Measurement and Simulation Based Effective Bandwidth Estimation*

Nelson X. Liu and John S. Baras
Institute for Systems Research
University of Maryland
College Park, MD 209742, USA
{nliu, baras}@isr.umd.edu

***Abstract*— Effective bandwidth represents the resource demand for network traffic to achieve its QoS goal. However, the estimation of the effective bandwidth for modern Internet traffic is difficult and inaccurate so far. In this paper we present a novel scheme using the measurement and simulation techniques to estimate the effective bandwidth. The advantages of the scheme lie in its accuracy, its applicability to various traffic environments, and its efficiency. We describe the simulation procedure, the efficient search algorithm to quickly lock the effective bandwidth, the sample size determination algorithm, and the performance judgment policy in the simulation to guarantee the estimation accuracy. We also examine the issue of estimating the effective bandwidth of aggregate traffic from sub-aggregates, and provide an empirical formula to achieve very good accuracy.**

***Key Words*— Effective bandwidth, Estimation, Measurement and simulation**

I. INTRODUCTION

Effective bandwidth is the exact amount of bandwidth a traffic stream needs to satisfy given QoS requirement. Directly measuring the resource demand for QoS fulfillment, it is a very useful tool in broad areas of QoS modeling, control, dimensioning, and management [1] [3] [5] [7] [8] [9] [15]. However, analysis of effective bandwidth is very difficult for general traffic processes, if not impossible. Paper [10] gives a formal definition and presents results for many traffic models especially Markovian processes. No exact solution exists for the long-range dependent process. So much work turns to large deviation to for asymptotic results, such as [6] [7] [11]. But engineering application of such approach is quite limited. Modern Internet traffic is generally a multi-scale process with complex dependent structure [13] [14]. We lack effective tools to analyze or approximate the effective bandwidth for such process.

Recent develop of measurement and simulation technologies provide new opportunities to seek the solution [2] [4] [12]. In this paper we present a measurement and simulation-based (MSB) approach for effective bandwidth estimation. It is intended to be a general approach to suite various traffic environments. Our focus is to establish conditions for applying the method, ensuring the sufficiency of the measurement and simulation and thus the accuracy of the estimation. In addition to the basic mechanisms, we will also consider the problem of estimating the effective bandwidth of aggregate traffic from the sub-aggregates, which has wide applications.

The paper is organized as follows. In Section II, we present key algorithms of the MSB effective bandwidth estimation, including the MSB procedure, the effective bandwidth search, the statistical inference for performance judgment, and the sample size determination. In Section III, we explore the accuracy of the estimation from sub-aggregates, and give an empirical formula to improve the accuracy. Simulation results are included in Sections II and III. Section IV concludes the paper.

II. ESTIMATION FOR AGGREGATE TRAFFIC

A. Procedure

The idea is as follows. We capture online a sample of the traffic (a section of traffic trace) for which the effective bandwidth will be estimated, and input the trace into a simulator attached to the point of observation (Poo), which is usually a link. The simulator simulates a single-server FIFO queue serving the traffic. The simulation is controlled by a search algorithm to decide the effective bandwidth satisfying certain QoS requirement. In this paper, the QoS requirement is specified with the delay bound and the loss probability in the form

$$P[d > D] \le e \tag{2.1}$$

which states that the probability that the packet delay d is beyond a bound D should not be greater than e.

The simulation procedure is as follows. It is a loop involving several algorithms.

Simulation Procedure:
1. Set a flag $fg = 0$.
2. Initialize $c = W_0$.
3. Run algorithm **Sample** to initialize k.
4. Run basic version of the queue simulation algorithm **QSim_v1** with parameters c and k.
5. Do hypothesis test: H: $p = e$ against H^+: $p > e$ and H^-: $p < e$.
6. If H holds,
    a. $fg = fg + 1$.
    b. If $fg < 2$, go to step 8.
    c. If $fg == 2$, stop.

* This work is supported by DARPA under contract No. N66001-00-C-8063.

7. Otherwise, run the following binary search algorithm to determine a new c:
   B$^+$, if H^+ holds;
   B$^-$, if H^- holds.
   Go to step 4.
8. Run dynamic version of algorithm **QSim_v2** with multi-scale sample size inflation to update k.
9. Go to step 5.

Here are meanings of some parameters: c represents the queue service rate, the final value of which is the effective bandwidth and W_0 is the initial value, k is the base 2 logarithm of the sample size, p denotes $P[d > D]$. In above procedure, step 1 and 2 initialize a control flag fg and the queue service rate c. The algorithm Sample in step 3 decides an initial sample size 2^k that controls the length of input traffic. The algorithm Qsim_v1 in step 4 simulates the queue with the initial sample size. When the simulation terminates, step 5 judges whether the performance requirement is met or not by doing a hypothesis test. If the requirement is met, the procedure will go to step 8 and does the simulation again with an "inflated" sample size, which is done in the algorithm Qsim_v2. This re-checking is to guarantee the sufficiency of the simulation, thus the accuracy of the effective bandwidth estimation, for *dependent* traffic. If the requirement is not met, the search algorithm B$^+$ or B$^-$ will choose a new c and the simulation is redone. The procedure stops when a service rate passes the re-checking. Different phases of the overall process are marked by fg.

Several important technical issues are involved in the design of above algorithms: One is that the search algorithm should be able to quickly lock the effective bandwidth. A related problem is the choice of the initial value of c to improve the search efficiency. These issues will be treated in sub-Section II-B. Another issue is the sample size of traffic for simulation. It is inefficient, unnecessary, and even improper to do simulations for an overly long trace. For example, it may cause the so-called "overestimation" problem. A good sample size is the minimum number that is sufficient for evaluating the performance metric at required significance level. A problem related with this is how to decide whether the performance requirement is met or not in the simulation. We will address these two problems in sub-Sections II-D and II-C.

As to the simulator itself, it is a very simple and efficient algorithm using the packet information (arrival time and packet size) and the queue information (queue service rate) to calculate the queueing delay of each packet. It does not actually process real, physical packets. Just like any single-queue simulation, only computations involved are addition, subtraction, and comparison. There are two versions of the queue simulation algorithm. The basic version, QSim_v1, is a pure FIFO queue simulator. There are numerous references about how to do a good implementation of queue simulation. The dynamic version, QSim_v2, involves in the queue simulation an algorithm to dynamically update the sample size based on the simulation result, which will be described in sub-Section II-D.

B. Search Algorithm

The search algorithm is to change the queue service rate in the simulation and eventually find the effective bandwidth W_e. Simple search algorithms such as Newton method and binary search algorithm may be used. One important thing to apply Newton method is to decide the initial value. Binary search is known for its simplicity and time efficiency. Standard binary search algorithm needs some conditions to be satisfied: the search region should be bounded and the search set should be sorted.

These problems can be nicely solved based on effective bandwidth theory. It is known that W_e is between the average rate R_1 and the peak rate R_2 of the traffic. Thus, to use Newton method, the initial value W_0 can be set as $W_0 = (R_1 + R_2)/2$, which should not be far from W_e. For binary search, since the loss rate is a monotonic function of the queue service rate c, the searched set, i.e., the set of service rate values, is actually automatically sorted. Given R_1 and R_2, the search region is bounded. So all prerequisite for binary search are satisfied.

It is easy to measure R_1 and R_2 of each path at a Poo. Then the binary search algorithm, which is separated into two parts for description purpose, is as follows.

Algorithm: B$^+$
Input: R_1, R_2
Output: c
1. $R_2 = c_{old}$
2. $c = (R_1 + R_2)/2$.
3. $c_{old} = c$

Algorithm: B$^-$
Input: R_1, R_2
Output: c
1. $R_1 = c_{old}$
2. $c = (R_1 + R_2)/2$.
3. $c_{old} = c$

c_{old} is a global variable, and is initialized as $c_{old} = W_0$. The initial value for binary search is also $W_0 = (R_1 + R_2)/2$, which is part of the nature of binary search. As loss rate is a monotonic function of c, the convergence of the binary search is guaranteed.

C. Performance Judgment

After each run of the simulation, we need to decide whether the simulated performance meets the performance requirement or not. Since the performance is statistical in nature, we use hypothesis test to make the decision. The performance metric is $p = P[d > D]$. As indicated in step 5 in the simulation procedure, the hypothesis test is

$$H: p = e \text{ against } H^+: p > e \text{ and } H^-: p < e. \quad (2.2)$$

This is a two-tailed hypothesis test. The sample size significantly affects the accuracy of the hypothesis test. We have a procedure to decide and update the sample size in the simulation, which will be described in next sub-Section. Given the sample size n, the accuracy also depends on the distribution of the traffic. We use the distribution-free or nonparametric approach to suit general traffic. A convenient method for our application is the binomial test. Let G be the set of delay measurements from the simulation. Let $m = \#\{s: s > D, s \in G\}$. If $p = e$ and is very small, it can be shown that $m \sim N(\mu_m, \sigma_m^2)$ with mean and variance being $\mu_m = ne$ and $\sigma_m^2 = e(1-e)$.

We define the test statistic as

$$m^* = \frac{m - \mu_m}{\sigma_m / \sqrt{n}}$$

It is easy to see $m^* \sim N(0, 1)$. To do a two-tailed hypothesis test, define the significance level as

$$\beta = P[x > |m^*| \mid H]$$

Given a significance level value $\beta = B$, we can eventually write down the test rule as follows.

Accept H	and	reject H^+ and H^-,	if $\lvert m^*\rvert < z_{1-B}$
Accept H^+	and	reject H and H^-,	if $m^* > z_{1-B}$
Accept H^-	and	reject H and H^+,	if $m^* < -z_{1-B}$

Here z_{1-B} is the 100(1-B)-th percentile of $N(0, 1)$.

D. Sample Size

The number of packets captured for simulations is an important parameter. It should be large enough to guarantee the accuracy of the estimation of the performance metric. However, larger sample size means higher overhead in terms of bandwidth and computing power consumption. Also, too large sample size may lead to the problem of "overestimation". So the sample size should be kept in a reasonable range.

Formally, the question of determination of the sample size in our context is as follows. Assume n packets are fed into a simulator, and m of them are measured to have delays greater than D. Given two small quantities ε and α, how big should n be so that m/n can be used as an estimator of p and the 100(1-α)% confidence interval is $m/n \pm \varepsilon$? If all delay measurements are independent, there is a well-established result for this question. However, if the measurements are dependent, which is obviously the case, the theoretical result may underestimate the sample size and leads to inaccurate estimation of p. Our estimation aims at adapting to a variety of traffic behaviors that may have different dependent structures. In addition, it prefers to make as few assumptions as possible about the distribution of the measurements, i.e., a distribution free estimator. There is so far no systematic treatment of these concerns. Here we will use the following approach to achieve above goals. First, a basic sample size is deduced based on the independence assumption. Then, a heuristic "inflation" algorithm is employed to find a proper sample size using the basic sample size as the initial value. The first part is done statically in step 3 of the simulation procedure, and the second part is done dynamically in step 8.

We start by assuming the delay measurements are independent. The sample size can be determined as follows. Denote measurement i by d_i. Obviously, either $d_i > D$ or $d_i \leqslant D$ holds. If there are m out of n measurements being greater than D, m has a binomial distribution with parameter p. Denote m/n by $\hat{p}$. Because $E[\hat{p}] = p$, $\hat{p}$ is an unbiased estimator of p. Let σ be the standard error of $\hat{p}$. It can be shown that the 100(1-α)% confidence interval of the estimator is $\hat{p} \pm z_{1-\alpha}\sigma$. Here $z_{1-\alpha}$ is the 100(1-α)-th percentile of the standard normal distribution, $N(0, 1)$. An unbiased estimator of σ^2 is

$$\hat{\sigma}^2 = \frac{\hat{p}(1-\hat{p})}{n}$$

Given α and ε, and p being the target estimation value, the above formula imply $n \geq z_{1-\alpha}^2 p(1-p)/\varepsilon^2$.

Denote the initial sample size by N_0. We choose N_0 to be a power of 2. Let $K = \lceil \log_2 z_{1-\alpha}^2 p(1-p)/\varepsilon^2 \rceil$. Then $N_0 = 2^K$. The algorithm Sample carries out above calculation of K. Then simulations run with N_0 until an effective bandwidth is found (the loop of step 3 to step 7 in the simulation procedure). This is the first phase of simulation (fg = 0). The effective bandwidth value is the final result if the traffic is Poisson. For general traffic, it provides a coarse estimation because N_0 may be insufficient. The multi-scale inflation algorithm to dynamically search the final sample size will be started. It is embedded in the dynamic version of the queue simulator QSim_v2, which is shown below. This is the second phase of the simulation (fg = 1). To control the termination, we select a factor $1 < \theta < 100$, and use it to limit the maximum of the sample size. θ's value is based on the overhead of the simulation that the user is willing to accept.

Algorithm: QSim_v2
Input: traffic samples, k
Output: k

1. Initialize $k = K$ and $n_{1/2} = 2^k$.
2. Run QSim_v1 with $n_{1/2}$ packets and get $n_{1/2}$ delay measurements. Denote the measurements by G_1. Let $m_1 = \#\{s: s > D, s \in G_1\}$, and $\hat{p}_1 = m_1/2^K$.
3. Continue to run QSim_v1 with next $n_{1/2}$ packets, and denote the delay measurements by G_2. Let $m_2 = \#\{s: s > D, s \in G_2\}$, and $\hat{p}_2 = m_2/2^K$.
4. Do hypothesis test: $H1$: $\hat{p}_1 = \hat{p}_2$ against $H2$: $\hat{p}_1 \neq \hat{p}_2$.
5. If $H1$ is rejected and $n = 2n_{1/2} < \theta \cdot 2^K$, then
   i. let $k = k + 1$.
   ii. let $G_1 = G_1 \cup G_2$.
   iii. re-calculate m_2, $\hat{p}_2$, and $n_{1/2}$ using new k.
   iv. go to step 3.
6. If instead $H1$ is accepted or if $n \geq \theta \cdot 2^K$, stop.

Basically, above procedure starts from the initial sample size. It checks whether the sufficiency is reached through a hypothesis test, which essentially checks whether the estimation is consistent in two successive samples. The techniques for the hypothesis test are similar to those in Section II-C, and will be omitted here. The sample size will be extended exponentially until it passes the sufficiency checking or reaches the maximum.

If this procedure gives a new sample size, the simulation and the effective bandwidth search will continue until a final value of the effective bandwidth is found, which forms the third phase of the simulation procedure (though still fg = 1). The loop from step 4 to step 7 is re-used for this phase.

E. Simulation Results

In the following simulation we determine the effective bandwidth of the aggregate traffic on an OC-12 (622Mbps) optical link. The traffic is generated to have a complex dependent structure that is a combination of the long-range dependence and the short-range dependence. It is an aggregate of 80 on-off flows with the average rate of 3Mbps per flow. Half of the flows have heavy-tailed on and off periods, and half are short-tailed on-off processes. Traffic like this is more "real" and but has no theoretical result for the effective bandwidth.

We choose different values of e between 0 and 0.1 and different values of D between 0 and 2.56 ms, and determine the effective bandwidths for various QoS requirements with above MSB scheme. Figure 1 shows the results. Axes D and e in the graph are in logarithmic scales. At larger e, e.g., e = 0.1, the effective bandwidth decreases linearly with the increase of D, and the change is quite conservative. For smaller e, the relation tends to be rather non-linear, and the falling distance enlarges. The effects of changing e are very different at different D. When D is large, say D = 2.56 ms, the bandwidth change is not impressive. For small D, e.g., D = 0.01ms, the change becomes quite visible. These changes in general conform to neither exponential law nor scaling law, as are the theoretical results for the pure short-range dependent and the pure long-range dependent traffic [6] [10]. To obtain these accurate estimations may be impossible without the MSB approach. In this experiment, all simulations pass the re-checking within three rounds of sample size inflation. So the MSB procedure is efficient.

III. Estimation From Sub-Aggregates

In traffic engineering and network dimensioning applications, it is often desirable to measure the traffic demands at the network edge and estimate the bandwidth needs of a core link based on them [2] [3] [4] [9]. So far this is mainly done with the average traffic rate. The advantage is clear if the effective bandwidth is used as the traffic demand and estimated this way, i.e., getting the path-wise effective bandwidths with the MSB scheme at network edge, and estimating the effective bandwidth W_e of a core link as

$$W'_e = \sum_{j \in J} w_j \qquad (3.1)$$

where J represents the set of all paths passing the link, and w_j is the effective bandwidth of path $j \in J$. Here we made the assumption that the effective bandwidth of the path-wise traffic changes little throughout the core network, which is reasonable since usually there are few packet losses in the core cloud and the traffic behavior propagates along the path. The question is, how accurate the estimation (3.1) is?

Amazingly, the estimation accuracy is very bad in general: the error may be unbounded. To see this, we do an evaluation by simulation with the network shown in figure 2. Nodes D, E, and H are pure intermediate routers, and all other nodes are

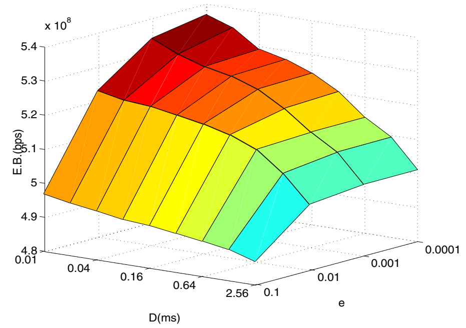

Fig.1. MSB effective bandwidths vs performance requirement for mixed traffic environment

edge nodes generating and receiving traffic. Every link is duplex with an OC-12 (622Mbps) in either direction. There are totally 30 unidirectional links. The small triangles symbolize the Poos at edge. In the simulation, every edge node sends traffic to all other edge nodes, and there are f flows from a node to another. $f > 0$ is a uniform random number between 0 and a fixed number F, rounded to the nearest integer. The flows are generated in the same way as in previous simulation. We create three traffic environments: pure short-range dependent environment, in which only short-tailed flows are generated, pure long-range dependent environment, which has only heavy-tailed flows, and the mixed environment, in which each sender generates either type of flows by 50%, and randomly distribute them among receivers. The routing information is known, so the paths through every link can be decided. We use the MSB approach to directly estimate W_e of every link at core, as well as all w_j at edge, which leads to W'_e. Then explore the relation between W_e and W'_e in the following way: use W'_e and W_e for each link as the x- and y-coordinates to draw a dot (W'_e, W_e) on a plane. Should $W_e = W'_e$, the dot would be on the 45°-angle line. The deviations of the dots from the 45°-angle line would indicate the error of the estimation.

Figure 3 shows the results for F = 80 under three traffic environments. The performance requirement is D = 0.1 ms and e = 0.001. The 95% confidence interval of all MSB estimations is $p \pm 0.0005$. From figure 3 we see generally $W_e < W'_e$, and the dots do not match the 45°-angle line (the diagonal line), but diverge from it. The higher the link load is, the greater is the divergence. Without a little thinking we understand this is due to the multiplexing gain. We notice the divergence is quite regular. In fact, it is linear to the increase of the link load.

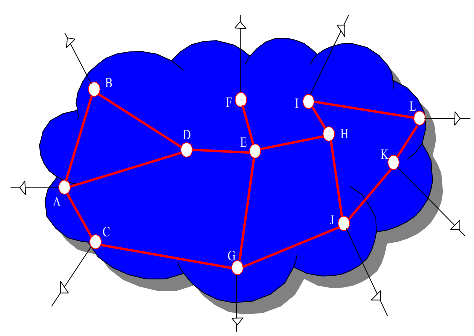

Fig.2. Network topology.

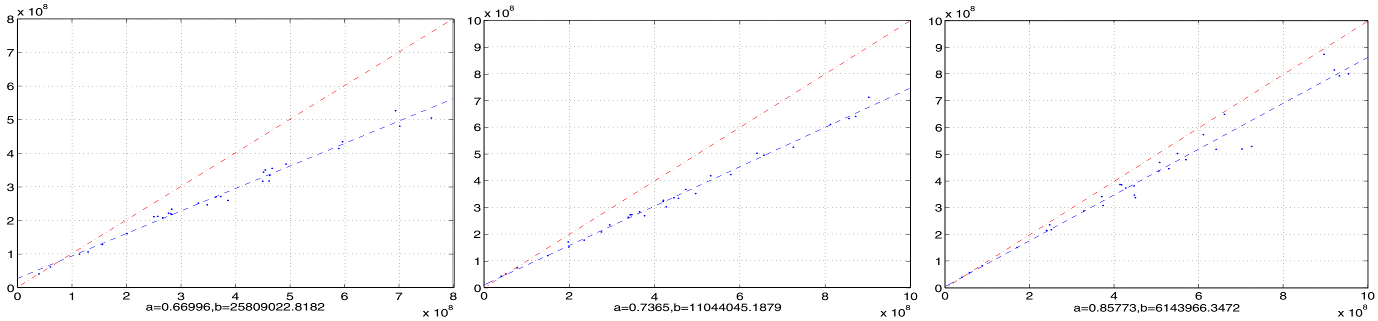

Fig.3. MSB effective bandwidth estimation accuracy under different traffic environments: mixed (left), short-range dependent (middle), and long-range dependent (right).

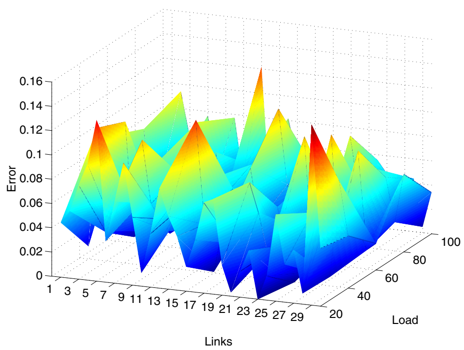

Fig.4. An error surface of MSB effective bandwidth estimation for mixed traffic environment.

Draw the least-square fitting line of the dots in each graph (the blue dashed line), of which the slope a and the bias b are marked at the bottom. The fitting shows convincingly the linearity is consistent for different traffic environments.

So we can give a systematic correction formula for estimation (3.1):

$$W_e = aW'_e + b \quad (3.2)$$

Choose different F values, which measure the load, and do more simulations. Figure 4 shows the error surface of the effective bandwidth estimation after correction for the mixed traffic environment. The error is calculated as $\delta = |(W'_e - W_e)/W_e|$. The results indicate that most errors are within 10% and all errors are below 15%. The average error for that surface is 4.5%. This level of estimation accuracy generally applies to various combinations of traffic loads and environments. Having said that, we note that in the long-range dependent environment the estimations are relatively more scattered around its fitting line. The short-range dependent traffic sees the best fitting and thus the least error, and the mixed traffic is in the middle.

I. CONCLUSIONS

In this paper we present a MSB approach for accurate estimation of the effective bandwidth. The advantages of the scheme lie in its accuracy, its applicability to various traffic environments, and its efficiency. We give the simulation procedure, the efficient search algorithm that enhances the binary search with the effective bandwidth theory, and the conditions of the sample size and the performance judgment for the simulation to guarantee the accuracy of the estimation. We also examine the issue of estimating the effective bandwidth of aggregate traffic from sub-aggregates, and provides an empirical formula to achieve good accuracy.

Because of the direct relation between QoS and the effective bandwidth, the MSB estimation provides a very useful tool in resource allocation, QoS management, and network dimensioning.

REFERENCES

[1] E. Bouillet, D. Mitre, and I. Ramakrishnan, “The Structure and Management of Service Level Agreements in Networks,” *IEEE Journal on Selected Areas in Communications*, 20(4):691-699, 2002.

[2] T. Bu, N. Duffield, F. Lo Presti and D. Towsley, "Network Tomography on General Topologies," *ACM SIGMETRICS* 2002.

[3] N.G. Dueld, P. Goyal, A. Greenberg, P. Mishra, K. K. Ramakrishnan, “A Flexible Model for Resource Management in Virtual Private Networks,” *ACM SIGCOMM* 1999.

[4] M. Coates, A. Hero, R. Nowak, and B. Yu, "Internet tomography," *IEEE Signal Processing Magazine*, 19(3):47-65, 2002.

[5] C. Courcoubetis, F. P. Kelly, V. A. Siris, and R. Weber, "A Study of Simple Usage-based Charging Schemes for Broadband Networks," *Telecommunication Systems*, 15(3-4):323-343, 2000.

[6] N.G. Duffield and N. O'Connell, "Large Deviations and Overflow Probabilities for the General Single-Server Queue with Applications," *Mathematical Proceedings of the Cambridge Philosophical Society*, 118:363-374, 1995.

[7] A. Elwalid, D. Mitra, and R. H. Wentworth, “A New Approach for Allocating Buffers and Bandwidth to Heterogeneous Regulated Traffic in an ATM Node,” *IEEE Journal on Selected Areas in Communications*, 13(6):1115--1127, 1995.

[8] T. Ferrari, W. Almesberger and J.-Y. Le Boudec "SRP: A Scalable Resource Reservation Protocol for the Internet", *Computer Communications*, 21(14):1200-1211, 1998.

[9] S. Jamin, P. Danzig, S. Shenker, and L. Zhang, “A Measurement-Based Admission Control Algorithm for Integrated Services Packet Networks,” *IEEE/ACM Transactions on Networking*, 5(1):56-70, 1997.

[10] F.P. Kelly, “Notes on Effective Bandwidths,” In *Stochastic Networks: Theory and Applications*, Editors F.P. Kelly, S. Zachary and I.B. Ziedins, Oxford University Press, 141-168, 1996

[11] G. Kesidis, J. Walrand, and C.-S. Chang, "Effective Bandwidth for Multiclass Markov Fluids and Other ATM Traffic Sources," *IEEE/ACM Transactions on Networking*, 1(4): 424-428, 1993.

[12] K. Lai and M. Baker, “Measuring Link Bandwidths Using a Deterministic Model of Packet Delay,” *ACM SIGCOMM* 2000.

[13] W. E. Leland, M. S. Taqqu, W. Willinger and D. V. Wilson, “On the self-similar nature of Ethernet traffic,” *IEEE/ACM Transactions on Networking*, 2:1-15, 1994.

[14] N. Liu and J. Baras, “Statistical Modeling and Performance Analysis of Multi-Scale Traffic,” *IEEE INFOCOM* 2003.

[15] E. van der Merwe, S. Rooney, I. Leslie, and S. Crosby, "The Tempest---a Practical Framework for Network Programmability," *IEEE Network*, 12(3):20-28, 1998.