (The data can be found at <http://pma.nlanr.net/Traces/long/ipls1.html>.) We have downloaded 1 of 2 available hours of data, starting at 9:00 a.m. The traces are in Dag PoS format. We use Libtrace (<http://research.wand.net.nz/software/libtrace.php>) to read the data.

The size of the file is approximately 6 GB of data. For each packet in the file, the full set of IPv4 packet fields is available, resulting in this very large file. In order to process the data using Lindley's Recursion, we had to filter the trace and create files that contained only the packet arrival time and size fields. We also had to check the data to be sure the arrival times were ordered from smallest to largest. We processed an hour's worth of data and saved it in 60 files; one for each minute in the hour. We then sorted each of the files using the UNIX sort command to be sure the arrival times were ordered. Each file had a little over 100 MB of data with around 4,500,000 records. It took around one minute on a 1.8 GHz PC to sort each file. We did not find any problems with the data. Then we used VBA to read and process the data using Lindley's Recursion. All processing was done on a 1.8 GHz PC, and depending on

the type of analyses being conducted, could take anywhere for a couple of minutes to a half an hour.

4 DATA AND CONGESTION ANALYSIS

In this section we present several analyses we have done with the data and Lindley's Recursion. Some of these analyses did not require the use of Lindley's Recursion. Figure 1 is a plot of the packets per second from 9:00 a.m. to 10:00 a.m. on the Cleveland to Indianapolis link. We see the arrivals are fairly constant for roughly 10 minute periods. Heyman (2005) also reported and discussed this observation. For the entire hour, the mean interarrival time is 0.0138 ms ($=E\{A\}$) and the standard deviation of the interarrival time is 0.0158 ms ($=\text{Var}\{A\}^{0.5}$). The coefficient of variation of interarrival time (SD/Mean) is 1.145; certainly not Poisson.

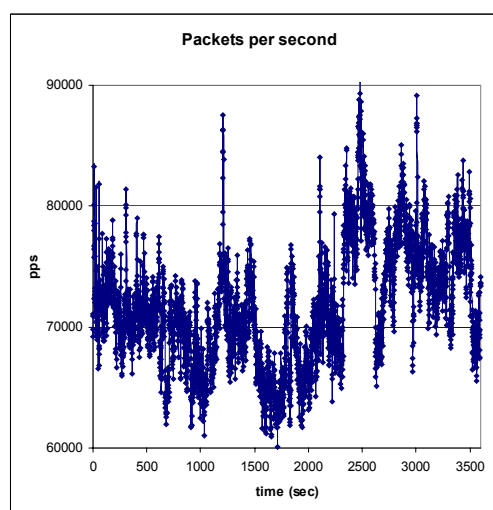


Figure 1: Packets per Second

Figure 2 is a plot of the packet size distribution. We see that packet primarily come in four or five main sizes, each with a certain probability. This observation is well known (Thompson 1997). The smallest size is 40 bytes, and the largest is 1500 bytes, with mean equal to 693.1 bytes (or about 5.54 kilobits [kb]) and a standard deviation of 672.5 bytes (or about 5.38 kb). Therefore, the coefficient of variation of packet size (SD/Mean) is 0.97. The service times used in Lindley's recursion are computed as the packet sizes divided by the link speed. For instance, for the Cleveland to Indianapolis OC-48 link, the mean service time $E\{S\} = 5.54 \text{ kb} / 2400 \text{ kb per ms} = .00231 \text{ ms}$

Figure 3 is our first use of Lindley's Recursion. It plots the running value of the expected buffer delay per packet in ms. We know the buffer delay for each packet and a running average can easily be computed from that data. Although the packet arrival rate (Figure 1) was not constant over the hour, now we also see the expected packet delay does not settle down over the hour. How does

Kingman's Upper Bound contribute to this analysis? That is, is the expected buffer delay less than the result given by Kingman's Upper Bound? For this trace, we have $E\{Wq\}_{KUB}=0.011062$ ms; and so the answer is yes.

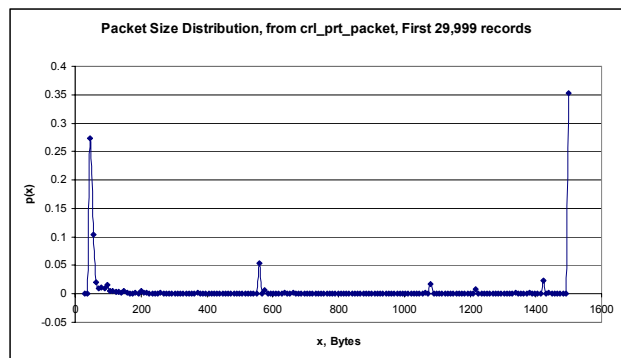


Figure 2: Packet Size Distribution

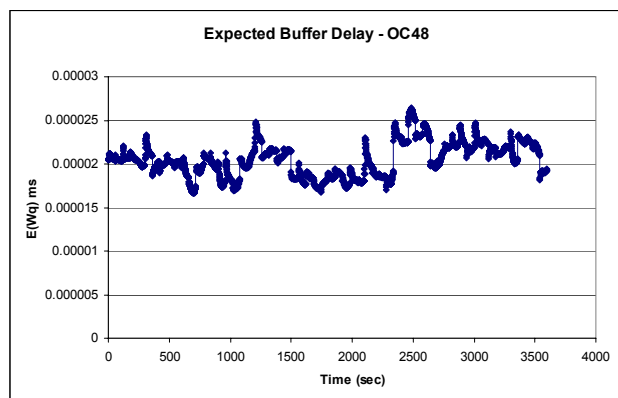


Figure 3: Expected Buffer Delay from Lindley's Recursion for OC-48

Next we investigated what happens if the speed of the link was decreased. This analysis is easily accomplished by running the data again through Lindley's Recursion, but with a slower speed line. To do this, the service times for Lindley's Recursion are re-computed by dividing the original packet sizes by a smaller line rate. We used an OC-12 line rate or about 25 percent of an OC-48. In a real network, the change in line rate may affect queue performance due to the use of TCP congestion control; however, as stated before, these models are not dealing with TCP. Those results are shown in Figure 4.

Two very important points are seen from Figure 4. First, the expected buffer delay is not constant and varies quite a bit; and second, and probably more importantly, Kingman's Upper Bound is violated. The question is "Why does this happen? The load only increased to 0.64; so Kingman's Upper Bound still should be valid."

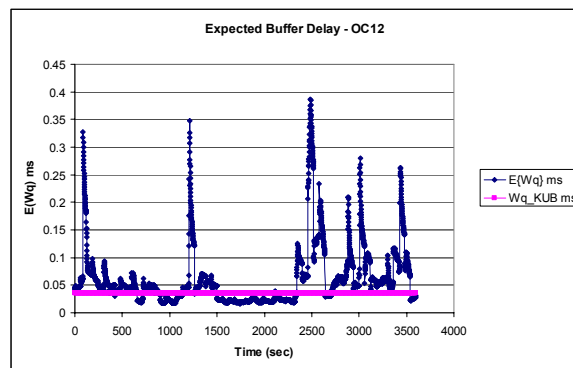


Figure 4: Expected Buffer Delay from Lindley's Recursion for OC-12

Kingman's Upper Bound depends on the packet inter-arrival times being independent of each other. It is well documented that there is some evidence that they can be long-range dependent for Internet packet arrival times. To determine if there is evidence of dependence, one can look at the autocorrelation function with lag k (see Adler et al. 1998, or Law and Kelton 2000). The autocorrelation function with lag k is shown in Figure 5.

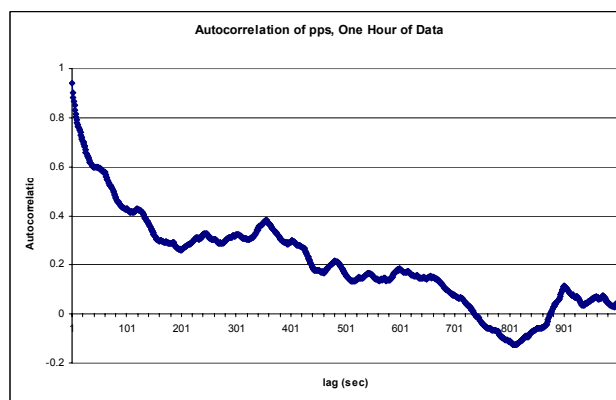


Figure 5: Autocorrelation in Packets per Second

We see that there is strong correlation especially for a lag of 10 minutes or less. Heyman (2005) also found evidence of autocorrelations in packet counts in Internet packet trace data. In addition, we used a runs test (Law and Kelton 2000) to determine if the packet interarrivals were independent. We tested the arrivals during minutes 5, 15, 25, 35, 45 and 55, and found that in all minutes tested, the hypothesis of independence in the arrivals was rejected based on the runs test.

Karaianis et al. (2004) found that for the data sets they were looking at, the packet interarrival distribution was exponential at the sub-second level. We tested the Internet2 data we were analyzing to see if that was true. Figure 6 is a plot of the Cumulative Distribution Function (CDF) for the data and an exponential distribution with the same mean for the first second of minute 5. Obviously, the

packet interarrivals are not exponential for this data; this result was also verified by the fact that the coefficient of variation (standard deviation divided by the mean) of the interarrival time was greater than one (for this example it was equal to 1.067).

Up until now we have been using Lindley's Recursion to analyze the packet congestion. Next we show how to expand its use. We have seen that for these traces, the packet interarrival times are non-stationary and dependent. Thus, to simulate the packet behavior using standard derived arrival and service probability distributions would be difficult. One could see which probability distribution best fits the arrival and size data or one could numerically generate those distributions and then use a standard discrete event simulation to study congestion. The problem with those approaches is that arrival times are drawn independently. We have seen that arrival times are not independent; therefore, this method will not result in the correct analyses. Lindley's Recursion overcame the lack of independence.

Suppose we want to study the congestion that VoIP packets would experience on this link. We can use the packet trace data directly and intertwine the voice packet into the data trace data. We then can use Lindley's Recursion to analyze the congestion the voice and data packets are experiencing. Sriram and Whitt (1986) were the first to model voice packet performance. Their results were continued by Fischer and Masi (2005 a and b). For the details of how the voice packets are simulated, see those references. Based on the codec that is used, a voice packet with a certain size is generated every T ms. We use a G.729 codec in the example below, and it has a $T = 20$ ms and a packet size of 0.528 kb. If there are N active voice conversations, then in the appendix of Sriram and Whitt there is an expression for the packet interarrival time probability distribution.

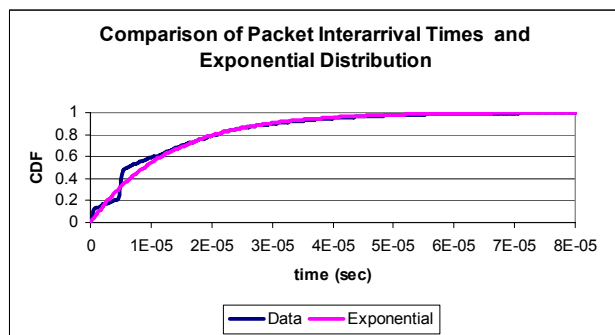


Figure 6: Comparisons of Packet Interarrival Distribution and Exponential First Second of Minute 5

We looked at the case of 2000 Erlangs of voice traffic being offered to the link with a call admission capability of 2000. Thus, 0.0176 of the calls were blocked initially. We simulated the number of active voice calls over the hour in

the standard way. That is, we simulated the number of calls present at the start and then simulated when there was a change in the number. We assumed a 3-minute holding time for a voice call. For the period of time when the first calls were fixed, we used Sriram and Whitt's results to generate voice packet arrival times. This resulted in a stream of voice packet arrivals during the hour and their sizes.

For this example, we assumed the First Come First Served service discipline in our example. Priority Queueing and Weighted Fair Queueing are other frequently used queueing disciplines (see Fischer and Masi 2005 a and b). We then used Lindley's Recursion to numerically simulate the congestion the voice and data packets were experiencing. All we needed to do was to keep track of what type of packet (data or voice) was the next arrival. We generated the voice packets for the entire hour before we used Lindley's Recursion. That structure limited the size of the arrays VBA could handle; so the examples below are just for the first five minutes of the hour. The easy solution to that issue would be to generate the voice packet only for the minute being simulated.

In Figure 7, we see that in both cases (with and without voice), the data packet delay is approximately 50 percent of the jitter. In the presence of voice packets, there was roughly a 50 percent increase in data packet delay and jitter. These measures appear to be constant over the five minutes, but that is only because of the scale of the Y axis in the figure.

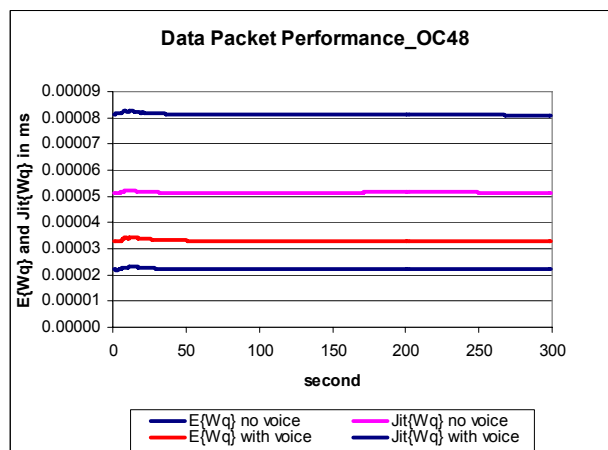


Figure 7: Data Packet Buffer Delay and Jitter With and Without Voice

Figure 8 shows the voice packet delay and jitter over the same period of time. Again, we see the jitter is much larger than the delay, and the overall measures are an order of magnitude larger than the data packet numbers. We conjecture this results stems from the fact that the data packet arrival process is not independent. For this case, the data packet load was around 0.16 and the voice packet load was

0.008. We have compared the data packet performance (no voice) with assuming a Poisson process for the data packet arrivals and found that model (with Poisson data packet arrivals) resulted in a larger expected buffer delay than was generated using Lindley's equation. From (Sriram and Whitt 1986) and (Fischer and Masi 2005a and 2005b), the voice packet arrival process tends to Poisson under certain loading conditions and so the voice packet stream would tend to be smooth (Poisson), see Figure 9. The delay and jitter the voice and data packets are experiencing seems to be constant, as is shown in Figures 7 and 8; but when we expand the Y axis, we see that is not true, as is presented Figure 10.

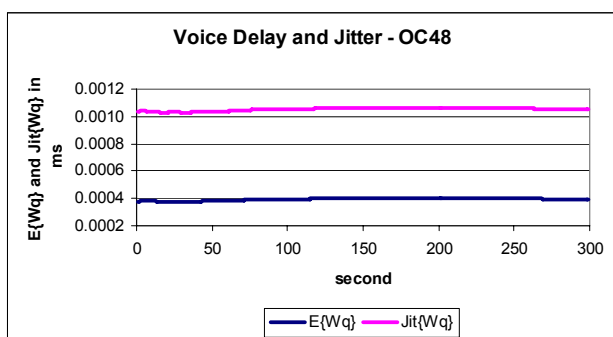


Figure 8: Voice Packet Buffer Delay and Jitter

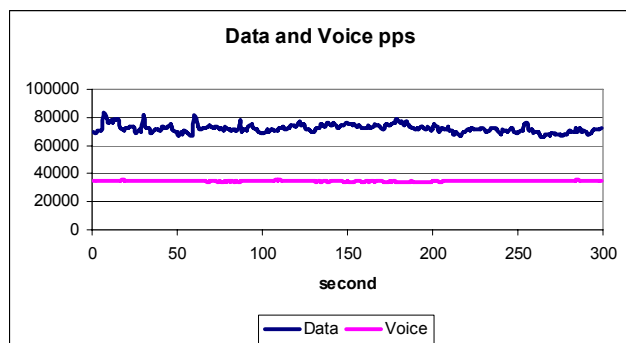


Figure 9: Voice and Data Packet Per Second

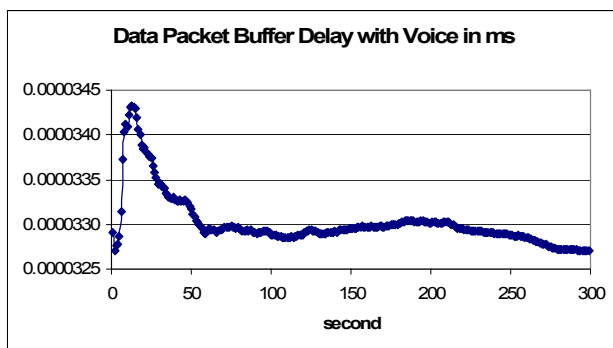


Figure 10: Data Packet Buffer Delay With Voice

5 CONCLUDING REMARKS

In this paper we have expanded the use of Lindley's Recursion to process and derive congestion results for Internet packet traces. It is extremely simple to use and largely assumption free, except for requiring FCFS queueing. We have successfully used it to analyze several Internet traces and found it to be computationally quick and yields important information about the congestion packets are experiencing within the trace. In addition, when used in conjunction with other Queueing Theory results (Kingman's Upper Bound) it can provide further insight into the congestion behavior of packets on an Internet link. Lindley's Recursion in this case demonstrates the benefits of simulating versus analytic queueing approaches that require assumptions that may not be justified. In addition, it is easier to implement than other well-known telecommunications simulation packages. We have also shown how to use it to extend the analysis of other situations of interest; in particular VoIP.

Future work will be directed at looking into using Lindley's Recursion to analyze other VoIP disciplines, Priority Queueing, and Weighted Fair Queueing. In addition, we will look into optimizing our VBA code so that we are not at the storage limits of VBA. We also will be looking into downloading other traces to see if we get similar results to the ones we have generated to date, especially dealing with the statistical properties.

ACKNOWLEDGMENTS

This work was partially funded by National Science Foundation Grant Number DMII 0140232, the National Communications System Contract Number NBCH-D-02-0039 (Task Order Number D0200390055) with Mitretek Systems, and by Mitretek Systems' internally funded research program.

REFERENCES

- Adler, R. J., R. E. Feldman, M. S. Taqqu. 1998. *A Practical Guide to Heavy Tails: Statistical Techniques and Applications*, Birkhauser.
- Cao, J. W., W. S. Cleveland, D. Lin, and D. X. Sun. 2002. Internet Traffic Tends Toward Poisson and Independent as the Load Increases. *Nonlinear Estimation and Classification*. Editors: C. Holmes, D. Denison, M. Hansen, B. Yu, and B. Mallick, Springer, New York, NY.
- Fischer, M. J. and D. M. B. Masi. 2005a. Voice Packet Arrival Models and Their Affect on Packet Performance. *Applied Telecommunications Symposium*, San Diego, CA.

- Fischer, M. J. and D. M. B. Masi. 2005b. Modeling Overloaded Voice over Internet Protocol (VoIP) Systems. In *Proceedings of the 2005 Winter Simulation Conference*. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.
- Gross, D. and C. M. Harris. 1998. *Fundamentals of Queueing Theory*, Third Edition, John Wiley, New York, NY.
- Heegaard, P. E., B. E. Helvik, and R. O. Andreassen. 2005. Application of Rare Event Techniques to Trace Driven Simulation. In *Proceedings of the 2005 Winter Simulation Conference*. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.
- Heyman, D. P. 2005. "Sizing Backbone Internet Links," *Operations Research*, Volume 53, Number. 4, 575–585.
- Karagiannis, T., M. Molle, M. Faloutsos, and A. Broido. 2004. A Non-stationary Poisson View of Internet Traffic. *IEEE INFOCOM*.
- Law, A. M. and W. D. Kelton. 2000. *Simulation Modeling and Analysis*, 3rd Edition, McGraw-Hill.
- Nogueira, A. N., P. S. Ferreira, R. Valadas, and A. Pacheco. 2004. Fitting Self-similar Traffic by a Superposition of MMPPs Modeling the Distribution at Multiple Time Scales. *IEICE Transactions on Communications*, Volume E87-B, Number 3, 678–688.
- Park, C., F. Hernandez-Campos, J. S. Marron, D. Rolls, and F. D. Smith. 2005. Long-Range-Dependence in a Changing Internet Traffic Mix. *Computer Networks* 48(3): pp. 401–422; also appears in *Statistical and Applied Mathematical Sciences Institute Technical Report* 2004.
- Paxson, V. and S. Floyd. 1995. "Wide-Area Traffic: The Failure of Poisson Modeling," *IEEE/ACM Transactions on Networking*, Volume 3, Number 3, 226–244.
- Resnick, S. I. 1997. Heavy Tail Modeling and Teletraffic Data. *The Annals of Statistics*. Volume 25, Number 5, pp. 1805–1869.
- Sriram, K. and W. Whitt. 1986. Arrival Processes in Packets Multiplexers for Voice and Data. *IEEE Journal on Selected Areas in Communications*, Volume SAC-4, Number 6, pp. 833–846.
- Thompson, K., G. J. Miller, and R. Wilder. 1997. Wide Area Internet Traffic Patterns and Characteristics. *IEEE Network*.

AUTHOR BIOGRAPHIES

MARTIN J. FISCHER is a senior fellow at Mitretek Systems. His experience includes network design and performance analysis in telecommunications. He has published approximately 40 articles in refereed journals. He received his doctorate degree in Operations Research from Southern Methodist University. His e-mail address is mfischer@mitretek.org.

DENISE M. BEVILACQUA MASI is a senior principal engineer at Mitretek Systems. Her experience and research interests include queueing theory and simulation applied to telecommunications networks. She received her doctorate degree in information technology and engineering at George Mason University. Her e-mail address is dmasi@mitretek.org.