SMAQ: A Measurement-Based Tool for Traffic Modeling and Queuing Analysis Part II: Network Applications

San-qi Li, Sangkyu Park, and Dogu Arifler

University of Texas at Austin

ABSTRACT SMAQ is a measurement-based tool for integration of traffic modeling and queuing analysis. It can be used in a variety of network design areas. For instance, it can be used as a traffic generator to generate various traces for network testing. It also provides numerical solutions of the queue length and loss rate performance for transport of multimedia traffic. Several application modules are built into the tool for the evaluation of statistical multiplexing, buffer dimensioning, and link bandwidth allocation. Other examples include the evaluation of traffic shaping, local congestion control, and the modeling of wireless channel dynamics. As one will find, the SMAQ tool indeed provides a solution technique for network engineers to solve many of the current design issues. A trial version of the Web-based SMAQ tool can be found on our Web site at http://www.apollo.ece.utexas.edu.

The Statistical Match and Queuing (SMAQ) tool can be widely used in solving various system design issues. First, it can compute the bandwidth requirement in a given finite-buffer system, or evaluate the effect of buffer dimensioning in a fixed-bandwidth system for transport of certain traffic. This is because the folding algorithm adopted in the SMAQ tool is capable of providing accurate queue length and loss rate solutions of a finite-buffer system, whereas most other queueing techniques have to make infinite-buffer assumption. Second, the SMAQ tool can obtain the statistical multiplexing performance and provide call admission control solutions for transport of aggregate traffic. This is accomplished through the measurement-based modeling technique, which can capture the aggregate statistics of different traffic streams without having the so-called state-space explosion problem as in nonmeasurement-based modeling [1]. Furthermore, the SMAQ tool can be applied to the performance analysis of various priority discarding and congestion control schemes. This analysis is achieved because our queuing technique allows both arrival and service processes to be adaptive and dependent on the present queue length.

One major advantage of measurement-based modeling is that it provides a mechanism for integration of computer simulation, theoretical analysis, and experimental testing, which are the three main vehicles for complex system analyses. For instance, in the design of a traffic-shaping mechanism at a network entry point, the key issue is how to evaluate the effect of traffic shaping on overall network performance improvement. The major difficulty is the modeling of a "shaped" source before entering the network. Through the statistical measurement of "shaped" traces generated by computer simulation from a given traffic source, one can use the SMAQ tool to build a matched model describing the shaping characteristics. Such an integrated approach of computer simulation and stochastic analysis will greatly help network designers evaluate many complex performance issues. Similarly, one can build a library of circulant-modulated Poisson process (CMPP) models matched to various sources, which can be used as a traffic generator to load a real or simulated network for experimental testing and network analysis. Note that traffic generator is a critical component for network testing and simulation study. A few traffic generator equipments available to date are practically useless without support of measurement-based traffic modeling.

In Part II of this article, we discuss several powerful potential applications of the SMAQ tool in system analysis and design. Most of the examples presented are current challenges facing system designers in the network industry. The lack of a measurement-based analytical tool sometimes prevents designers from obtaining an optimally designed network. The introduction of the SMAQ tool [2] provides a step toward resolving the current network design issues.

The following sections provide various applications of the SMAQ tool with examples of resolving the current design issues.

THE SOURCE LIBRARY

The SMAQ tool provides a source library that consists of three types of representations, namely real trace, statistical functions, and stochastic model, each of which can be used as a source of traffic input. Real traces are collected from representative traffic sources. A matched CMPP can be constructed for each source in the library. Aggregate traffic can also be formed from these sources either by the superposition of sta-

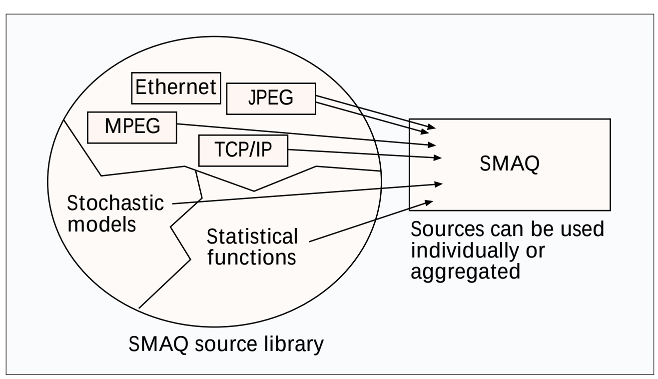

Figure 1. *The SMAQ tool and source library.*

The research reported here was supported by NSF under grant NCR-9314387.

0163-6804/98/$10.00 © 1998 IEEE

tistical functions or by the direct aggregation of the traces. The constructed CMPPs can be used in various applications, including traffic generation and network performance analysis [3]. A diagram showing the structure of SMAQ's source library is depicted in Fig. 1.

The SMAQ tool has the capability to maintain a dynamic source library. This feature enables addition of new representative traffic traces as they become available, modification of the already existing traces, and removal of the old ones from the library.

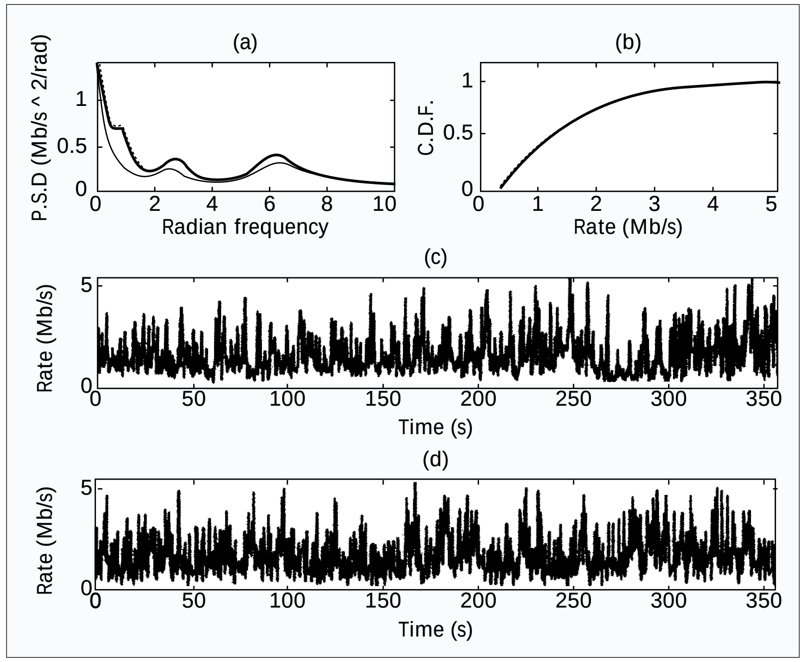

Figure 2. *Comparison of Ethernet trace (dotted line) with matched CMPP (solid line): a) power spectrum; b) rate distribution; c) original filtered trace; d) CMPP sequence.*

TRAFFIC GENERATION

Traffic generation is a very important part of network testing. However, the traffic generators available today are not useful without measurement-based traffic modeling. Usually, every real traffic trace takes huge storage space, making traffic generation an expensive process. Thus, generation of massive traffic flows in a large high-speed network for testing and simulation is impractical, if not impossible.

The SMAQ tool can generate a matched CMPP to act as a traffic generator of a certain kind or an aggregation of the same or different kinds of traffic traces. In this process, we can use the source library to select representative traffic traces. Each traffic generator (i.e., a CMPP) is simply represented by two vectors α and γ, as described in Part I, which do not require much storage space.

In order to demonstrate the effectiveness of the SMAQ's traffic generation ability, we choose three different kinds of traffic traces, namely Ethernet, JPEG, and MPEG, and compare them with those formed by the matched CMPPs [3] (Figs. 2–5).

In Fig. 2, the power spectrum of Ethernet data is matched by a CMPP whose dimension is $N = 401$ with an error tolerance of 30 percent. This is due to the fact that we cannot find a feasible CMPP if the size of the circulant is ≤ 401 unless we allow a certain degree of error tolerance. Although the matching error in the power spectrum is relatively significant, the distribution matching is excellent, and we find that the dynamics of two traces in Fig. 2c and 2d are also in a good agreement. Note that the selection of a large N is mainly for matching the heavy tail portion of the distribution. For queuing analysis, a small N can be selected [3].

In Fig. 3, we investigate a 2-hr JPEG segment of the movie *Star Wars*. A CMPP is matched at $N = 401$ with 0 percent power spectrum error tolerance. As we can see, both the power spectrum and distribution function are matched well except for a small tail portion of the distribution. This is why the peak rate of the CMPP can only reach 9 Mb/s, whereas the peak rate of the real trace has no restriction. Nevertheless, the CMPP sequence captures the overall feature of the real JPEG trace, as shown in Fig. 3c and 3d.

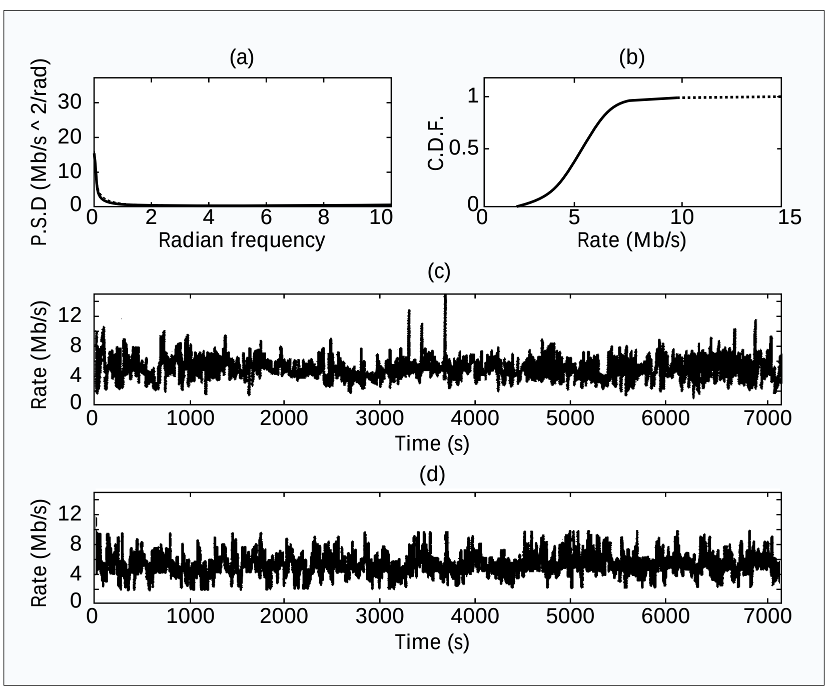

Figure 3. *Comparison of JPEG trace (dotted line) with matched CMPP (solid line): a) power spectrum; b) rate distribution; c) original filtered trace; d) CMPP sequence.*

In Fig. 4 we study a smoothed MPEG trace. The matching results of the trace by a CMPP at $N = 401$ are shown. In Fig. 5, we model Internet traffic collected at a particular time of day for a finite duration of time. A feasible matched CMPP is found at $N = 301$ with 0 percent error tolerance. As we can see, the matched power spectrum and matched distribution function results are in good agreement with those of the traces.

TRAFFIC SHAPING

Traffic shaping (or smoothing) involves delaying the packets (or cells in the context of ATM) at the source or network access device rather than sending them immediately and having to store them in the network. By using

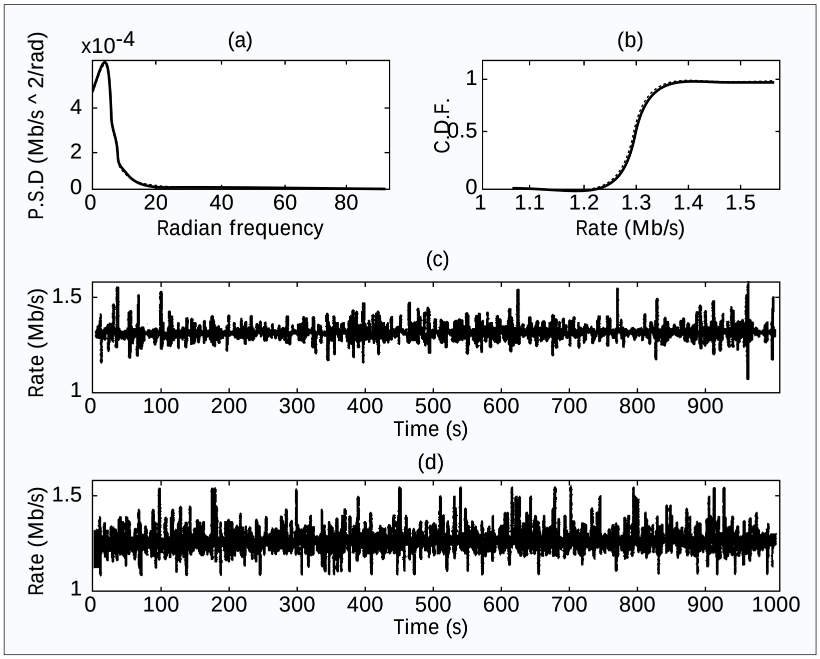

Figure 4. *Comparison of MPEG trace (dotted line) with matched CMPP (solid line); a) power spectrum; b) rate distribution; c) original filtered trace; d) CMPP sequence.*

the network only to transport packets, not to store them, we can effectively prevent one source from perturbing the other excessively.

We can consider the traffic shaper as an access controlling scheme with source traffic as its input, and the shaped (or smoothed) traffic as its output to network. Shaping is usually done by means of usage parameter control (UPC) and network parameter control (NPC). Both UPC and NPC can constrain the input traffic on peak rate and burst size so that the output from the traffic shaper is a smoothed version of the input traffic. For example, the most popular input smoothing scheme using UPC/NPC is leaky bucket control. Most of the time, the shaped traffic that enters the network is of the most interest in evaluating the performance of the network [4, 5].

Another effective smoothing scheme proposed for video applications is to delay the traffic for some time T, such that arrivals during the time interval $[t - T, t)$ will be scheduled for smooth transmission during the next time interval $[t, t + T)$. For example, T can be selected equal to the interarrival of adjacent I-frames in MPEG video. In consequence, the burstiness of I-frames is smoothed out at the network entry point, which may otherwise congest the network. By collecting the statistics at the output of the smoothing buffer, we can study the effect of this scheme on network performance using the SMAQ tool.

To the best of our knowledge, no analytical results are available to formulate the outputs of these schemes other than to capture partial stochastic properties. Using the measurement-based SMAQ tool, we can collect the statistics of the output and construct a CMPP to study the effects of traffic shaping on network performance [6].

As an example, we choose an MPEG trace with an average rate of 778 cells/s with 5 I-frames/s. We regulate the MPEG trace using the delaying scheme mentioned above to obtain the shaped trace. We choose to delay the traffic for a time that is equal to the interarrival time of the I-frames. Since we have 5 I-frames/s, original MPEG traffic is delayed by 0.2 s.

We plot the analytical queuing solutions for the shaped MPEG traffic in Fig. 6 using the Queuing Solutions Module of the SMAQ tool, setting buffer size to $K = 511$ and utilization to $\rho = 0.8$. With the given buffer size and utilization, the analytical mean queue size is 45.67 cells, standard deviation of the queue size is 122.8 cells, and the mean loss rate is 0.0056. The results are in agreement with those obtained by a fluid flow simulation that uses the original shaped trace. The fluid flow simulation computes the mean queue size as 37.70 cells, the standard deviation of the queue size as 118.2 cells, and the mean loss rate as 0.0037. Note that the fluid flow approximation usually underestimates the queuing/loss solutions due to the ignored local dynamics of cell departures and interarrival times.

Congestion Control

In many real-time applications such as voice and video, the source input can be divided into packets with different priorities. In such applications, high-priority packets may provide an approximation to the original input. In order to satisfy the quality of service (QoS) of applications, no high-priority packets should be lost unless all the lower-priority packets have been discarded. By properly designing the buffer threshold to discard low-priority packets in a congestion period, we can substantially reduce the loss probability of high-priority packets. For example, if we denote the buffer threshold for the ith priority class by K_i, the arriving packet of the ith priority class will be discarded when the queue size exceeds K_i (Fig. 7).

We can use the SMAQ tool to design buffer thresholds to

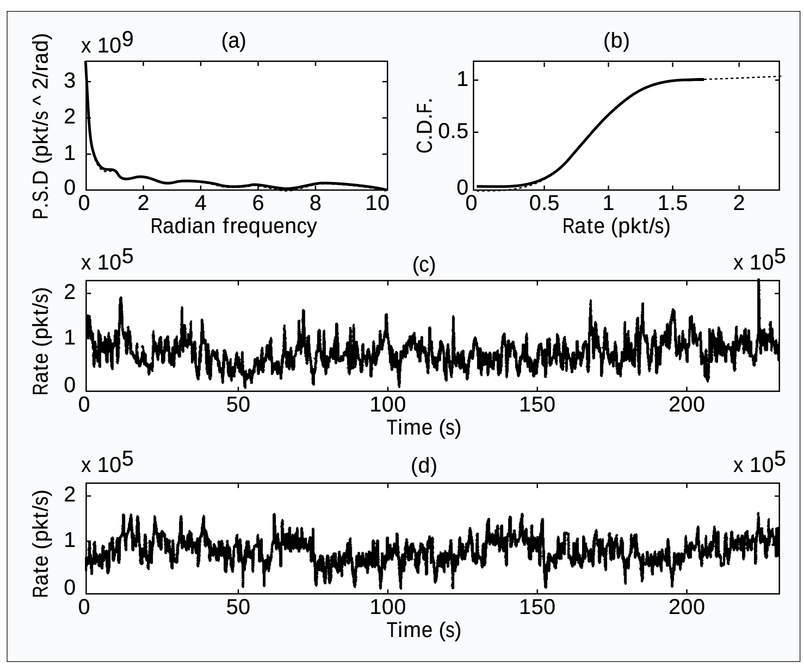

Figure 5. *Comparison of TCP trace (dotted line) with matched CMPP (solid line); a) power spectrum; b) rate distribution; c) original filtered trace; d) CMPP sequence.*

discard low-priority packets during congestion. We consider a traffic source with M loss priority streams. Let X_m be the overall fraction of the traffic with priority classes greater than m. Under the priority control scheme, no arriving packets with priority classes $\leq m$ will be lost at any time t unless the instantaneous loss rate is greater than X_m. That is, a smaller m is defined for a higher priority class. Ideally, we expect the loss rate distribution function to be a piecewise step function shown in Fig. 8 in which all the segments are divided at $x = X_m$. Without congestion control, we expect a smooth loss rate distribution curve, shown by a solid line (Fig. 8). The loss rate distribution curve can be tuned by properly designing buffer thresholds for each priority class. The loss rate distribution of this scheme, as we mentioned above, forms a piecewise step function. Note that we have assumed the service process is first-in first-out.

One effective congestion control mechanism proposed for the Internet is early packet discard (EPD). During local congestion, traffic of certain types can be selectively discarded based on different queue thresholds. Similarly, in ATM networks a cell loss priority (CLP) bit in each cell may be used to discard cells once certain queue thresholds are exceeded.

In a numerical example we studied in [7], an input rate process constructed by 50 i.i.d. two-state Markov chains has three priority classes assigned 60, 20, and 20 percent, respectively. We load the queue, which has buffer capacity 200 at $\rho = 0.85$. Our QoS criterion sets the loss rates for the three classes as $(L_0, L_1, L_2) \leq (1.0 \times 10^{-12}, 1.0 \times 10^{-6}, 5.0 \times 10^{-2})$. When no control is applied, we get $L_0 = L_1 = L_2 = 8.4 \times 10^{-4}$. If we design a one-level control system at $K_2 = 40$, such that all packet arrivals of class 2 will be discarded once the queue size exceeds K_2, we obtain $(L_0, L_1, L_2) = (4.2 \times 10^{-8}, 4.2 \times 10^{-8}, 3.2 \times 10^{-2})$. Although the overall average loss rate is increased from 8.4×10^{-4} to 6.4×10^{-3}, the QoS criteria for L_1 and L_2 are satisfied. To further reduce L_0, we design a second-level control at $K_1 = 180$, such that all packet arrivals of class 1 will be dropped once the queue size exceeds K_1. With the two-level control system we can achieve $(L_0, L_1, L_2) = (1.3 \times 10^{-13}, 7.8 \times 10^{-7}, 3.2 \times 10^{-2})$, which satisfies the QoS criteria. The results are summarized in Table 1 and plotted in Fig. 9. The queuing solutions obtained by the Queuing Solutions Module for the corresponding control levels are shown in Fig. 10.

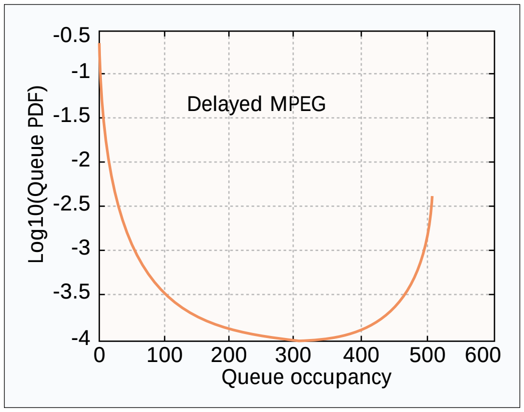

Figure 6. *The queuing solution of MPEG traffic smoothed by the delaying scheme.*

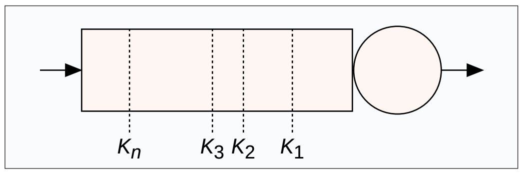

Figure 7. *Designing buffer thresholds for multiple levels of control.*

STATISTICAL MULTIPLEXING, BUFFER SHARING, AND LINK CAPACITY ALLOCATION

Economies of scale suggest that the average cost per customer decreases as the number of customers using a facility increases. By integrating services such as data, telephone, and multimedia, the broadband ISDN (B-ISDN) has the potential to share the same network resources to accommodate different kinds of services, thus increasing the number of network users and decreasing the cost of using the network per user. Therefore, it is very important to evaluate the sharing strategies in networking environments. The most important sharing strategies in networking are statistical multiplexing and buffer sharing. The fact that these schemes are still not well-understood in broadband networks makes our statistical matching tool promising in the network performance evaluation area.

The SMAQ tool overcomes many of the current limitations in evaluating statistical multiplexing and buffer sharing. For example, previous analytical approaches use simple source models, which fail to capture the diverse characteristics of real traffic sources [8]. On the other hand, the SMAQ tool uses the CMPP model to construct a tractable analytical source model to match the power spectrum and rate distribution of a given traffic trace, as described in Part I.

In order to describe how the SMAQ tool can be used to evaluate different design trade-offs in statistical multiplexing and buffer sharing, we use our source library to investigate

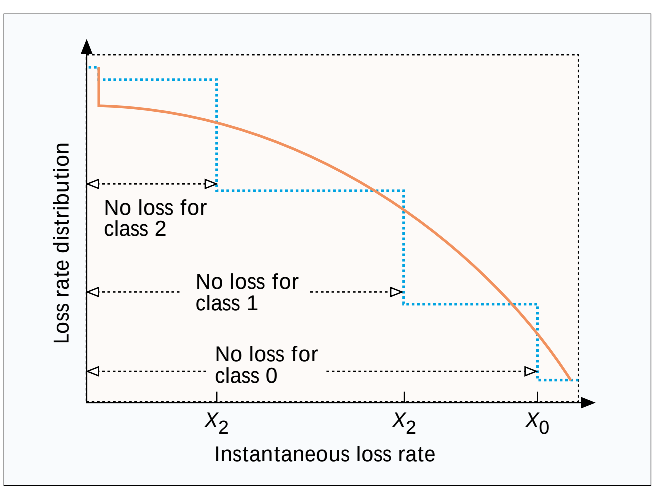

Figure 8. *Loss rate distribution with and without control.*

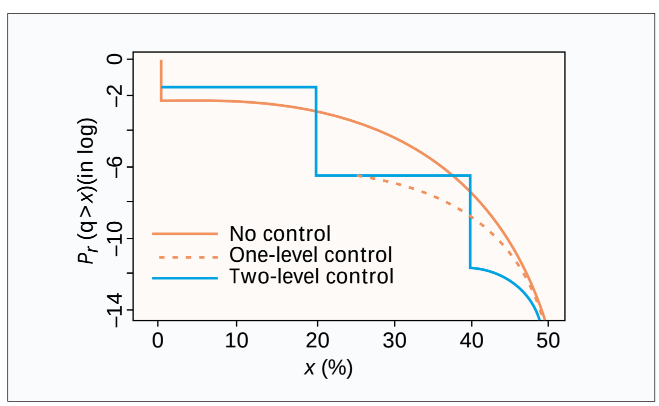

Figure 9. *An example of loss rate distribution with and without control.*

three representative traffic traces. The traces we select are JPEG, MPEG, and Ethernet traces. As mentioned before, we can also aggregate these sources and investigate the effects of aggregated network traffic. We then use the SMAQ tool to statistically match and obtain CMPPs for the sources above. Once the matched CMPP is constructed, queuing performance can be evaluated by using the Queuing Solutions Module of the SMAQ tool.

To check the validity of our single-source CMPP models, we simulate a finite-buffer FCFS system using the three original sample traces as input. We consider different utilization factors ρ with different buffer sizes K. For example, Fig. 11 shows the comparison between the simulation results and the CMPP-based analytical solutions for the MPEG trace [9]. We see that the mean queue length $\bar{q}$, the standard deviation of queue size σ_q, and cell loss probability L derived from the CMPP model agree well with those obtained from the simulations. Similar results can be found in [9] for the JPEG and Ethernet traces. One can further verify the accuracy of constructing a single CMPP to model the aggregation of N i.i.d. sources. The corresponding results for all three traces are tabulated in Table 2 with N = 5 and 25. We find that the analytical solutions for the mean queue length $\bar{q}$ and standard deviation of the queue size σ_q are very close to the simulation results within the 95 percent confidence intervals. Any discrepancies may be attributed to matching errors during CMPP construction, inability to capture higher-order statistics, and the fact that the results for the CMPP model are steady-state solutions while the simulation results are based on sample traces of limited length in which the steady state may not have been reached.

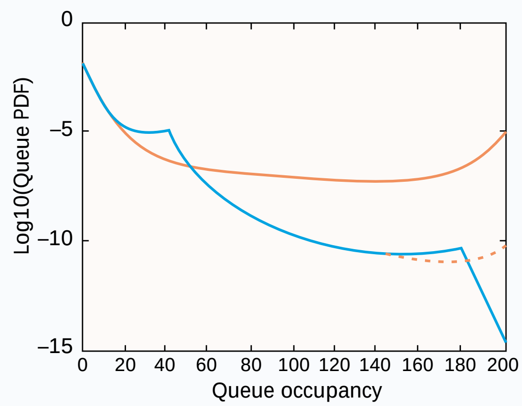

Figure 10. *Queuing solutions for no control (solid red line), one-level control (dashed red line), and two-level control (blue line).*

We also evaluate the multiplexing performance of the various sources from their CMPP models using the Queuing Solutions Module [9]. First, in Fig. 12, we plot the maximum throughput, ρ_{max}, of a single source as a function of buffer size K with the loss constraint L. We see that the throughput of the voice traffic increases rapidly with an initial increase in buffer size. However, for the two video streams (MPEG and JPEG) the throughput remains below 60 percent even with large buffer sizes for L less than or equal to 10^{-6} due to the long-range dependence (LRD) in the video sources. Increase in buffer size, therefore, has relatively little effect. The throughput for the traditional two-state Markov chain video model increases at a much faster rate than that of the CMPP-modeled video streams. However, this

	Priority class			Overall
	0	1	2	
	L_0	L_1	L_2	L
Required	1.0e^{-12}	1.0e^{-6}	5.0e^{-2}	
No control	8.4e^{-4}	8.4e^{-4}	8.4e^{-4}	8.4e^{-4}
One-level control K_2 = 40	4.2e^{-8}	4.2e^{-8}	3.2e^{-2}	6.4e^{-3}
Two-level control K_2 = 40, K_1 = 180	1.3e^{-13}	7.8e^{-7}	3.2e^{-2}	6.4e^{-3}

Table 1. *The effect of priority control on average loss rate (ξ_i is the percentage of* i*th priority in the overall input).*

	K	N	$\bar{q}(x, y \pm \delta y)^1$	$\sigma_q(x, y \pm \delta y)$	L^2(target at 10^{-3})
JPEG	1/4k	5	(10.20, 10.03 ± 0.99)	(34.00, 33.03 ± 2.48)	1.14 10^{-3} ± 2.07 x 10^{-4}
	1k	5	(33.38, 33.62 ± 5.07)	(153.0, 150.6 ± 12.85)	1.44 x 10^{-3} ± 3.21 x 10^{-4}
	16k	5	(737.2, 616.5 ± 242.1)	(2801, 2481 ± 608.1)	1.13 x 10^{-3} ± 7.24 x 10^{-4}
	1k	25	(52.97, 54.68 ±6.63)	(182.2, 183.8 ± 14.02)	1.02 x 10^{-3} ± 2.47 x 10^{-4}
MPEG	1/4k	5	(8.68, 7.72 ± 0.96)	(31.42, 28.18 ± 3.11)	9.97 x 10^{-4} ± 2.84 x 10–4
	1k	5	(34.04, 30.62 ± 7.47)	(142.2, 136.2 + 20.55)	1.40 x 10^{-3} ± 6.06 x 10^{-4}
	16k	5	(1056, 740.8 ± 145.2)	(3145, 2542 ± 404.2)	1.02 x 10^{-3} ± 8.95 x 10^{-4}
	1k	25	(47.02, 43.83 ± 4.42)	(167.7, 159.0 ± 9.76)	9.19 x 10^{-4} ± 1.40 x 10^{-4}
Ethernet	1/4k	5	(3.70, 3.61 ± 0.19)	(19.47, 18.31 ± 0.69)	7.45 x 10^{-4} ± 8.61 x 10^{-5}
	1k	5	(27.81, 27.04 ± 1.41)	(104.5, 99.81 ± 3.65)	6.80 x 10^{-4} ± 7.81 x 10–5
	16k	5	(1666, 1675 ± 150.6)	(3171, 3367 ± 213.9)	2.38 x 10^{-3} ± 6.22 x 10^{-4}
	1k	25	(30.47, 30.17 ± 1.02)	(121.6, 114.3 ± 2.57)	5.88 x 10^{-4} ± 4.06 x 10^{-5}

Notes

[1] x: analytical solution for aggregated CMPP; y: simulation result for N signle-source CMPPs; δ_y: 95% C.I.

[2] Simulation results of L for N single-source-CMPPs. L for the aggregated CMPP is analytically set to 10^{-3}.

Table 2. *A comparison of aggregated and individual CMPP modeling.*

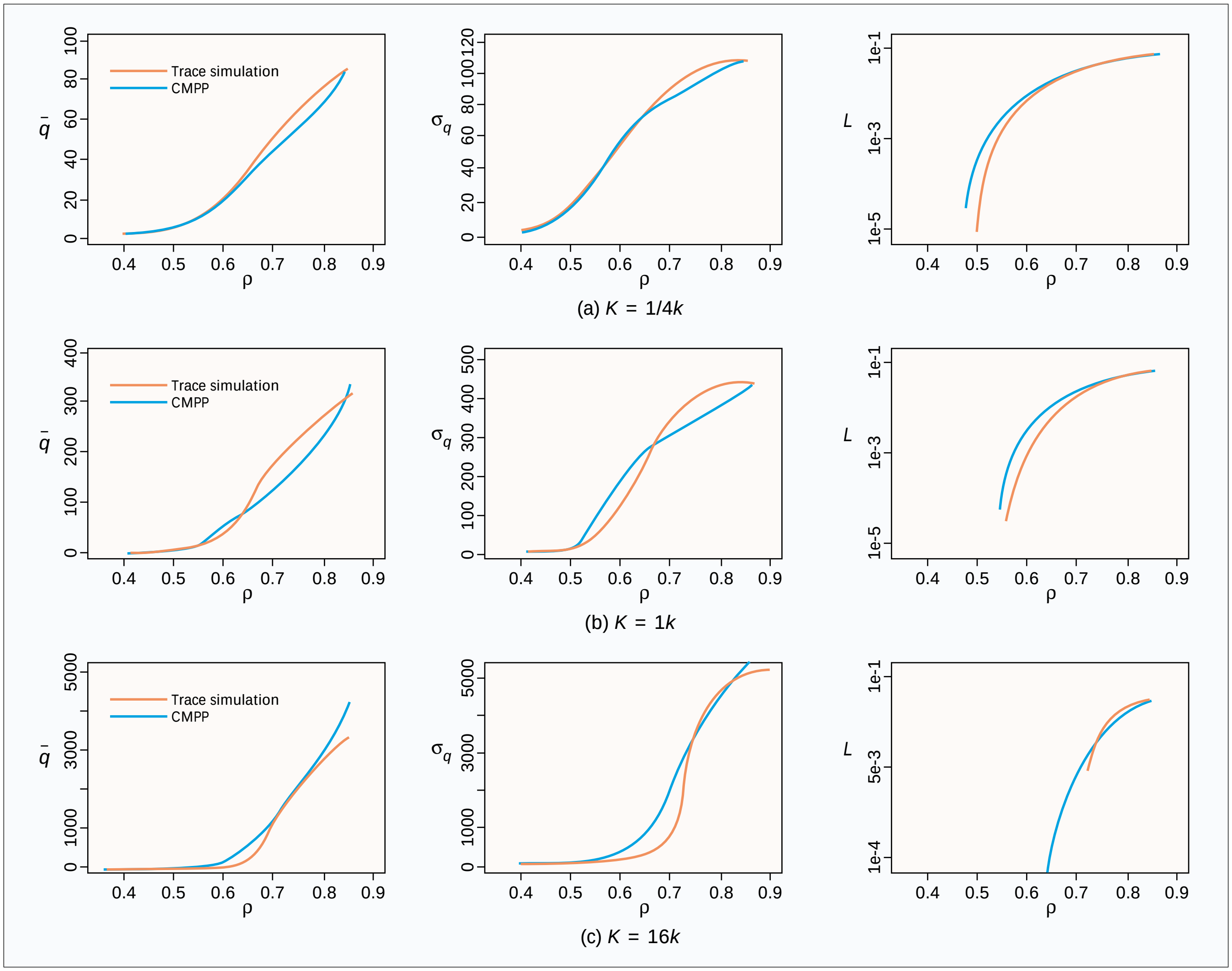

Figure 11. *Performance comparison between the MPEG-coded video trace and its CMPP model.*

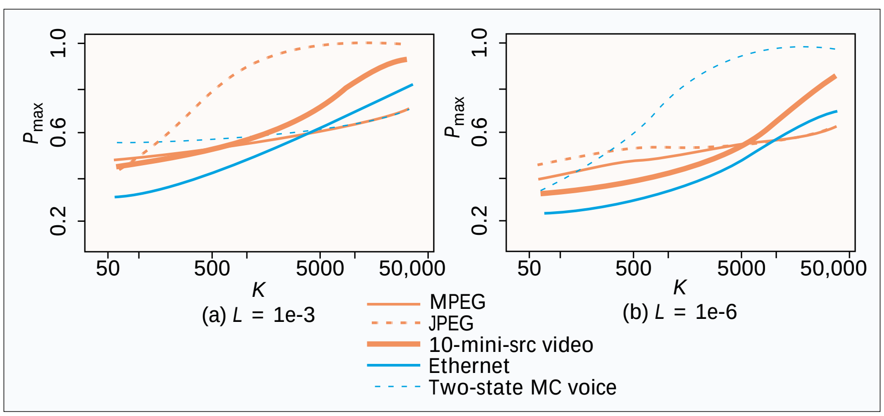

Figure 12. *Maximum throughput for* N = *1.*

model may not be able to characterize the LRD of a video source. For the Ethernet traffic, the throughput increases with increasing buffer size.

In Fig. 13 we analyze ρ_{max} for multiplexing N sources with different buffer sizes K. We observe that throughput increases considerably with increasing N. We also observe that throughput gain becomes less significant with increasing buffer size. An exception to this is the voice traffic in which relatively little throughput improvement is achieved as N increases, since its ρ_{max} is already close to 100 percent.

Finally, we define the multiplexing gain G, the ratio of the sum of the bandwidths required to transmit N individual homogeneous sources to the bandwidth required for the aggregated stream of the N sources under a given loss probability with a fixed buffer size. In Fig. 14 we plot G achieved in multiplexing $N = 25$ sources as a function of different buffer sizes. The multiplexing gain is most significant for the Ethernet traffic. The voice traffic achieves the least gain from multiplexing. For the MPEG and JPEG video sources, bandwidth savings reduce from about 60~80 percent to about 40 percent as K increases. In general, the gain G decreases as buffer size K increases.

The effectiveness of statistical multiplexing, buffer sharing and link capacity allocation is unquestionable in B-ISDN. It is very important to evaluate the trade-offs between these schemes to meet the QoS requirements efficiently. The SMAQ tool provides the network engineers with a powerful tool to evaluate different design considerations, especially multiplexing, buffer sharing, and link capacity allocation performances, as demonstrated in the examples above.

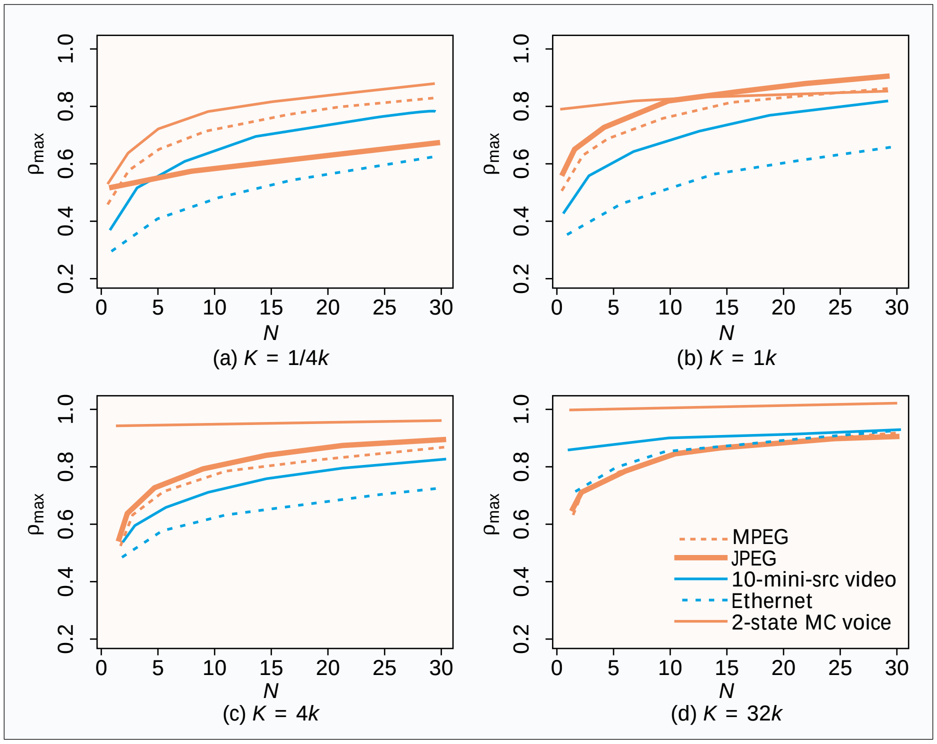

Figure 13. *The effect of* N *on throughput* (L = 10^{-6}).

WIRELESS NETWORKS

So far, in all examples we have assumed a constant service rate in the transport of correlated traffic. However, the SMAQ tool is capable of building Markov chain models for wireless channels to match the first- and second-order statistics within a measurement window. The fact that the service rate of a wireless channel is time-varying, as opposed to a wired network such as ATM, presents no difficulty to the SMAQ tool, since the tool is measurement-based.

Previous work in wireless networks has focused on the stochastic modeling of channel dynamics at the physical layer, measured by received signal strength or bit error rate. Such models cannot be directly used to evaluate higher-layer network performance, such as packet queuing delay or loss. In addition, although some work exists on packet-level channel modeling, the modeling is restricted to the use of two-state Markov chains that have very limited statistical properties.

Developing a generic stochastic model to capture the wireless channel dynamics is highly unlikely, since the dynamics are time-varying, frequency selective, and highly dependent on many other system factors such as noise, distance, mobile speed, multipath interference, power control, and coding. Therefore, the modeling technique must be measurement-based in order to capture real channel statistics under various conditions.

By using the SMAQ tool, one can investigate the queuing response to channel dynamics under various conditions, especially with respect to different driving patterns, fixed/adaptive channel coding schemes, and cell sizes, all of which are beyond the scope of this article [10, 11].

Here, we present the results of analyzing the queuing response to the single fast-/slow-fading channel dynamics. Fast fading refers to the fluctuation of radio signal strength as a result of the interference among multiple versions of the transmitted signal arriving at a receiver, whereas slow fading refers to the attenuation of the signal by irregular terrains. We assume that our arrival process consists of M = 4 i.i.d. sources, each of which may represent a virtual connection on the channel. Each source is modeled by a two-state Markov chain. The single-channel packet error rate (PER) traces are generated from the fast/slow fading energy statistics collected in the downtown Austin area produced by a mobile agent with a random driving pattern. The speed of the agent is about 30 mph, the generated packet size is 480 bits, and the signal-to-noise (SNR) level is 20 dB. The channel speed is fixed at 0.48 Mb/s. We choose a forward error correction scheme with adaptive coding set at m = 20 (the maximum number of error bits that can be corrected per packet).

In Fig. 15a and b, the packet success probability (1-PER) trace generated from the original channel sequence is compared with that generated by the Markov model formed by the SMAQ tool. We see that the two (1-PER) traces are statistically similar.

In the two-state Markov chain modeling of each source, its peak rate is set at the channel speed (0.48 Mb/s), the average burst size is 9 packets and the average rate is equal to 0.108 Mb/s. Hence, the channel utilization is ρ = 0.9. Figure 15c compares the channel power spectral density (PSD) with the arrival PSD. Since the channel PSD is dominant over the arrival PSD especially in the low-frequency domain, one can simply ignore the arrival traffic dynamics and replace the two-state Markov chains by a simple Poisson Process in the arrival modeling. In other words, only the channel statistics are to be matched by the SMAQ tool in the analytic modeling.

In Fig. 15d, we plot the queuing solution obtained by using the SMAQ tool's Queuing Solutions Module with buffer size K = 1024 and utilization ρ = 0.9. Finally, in Fig. 15e, we plot the average queue length and standard deviation of the queue length as a function of load of the mobile agent. In both cases, we see that the analytical solutions obtained by the SMAQ tool's model agree well with the solutions obtained by the simulation.

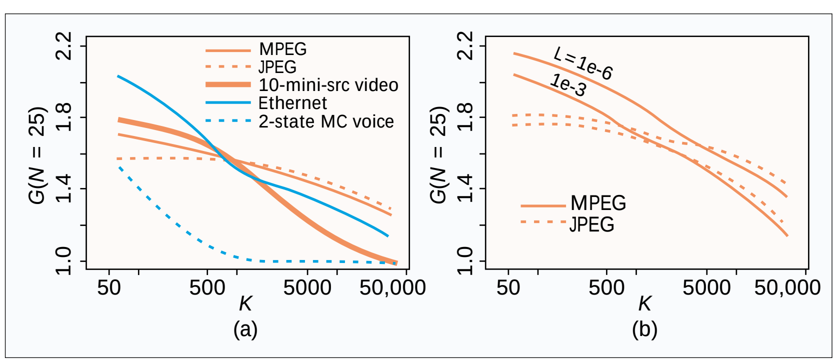

Figure 14. *a) Multiplexing gain for* N = *25* (L = 10^{-3})*; b) effect of* L *on multiplexing gain.*

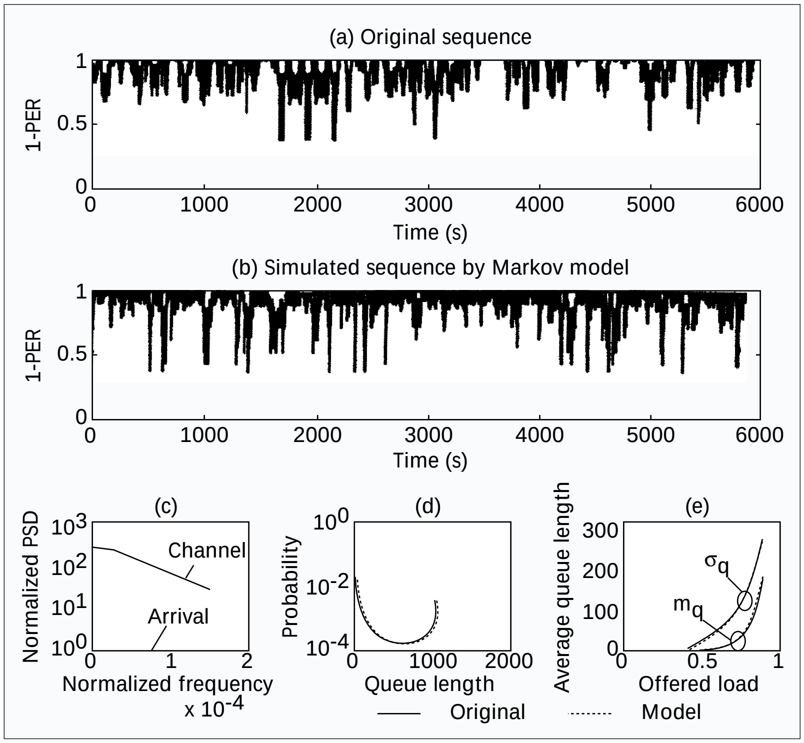

Figure 15. *a)(1-PER) of the original sequence; b) (1-PER) of the sequence produced by the Markov Model; c) arrival PSD vs. channel PSD; d) comparison of the queuing solutions of the original and the Markov model (*$K = 1024$*,* $\rho = 0.9$*); e) average queue length and queue length standard deviation as a function of offered load (*$K = 1024$*).*

The SMAQ tool provides a new direction toward the integration of wireless channel modeling and network performance analysis.

CONCLUSION

The SMAQ tool can be used in many network design areas as described throughout Part II. We believe that the tool can help system designers resolve critical design tradeoffs that they encounter during system design. Current design of the tool allows many other functionalities to be built into the system. One may refer to the electronic version of this article at http://www.comsoc.org/~ci, August 1998, for other examples, including the evaluation of priority schemes and buffer dimensioning [12], the analysis of sojourn time performance in nodal congestion [13, 14] and the design of call admission control.

ACKNOWLEDGMENT

The authors would like to thank Young Yong Kim for providing the results of Wireless Networks section.

REFERENCES

[1] S. Q. Li and C. Hwang, "On Input State Space Reduction and Buffer Noneffective Region," *Proc. IEEE INFOCOM '94*, June 1994, pp. 1018–28.
[2] S. Q. Li and C. Hwang, "On the Convergence of Traffic Measurement and Queueing Analysis: A Statistical-Match and Queueing (SMAQ) Tool,'' *IEEE/ACM Trans. Networking*, Feb. 1997, pp. 95–110.
[3] H. Che and S. Q. Li, "Fast Algorithms for Measurement-Based Traffic Modeling,'' *IEEE JSAC*, vol. 16, no. 5, June 1998, pp. 612–25.
[4] G. de Veciana and J. Walrand, "Traffic shaping for ATM networks: Asymptotic analysis and simulations,'' tech. rep., U.C. Berkeley, UCB/ERL M92/135, 1992.
[5] A. I. Elwalid and D. Mitra, "Analysis and design of rate-based congestion control of high speed networks. I: Stochastic fluid models, access regulation,'' *Queueing Sys.*, vol. 9, 1991, pp. 29–63.
[6] S. Chong and S. Q. Li, "Performance of Access Rate Control for Correlated Sources in High Speed Networks: Spectral Analysis,'' *Int'l. J. Commun. Sys.*, vol. 9, 1996, pp. 237–52.
[7] H. D. Sheng and S. Q. Li, "Spectral Analysis of Packet Loss Rate at Statistic Multiplexer for Multimedia Services,'' *IEEE/ACM Trans. Networking*, vol. 2, no. 1, Feb. 1994, pp. 53–65.
[8] D. Tse, R. G. Gallager, and J. N. Tsitsiklis, "Statistical multiplexing of multiple time-scale Markov Streams,'' *IEEE JSAC*, vol. 13, no. 6, Aug. 1995, pp. 1028–39.
[9] W. C. Lau and S. Q. Li, "Statistical Multiplexing and Buffer Sharing in Multimedia High-speed Networks: A Frequency Domain Perspective,'' *IEEE/ACM Trans. Networking*, vol. 5, no. 3, June, 1997, pp. 678–92.
[10] Y. Y. Kim and S. Q. Li, "Capturing Important Statistics of Fast/Slow Fading Channel for Network Performance Analysis,'' submitted to a special issue of wireless modeling in *IEEE JSAC*.
[11] Y. Y. Kim and S. Q. Li, "Modeling Fast Fading Channel Dynamics for Packet Data Performance Analysis,'' *Proc. IEEE INFOCOM '98*, Mar. 1998.
[12] C. Fulton, S. Q. Li, A. Lin, "Measurement-Based Performance Evaluation of an ATM Switch with External Multi-casting Engine and Multiple Priority Classes,'' *IEEE JSAC*, vol. 15, no. 5, June 1997.
[13] L. A. Kulkarni and S. Q. Li, "Transient Behavior of Queueing Systems with Correlated Traffic,'' *Perf. Eval. J.*, vol. 27 & 28, Oct. 1996, pp. 117–46.
[14] W. C. Lau and S. Q. Li, "Sojourn-time Analysis of Nodal Congestion in Broadband Networks and its Impact on QoS Specifications,'' *Proc. IEEE INFOCOM '96*, Apr. 1996, pp. 1327–38.

BIOGRAPHIES

SAN-QI LI (sanqi@globe.ece.utexas.edu) received a B.S. degree from the Beijing University of Posts and Telecommunications, China, in 1976, and M.A.Sc. and Ph. D. degrees from the University of Waterloo, Ontario, Canada, in 1982 and 1985, respectively, all in electrical engineering. He is presently a professor in the Department of Electrical and Computer Engineering at the University of Texas at Austin. To date, he has published more than 100 papers in international archival journals and refereed international conference proceedings. The main focus of his research has been to develop new analytical methodologies and carry out performance analysis of multimedia service networks, and, based on these methodologies, to understand system fundamentals and explore new design concepts.

SANGKYU PARK (spark@globe.ece.utexas.edu) received M.S. and B.S. degrees in control and instrumentation engineering from Seoul National University, Korea, in 1991 and 1989, respectively. He is currently working toward a Ph.D. degree at the Department of Electrical and Computer Engineering at the University of Texas, Austin, under Prof. San-qi Li's supervision. His research interests include high-speed network modeling and analysis.

DOGU ARIFLER [M] (arifler@ece.utexas.edu) graduated with a B.S.E.E. degree in electrical and computer engineering from the University of Texas at Austin in 1997, where he also received the A. A. Dougal Academic Excellence Award in electrical and computer engineering. He returned to the University of Texas to work on his M.S. degree under Prof. San-qi Li's supervision in 1997. His current research interests are network performance evaluation and Web-based software development. He is a member of Sigma Xi.