On the Predictability of Data Network Traffic

Khushboo Shah
Dept. of Electrical Eng.
University of Southern California
khushboo@usc.edu

Stephan Bohacek
Dept. of Electrical and Comp. Eng
University of Delaware
bohacek@eecis.udel.edu

Edmond Jonckheere
Dept. of Electrical Eng.
University of Southern California
jonckhee@usc.edu

Abstract

The predictably of data network traffic is assessed. Different topologies and types of traffic are studied. Linear and nonlinear AR(MA) models as well as state space and models based on canonical correlation are employed. These predictors are compared against two simple predictors: 1. the prediction is the mean value of the time series, 2. the prediction is the last observation. The significant conclusion is that the dynamic predictors fail to perform significantly better than the simple predictors over higher frequencies. The implication of this result with regard to active queue management is discussed.

1 Introduction

There has been a large body of work focused on developing dynamic controllers for computer data networks [1], [2], [3]. However, before a controller can be developed, the open-loop system must be understood. This paper examines the predictability of data network traffic through models, since AQM relies on anticipation of immitent queue overflow. The main result is that network traffic is not predictable at high frequencies. Thus, there seems to be little hope of developing a controller to damp out the high frequency variations of TCP/network's dynamics. This result does not contradict other work that produced controllers and demonstrated their effectiveness over lower frequency band.

Network traffic analysis has been the focus of countless papers following three avenues of approach. First, much work has focused on measuring traffic flow across real networks [4], [5]. These measurements have attempted to quantify the variation or growth in traffic at very long time scale. The second and very active area of investigation in network traffic has been the study of steady state sending rate produced by a single TCP flow. This work has led to the TCP-friendly equations [6], [7], which establish a steady state relationship between the round-trip time, the packet loss probability, and the TCP sending rate. The third approach has been the dynamic behavior of TCP. Much work has focused on a first principles approach to modeling the dynamics of TCP [8], [9], [10], [11], [12]. The approach followed in the present paper is distinct for these other works in that no modeling assumptions are made, that is, the models are developed via the "black box" approach by collecting a large data record and applying time series modeling techniques. In [13], the variability of TCP dynamics was demonstrated. In [14], it was shown that TCP can display chaotic dynamics. These results led to the suspicion that TCP traffic is difficult to predict. This paper seeks to confirm this suspicion.

The result that the dynamic aspects of TCP are not predictable is based on experimentation on two topologies (dumbbell and parking lot) and two types of traffic (FTP and HTTP) for varying drop probability. Before the investigation begins, the types of models considered and the simulation setup are discussed.

2 Models and Simulation Set-up

2.1 Prediction Models

Many classes of models are considered. These classes include linear AR models [15] with or without input

$$y(k) = \sum_{i=1}^{L} a_i y(k-i) + \sum_{i=0}^{L-1} b_i u(k-i) + w(k),$$

where w is a noise. Also, nonlinear AR models were considered,

$$\begin{aligned} y(k) &= \sum_{i=1}^{L} a_i y(k-i) + \sum_{d\in D}\sum_{i=1}^{L} a_{d,i} y(k-i)^d + \ldots \\ &\quad + \sum_{i=0}^{L-1} b_i u(k-i) + \sum_{d\in D}\sum_{i=0}^{L-1} b_{d,i} u(k-i)^d \ldots . \\ &\quad + w(k), \end{aligned}$$

where the D is one of the following sets:

$$\begin{aligned} D &= \{1,2,3\ldots,10\}, \textit{for "All" nonlinearities} \quad (1) \\ D &= \{1,2,4,6,8,10\}, \textit{for "Even" nonlinearities} \\ D &= \{1,3,5,7,9\}, \textit{for "Odd" nonlinearities} \end{aligned}$$

In the case where the input is not used, the coefficients b_i are set to zero. Standard least-squares techniques were used to estimate the coefficients of these AR models. Linear state space models are of the form

0-7803-7896-2/03/$17.00 ©2003 IEEE

Proceedings of the American Control Conference
Denver, Colorado June 4-6, 2003

$$x(k+1) = Ax(k) + B_1u(k) + B_2w(k)$$
$$y(k) = Cx(k) + Dw(k)$$

Here, estimates of the parameters were identified using techniques described in [15].

Lastly, prediction models based on the canonical correlation analysis (CCA) were considered. The two models that have been investigated here are nonlinear AR and state space models. Both models are implemented as described in [16].

Two simple predictors were employed to compare the performance of the dynamic models described above. The first predictor is written as

$$\widehat{y}(k+1) = y_{mean} \tag{2}$$

Hence, this predictor simply uses the mean as a prediction. The second simple predictor is

$$\widehat{y}_{simple}(k+1) = y(k). \tag{3}$$

In this case the prediction is simply the last observation.

2.2 Measures of Model Fit:

The most common measure of predictability is the mean square error (MSE) defined as, $MSE = E((\widehat{y}_{est} - y)^2)$ and the normalized mean square error (NMSE) defined as, $NMSE = \frac{E((\widehat{y}_{est}-y)^2)}{E((\widehat{y}_{mean}-y)^2)}$. One can view this normalization as a comparison between two predictors. One predictor yields the prediction $\widehat{y}_{est}$ while the other predictor trivially predicts the mean.

Following this interpretation of NMSE as a comparison between two predictors, we consider the relative performance of the simple predictor (3) defined as, $NMSE - SP = \frac{E((\widehat{y}_{est}-y)^2)}{E((\widehat{y}_{simple}-y)^2)}$, where y_{simple} is given by (3).

Note that if NMSE is near 1, then the dynamic predictor performs about the same task as just using the mean as the prediction. Thus, only when both the NMSE and NMSE-SP are small can we conclude that the dynamic predictor is a good predictor.

2.3 Model Order Selection:

Because the best choice of the filter order, L, is generally not known a priori, it is usually necessary in practice to postulate several model orders. Many criteria have been proposed as allegedly objective functions for selecting the AR model order. The two best known ones are Akaike's Information Theoretic Criterion, AIC [17], which has the form (for gaussian disturbances), $AIC[L] = N\ln(MSE_L) + 2L$ and Rissanen's Minimum Description Length Criterion, MDL [18], which has the form $MDL[L] = N\ln(MSE_L) + L\ln(N)$, where N is the length of the data record. Here the order L is selected to minimize the MDL criterion.

2.4 Simulation Setup

We used the Network Simulator (ns-2) developed by LBNL to perform our simulations. Ns is a discrete event simulator widely accepted for networking research. We studied many environment set-ups: two types of traffic, and two topologies for variable drop probability system. In this system, the queue imposes a drop probability on every arriving packet. This drop probability is uniformly distributed over [0.0295, 0.0305]. Hence, at time step k, the probability p_k is set for the time period $[kT, (k+1)T)$, where T is the sample period. We consider sample periods from 10ms to nearly an hour and a half. In order to keep the scale of the packet arrivals the same for all sample periods, we define y_{k+1} to be the normalized packet arrivals over the period $[kT, (k+1)T)$. The normalization is done by dividing the observed packet arrivals by the link speed which is the maximum number of packets per time period $[kT, (k+1)T)$. In this set-up, drops could occur if the queue fills. However, the queue size is taken sufficiently large and the drop probability is taken large enough, so that the queue does not fill up.

Both the dumbbell and parking lot topologies are investigated. The dumbbell topology is shown in Figure 1. The nodes S_i (i = 2,...,6) are set as the sources and the nodes D_i (i= 7 ,...,11) are set as the destinations. The monitored link, the bottleneck link, is 0 to 1. The parking lot topology is a more complicated topology and is shown in Figure 2. The nodes 0, 8, 10, 12 are set as the sources and the nodes 7, 9, 11, 13 are set as the destinations. For this topology, the monitored link is the one from 4 to 5.

FTP traffic was modeled as long lived TCP traffic, that is, for each source-destination pair a single TCP connection sent data for the entire simulation. The starting time of the flows was varied slightly randomly so that each simulation was different. HTTP traffic was modeled by a collection of flows with an ON/OFF behavior. Specifically, a single HTTP connection was made up of a single TCP flow. This flow transmits a single file. The size of the file is a random variable with Pareto distribution with shape parameter equal to 1.06 and minimum file size equal to 10000. These parameters are common estimates of the files size distribution found on the web [19],[20]. Upon completion of the transmission of the file, the connection lies dormant for a period of time that is exponentially distributed with mean 60 seconds.

Each simulation was run for extremely long runs to ensure that all parameters were accurately estimated. For example, most simulations used over 30000 sample points. Hence, when the sample period was 100 seconds, the simulation ran for 3,000,000 secs or nearly 35 days. Indeed, it is easily argued that the simulation environment is overly generous and that any predictor that would be deployed would have to perform well in far more difficult situations.

3 Models Type Selection

We begin by investigating which model, linear AR, nonlinear AR, statespace, nonlinear AR (CCA), or statespace (CCA), yields the best predictor. To this end,

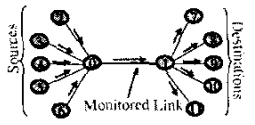

Figure 1: Dumbbell topology. The traffic is sent from sources to destinations. Link 0-1, bottleneck link, is monitored.

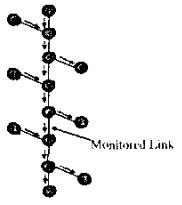

Figure 2: Parking lot topology. The traffic is sent from sources to their downsteam destinations. Link 4-5 is monitored.

several environments were simulated. Here, the results from one environment are presented.

The topology being considered is dumbbell (Figure 1) and the traffic is FTP. The analysis is 2-fold: first, an "input -output" system (where p_k is the input and y_k is the output); second, a "free" system (where there is no input to the system and y_k is the freely generated signal). The comparison between the NMSE for both types of systems for six models (linear AR, linear ARMA, nonlinear AR, statespace, nonlinear AR (CCA), statespace (CCA)) is performed. Furthermore, the nonlinear AR models are investigated in detail by considering various nonlinearities for both systems.

Figure 3 shows a comparison between NMSE for various models for both types of systems. For both systems (Figures on the left and on the right in Figure 3), at sampling period 0.01, linear AR works slightly better $(5-10\%)$ than the rest of the models. As the sampling period increases, the NMSE becomes nearly the same for all the predictors. Hence, we can conclude that the type of predictor does not matter for the prediction of y_k. Furthermore, considering the nonlinearities in the network (eg. queue overflow, dividing congestion window by 2, etc.), one would expect that nonlinear models might perform better. However, our simulation cases, the nonlinearities do not appear to improve the prediction quality. The reason behind that is that the residual error from the linear predictor is gaussian as shown in Figure 5. Hence, we can conclude that the nonlinear predictors would not achieve better results than the linear predictors for the prediction of y_k.

Comparing the Figures on the left and on the right (Figure 3), we see that for small sampling periods (0.01, 0.05), the NMSE are similar. Thus we can deduce that,

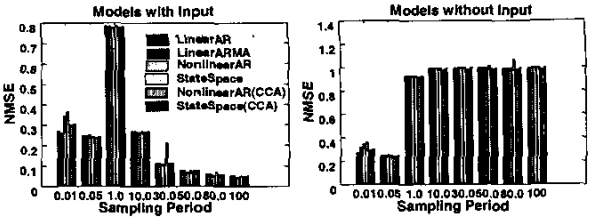

Figure 3: Comparison of various models for the input-output system and the free system. Topology is dumbbell and traffic is FTP.

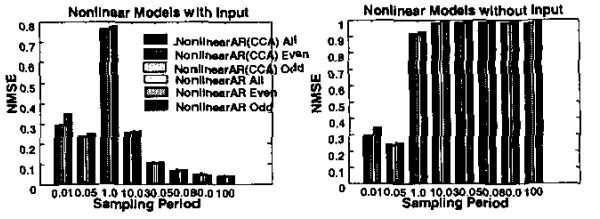

Figure 4: Comparison of various nonlinear models for the input-output system and free system. Topology is dumbbell and traffic is FTP.

at smaller sampling periods, the p_k does not have much effect on the prediction of y_k, only the past y_k is important. In contrast, at large sampling periods, NMSE decreases for the input-output system while NMSE increases for the free system. This means that the p_k begins to have a great effect while the past y_k does not affect the prediction as much.

Next, we compare nonlinear models (nonlinear AR and nonlinear AR (CCA)) to see which nonlinear model is better and which nonlinearities in particular play an important role in prediction. "All" means all the nonlinearities are considered; "even" means all the even and "odd" means all the odd nonlinearities are considered (1).

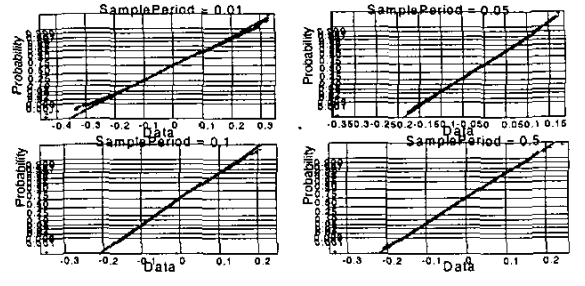

Figure 5: Probability versus residual error (from linear AR) plot for different sampling periods. The black curve fits the red line, which means that the residual error is gaussian.

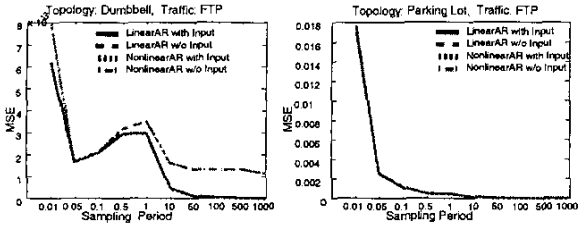

Figure 6: MSE versus sampling period for dumbbell and parking lot topologies for FTP traffic.

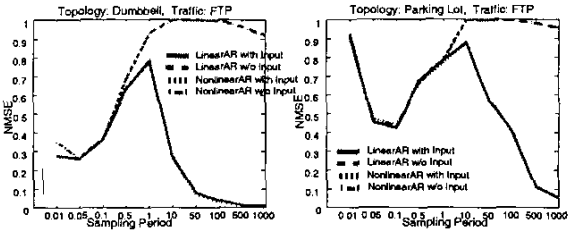

Figure 7: NMSE versus sampling period for dumbbell and parking lot topologies for FTP traffic.

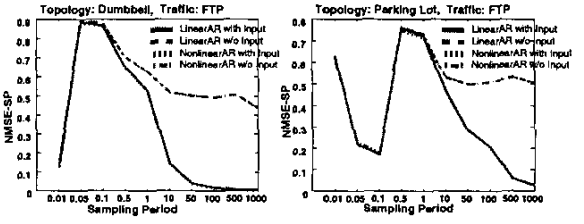

Figure 8: NMSE-SP versus sampling period for dumbbell and parking lot topologies for FTP traffic.

Figure 3 shows the comparison between all the nonlinear models. Observe that there is a slight gap between nonlinear AR and nonlinear AR (CCA) for small sample period. As the sample period gets larger, the gap gets smaller. Identification of nonlinear AR models is computationally much faster than nonlinear AR (CCA), as the later involves large matrix computation, SVD, etc. Since the gain in prediction error by using nonlinear AR (CCA) is only slight (less than 5%), we use linear AR and nonlinear AR models for further prediction analysis.

Figure 4 also shows that there is not much difference in NMSE when different nonlinearities are used. Specifically, models nonlinear AR "Even" and nonlinear AR "Odd" give the same prediction error as model nonlinear AR "All". Since, nonlinear AR "All" is significantly simpler than the other nonlinear AR models, we restrict our attention to nonlinear AR "All".

Similar tests were carried out for other types of nonlinearities, including cross terms between the past and the present samples. Moreover, the study was extended by investigating Fractional ARIMA models. But all these tests yield similar conclusion as above. In addition, other simulations for other topologies and other types of traffic also yield similar results. Thus, we only consider linear AR and nonlinear AR "All" models.

4 Predictability of FTP and HTTP Traffic

In this section, we investigate the predictability of FTP and HTTP traffic for the variable drop probability. The study is divided into a 4-fold study, two different topologies (dumbbell and parking lot) and two types of traffic (FTP and HTTP). First, we discuss the results for FTP traffic and then for HTTP traffic. And last, we study the worst case prediction scenario. We restrict our attention to the input-output system. The predictability for to free system is similar to the input-output system presented here.

4.1 Predictability of FTP Traffic with Variable Drop Probability

Figures 6, 7, 8, and 9 show the MSE, NMSE, NMSE-SP, and the order, respectively, for both topologies. In these simulations, the traffic is FTP. The models under investigation are linear AR and nonlinear AR. Each model makes a one step ahead prediction of packet arrivals given the packet arrivals and drop probability (labeled as "with input") or given only the past packet arrivals (labeled as "without input").

First consider the performance at small sample periods. Observe that the MSE, NMSE, and NMSE-SP are the same regardless of whether input is used or not (Figures 6, 7, and 8). Hence, y mainly depends on its past and the drop probability does not play a significant role. This conclusion seems to hold for both parking lot and dumbbell topologies. By examining Figure 7, it appears that the dynamic model outperforms a predictor that just uses the mean as the prediction. This conclusion seems to hold for both the dumbbell topology and the parking lot topology. However, for the very small sample period of 10ms, NMSE is large for the parking lot topology . This large error may be due to the fact that the system order for these small sample periods are very large and we limited the system order to less than 50. Note that Figure 9 shows that for small sample periods, the system order is large. This seems to indicate that the traffic is described by a complicated dynamical system. However, for these small sample periods, it is possible to use the simple predictor (3). Such a simple predictor performs quite well for small sample periods (Figure 8). Hence, we conclude that, while the traffic may incorporate some complicated dynamics, a significant part of the dynamics is simple.

One might expect that for very small sample periods such as 10ms, the predictor (3) should perform well. However, the packet arrivals make a discrete event system. Hence, for very small sample periods, there is

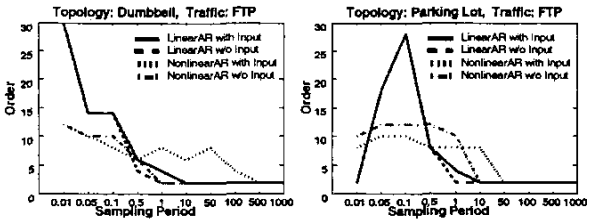

Figure 9: Order of the system versus sampling period for dumbbell and parking lot topologies for FTP traffic.

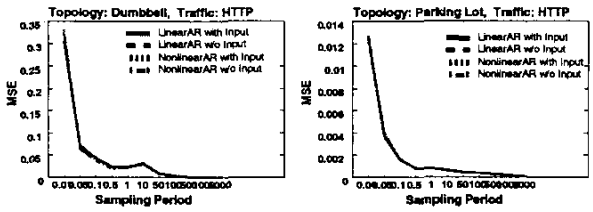

Figure 10: MSE versus sampling period for dumbbell and parking lot topologies for HTTP traffic.

either one arrival or none. Such a signal might not be well modeled by (3).

For large sampling periods, an increase in the sampling period leads to decrease in the MSE for the models with input whereas it leads to an increase in the MSE for the models without input (Figure 6). Similarly, Figures 7 and 8 indicate that a significantly better predictor can be obtained if drop probability is utilized. Furthermore, Figure 9 shows that for large sample periods the system order is small. By examining the coefficients, it can be seen that, for these sample periods, the arrivals solely depends on the drop probability. Essentially, for large sample periods, the predictor $\widehat{y}(k+1)$ is an approximation of the function $E\left(\widehat{y}(k+1)|p_k\right)$.

Comparing the left and right plots in Figures 7 and 8 it can be observed that, as the sample period is increased, the prediction error decreases faster for the dumbbell topology than for the parking lot topology. This is due to the more complicated dynamics of the parking lot topology. Specifically, it seems that the transients take longer to die out in the case of the parking lot topology. This implies that in order to determine the average behavior of the link, it is necessary to average over time windows at least 500 seconds long. It is plausible that the time windows would have to be even larger for more complicated topologies found in the Internet.

Finally, note that the nonlinear AR has nearly the same or slightly less prediction error than the linear AR.

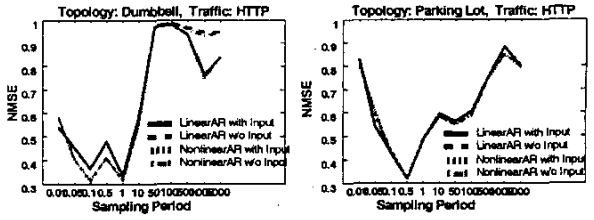

Figure 11: NMSE versus sampling period for dumbbell and parking lot topologies for HTTP traffic.

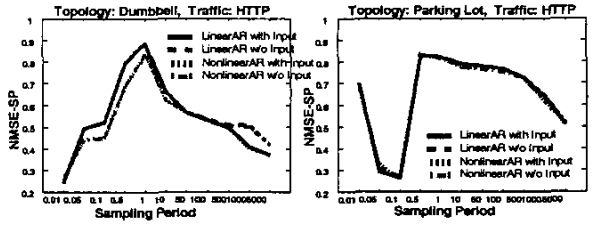

Figure 12: NMSE-SP versus sampling period for dumbbell and parking lot topologies for HTTP traffic.

4.2 Predictability of HTTP Traffic with Variable Drop Probability

Figures 10, 11, and 12 show the MSE, NMSE, and NMSE-SP, respectively, for both topologies. In these simulations, the traffic is HTTP.

HTTP traffic is more stochastic as there are many factors influencing the traffic. As a consequence, even for the dumbbell topology, the degree of predictability for all the models is the same (Figure 10). As in the case of FTP traffic, the prediction error is fairly large for small sample period. However, when normalized by the variance of the signal, we see that the NMSE is small for these sample periods. For example, in all cases, the NMSE is less than 0.50 when sample period is 100ms. Furthermore, as above, nonlinear AR performs slightly better at smaller sampling period than linear AR (Figures 7, 12).

As the sampling period gets larger, we see a strong discrepancy between FTP and HTTP traffic. One might expect the NMSE to be small for larger sampling periods. But that does not hold true in the case of HTTP traffic. This is due to the variability of the HTTP traffic. For example, during one sample period there may be just a few long-lived TCP flows, while during the next sample period, there may be many.

4.3 Predictability in the Worst Case Environment

While the above showed that in some cases the dynamic predictor works well, it seems that its performance is dependent on the topology and type of traffic.

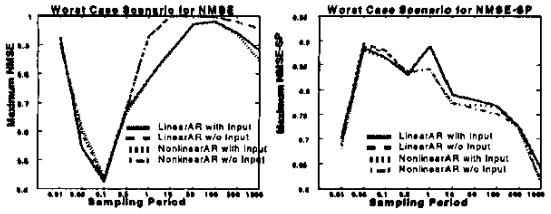

Figure 13: Maximum NMSE and NMSE-SP versus Sampling Period. NMSE and NMSE-SP are maximized over both the topologies and both types of traffics.

In general, the exact mix of traffic is not known in advance and the topology is not fixed. Thus, we demand that a predictor works well in all environments. Figure 13 shows the worst case normalized prediction error. Specifically, the normalized error for a particular sample period is normalized over both topologies and both types of traffic. Note that in no case can we expect the error variance from the dynamic predictors to be less than half the size of the error variance due to the simple predictors.

5 Conclusion

Improving congestion control and queue management algorithms have been active areas of research over the past ten years. This work says that there is a possibility of controlling at larger time scale when the focus is centered around end-user action. On the other hand, at smaller time scale where the TCP's dynamics dominates the traffic characteristics, control does not seem to be possible. Finally, in TCP's steady state, the dynamical predictors do not show improvement over simple predictor $E(y|p)$. Hence, the mean of the packet arrivals is a good estimate of end-user behavior.

References

[1] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," vol. V 1 N.4, pp. 397–413, August 1993.

[2] S. Athuraliya, V. H. Li, S. H. Low, and Q. Yin, "REM: Active queue management," in *IEEE Network*, May/June 2001.

[3] S. Kunniyur and R. Srikant, "Analysis and design of an adaptive virtual queue (AVQ) algorigthm for active queue management," in *SIGCOMM*, 2001.

[4] C. A. for Internet Data Analysis (CAIDA), "http://www.caida.org."

[5] K. Thompson, G. J. Miller, and R. Wilder, "Wide-area internet traffic patterns and characteristics," *IEEE Network*, vol. 11, 1997.

[6] S. Floyd, "Connections with muliple congested gateways in packet-switched networks part 1: One-way traffic," *Computer Communication Review*, vol. 21, pp. 30–47, 1991.

[7] M. Mathis, J. Semke, J. Mahdavi, and T. Ott, "The macroscopic behavior of TCP congestion avoidance algorithm," *Computer Communication Review*, vol. 27, 1997.

[8] S. Low, F. Paganini, and J. Doyle, "Internet congestion control," *IEEE Control Systems Magazine*, vol. 22, pp. 28–43, 2002.

[9] S. Kunniyur and R. Srikant, "End-to-end congestion control: Utility functions, random loss and ECN marks," in *Proceedings of INFOCOM 2000*, March, 2000.

[10] S. Bohacek, J. Hespanha, K. Obraczka, and J. Lee, "Analysis of a TCP hybird model," in *In Proceedings of the 39th Annual Allerton Conference on Communication, Control and Computing,*, 2001.

[11] F. Baccelli and D. Hong, "TCP is max-plus linear," *Proc. of ACMSIGCOMM*, vol. 30, No. 4, pp. 219–230, 2000.

[12] Y. Chait, C. Hollot, V. Misra, H. Han, and Y. Halevi, "Dynamic analysis of congested TCP networks," in *ACC*, 2002.

[13] Y. Joo, V. Ribeiro, A. Feldmann, A. C. Gilbert, and W. Willinger, "TCP/IP traffic dynamics and network performance: A lesson in workload modeling, flow control, and trace-driven simulations," *SIGCOMM Computer Communication Review*, vol. 31, 2001.

[14] A. Veres and M. Boda, "The chaotic nature of TCP congestion control," in *INFOCOM*, pp. 1715–1723, 2000.

[15] L. Ljung, *System Identification; Theory for the User*. Prentice Hall, 1987.

[16] K. S. Edmond Jonckheere and S. Bohacek, "Time series modeling of internet traffic for intrusion detection," in *American Conference on Control*, June 2002.

[17] H. Akaike, "A new look at the statistical model identification," in *IEEE Transactions on Automatic Control , AC-19*, pp. 716–723, 1974.

[18] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, pp. 465–471, 1978.

[19] B. A. Mah, "An empirical model of HTTP network traffic," (Kobe, Japan), Published in the Proceedings of INFOCOM, April 7-11,1997.

[20] M. E. Crovella and A. Bestavros, "Self-similarity in world wide web traffic: Evidence and possible causes.," in *In the Proceedings of the ACM SIGMETRICS Confernece on Measurement and Modeling of Computer Systems.*, (Philadelphia, PA), pp. 160–169, May 1996.