# Comparison of Measurement-based Admission Control Algorithms for Controlled-Load Service


Sugih Jamin*
EECS Department
University of Michigan
Ann Arbor, MI 48109-2122
*jamin@eecs.umich.edu*

Scott J. Shenker
Palo Alto Research Center
Xerox Corporation
Palo Alto, CA 94304-1314
*shenker@parc.xerox.com*

Peter B. Danzig
Department of Computer Science
University of Southern California
Los Angeles, CA 90089-0781
*danzig@usc.edu*



## Abstract

*We compare the performance of four admission control algorithms—one parameter-based and three measurement-based—for controlled-load service. The parameter-based admission control ensures that the sum of reserved resources is bounded by capacity. The three measurement-based algorithms are based on measured bandwidth, acceptance region [9], and equivalent bandwidth [7]. We use simulation on several network scenarios to evaluate the link utilization and adherence to service commitment achieved by these four algorithms.*



**Keywords:** Admission Control, Quality of Service (QOS), Internet.


## 1 Introduction

The role of any admission control algorithm is to ensure that admittance of a new flow into a resource constrained network does not violate service commitments made by the network to admitted flows. The service commitments made could be quantitative (e.g., a guaranteed rate or bounded delay), or it could be more qualitative (e.g., a "low average delay"). There are two basic approaches to admission control: the first, which we call the parameter-based approach, computes the amount of network resources required to support a set of flows given *a priori* flow characteristics; the second, the measurement-based approach, relies on measurement of actual traffic load in making admission decisions. In this paper, we report on an initial comparative study of four admission control algorithms, one parameter-based, three measurement-based. The main criterion used in evaluating any admission control algorithm must be how well it fulfills its primary role of ensuring that service commitments are not violated. The simplest way to ensure complete commitment conformance is to allocate enough resources to meet the worst-case requirements of each flow. For bursty sources, however, this scheme results in low network utilization. Hence, the second evaluation criterion is how high a level of network utilization an admission control algorithm can achieve while still meeting its service commitments.[1] The third evaluation criterion is the implementation and operational costs of an algorithm. Since admission control is a session-level, not packet-level, control mechanism, we do not expect its implementation or operational cost to be a prohibitive factor and only consider the first two criteria in this study.

Parameter-based admission control algorithms can be analyzed by formal methods. Measurement-based admission control algorithms can only be analyzed through experiments on either real networks or a simulator. We strive to make the simulation environments under which we investigate the behavior of the various algorithms as comparable as possible, but this does not mean the operating conditions would not be unfairly disadvantageous to any particular algorithm. Our intention here is not to pick a winner, but to start a dialog, not only on the evaluation criteria to be used in comparing different measurement-based admission control algorithms, but also on the definition of a meaningful comparison of measurement-based admission control algorithms given the myriad tunable parameters.

Given the reliance of measurement-based admission control algorithms on source behavior that is not static in

*Part of this work was done while Sugih Jamin was at the University of Southern California, with equipment support from the NSF small-scale infrastructure grant, award number CDA-9216321 and equipment loan from Sun Microsystems Inc.



[1] Note that in an environment where there is a large fraction of best-effort traffic, the secondary goal of the admission control algorithm may simply be to control the fraction of bandwidth allotted to realtime traffic, rather than maximizing it.

general, service commitments made by such algorithms can never be absolute. Measurement-based approaches to admission control can only be used in the context of service models that do not make guaranteed commitments, such as the controlled-load service model. Controlled-load service is designed for adaptive real-time applications that can tolerate variance in packet delays. The *controlled-load* service model, as defined in reference [19], "tightly approximates the behavior visible to applications receiving best-effort service *under unloaded conditions*" over the same path. Furthermore, applications requesting controlled-load service may assume that its packet loss rate is on the order of the transmission medium's error rate and that its typical experienced delay should be on the order of the path's transmission and propagation delays. More specifically, average packet queueing delay should be no greater than the flow's "burst time" and there should be minimal loss rate averaged over time-scales larger than "burst time"—where the "burst time" is defined as the time required to serve a flow's maximum burst at the flow's reserved rate. For a flow described by a token bucket filter, the "burst time" is $b/r$, where $b$ is the token bucket depth and $r$ its replenishment rate (for such a flow, the total traffic during any time period of length $U$ is bounded by $rU + b$).

The minimal commitment made by controlled-load service makes it especially well suited to the decentralized and heterogeneous Internet. This service model is in the process of being standardized by the Internet Engineering Task Force (IETF). To provide the illusion of lightly loaded network to flows receiving controlled-load service, network switches and routers must perform admission control at the call level to ensure that sufficient resources are available to serve the flows. While the specification of controlled-load service does not dictate specific target values for quality of service parameters such as delay bound or loss rate, operationally the admission control algorithms must be evaluated based on the levels of delay and loss they produce. In this paper we investigate four admission control algorithms that could support controlled-load service.

## 2 Four Admission Control Algorithms

**Simple Sum.** The first admission control algorithm simply ensures that the sum of requested resources does not exceed link capacity. Let $\nu$ be the sum of reserved rates, $\mu$ the link bandwidth, $\alpha$ the name of a flow requesting admission, and $r^\alpha$ the rate requested by flow $\alpha$. This algorithm accepts the new flow if the following check succeeds:

$$\nu + r^\alpha < \mu. \tag{1}$$

We call this the "Simple Sum" algorithm in the rest of the paper. This is the simplest admission control algorithm and hence is being most widely implemented by switch and router vendors. Often, to ensure low queueing delay called for by controlled-load service, an approximation of the weighted fair queueing (WFQ) scheduling discipline is implemented with this admission control algorithm. WFQ assigns each flow its own queue served at its own reserved rate, thereby isolating flows from each other's bursts. In this paper, we always use the WFQ scheduling discipline in conjunction with the "Simple Sum" admission control—incidentally, this setup also satisfies the *committed rate* service model described in [1]. For the other, measurement-based algorithms, we assume the first-in-first-out (FIFO) scheduling discipline.

**Measured Sum.** Whereas the "Simple Sum" algorithm ensures that the sum of existing reservations plus a newly incoming reservation does not exceed capacity, the "Measured Sum" algorithm uses measurement to estimate the load of existing traffic. This algorithm admits the new flow if the following test succeeds:

$$\widehat{\nu} + r^\alpha < \upsilon\mu, \tag{2}$$

where $\upsilon$ is a user-defined utilization target as explained below, and $\widehat{\nu}$ the measured load of existing traffic. We will explain how load measurement is done in the next section. Upon admission of a new flow, the load estimate is increased using $\widehat{\nu}' = \widehat{\nu} + r^\alpha$. As we pointed out in reference [11], in a simple $M/M/1$ queue, variance in queue length diverges as the system approaches full utilization. A measurement-based approach is doomed to fail when delay variations are exceedingly large, which will occur at very high utilization. It is thus necessary to identify a *utilization target* and require that the admission control algorithm strive to keep link utilization below this level. In this paper, we let $\upsilon = 0.9$.

**Acceptance Region.** The second measurement-based algorithm, proposed in [9] computes an acceptance region that maximizes the reward of utilization against the penalty of packet loss. Given link bandwidth, switch buffer space, a flow's token bucket filter parameters, the flow's burstiness, and desired probability of actual load exceeding bound, one can compute an acceptance region for a specific set of flow types, beyond which no more flow of those particular types should be accepted. The computation of the acceptance region also assumes Poisson call arrival process and independent, exponentially distributed call holding times. We refer the interested readers to [9] for the computation of the acceptance region. The measurement-based version of this algorithm ensures that the measured instantaneous load plus the peak rate of a new flow is below the acceptance region. The measured load used in this scheme is not artificially adjusted upon admittance of a new flow. For flows described by a token bucket filter $(r, b)$ but not peak rate, [7] derives their peak rates ($\widehat{p}$) from the token bucket parameters using the equation:

$$\widehat{p} = r + b/U, \quad (3)$$

where $U$ is a user-defined averaging period. We have adopted the same scheme to be used with the acceptance region algorithm. If a flow is rejected, the admission control algorithm does not admit another flow until an existing one leaves the network. In the remainder of this paper, we use the terms "utilization target" and "utilization threshold" interchangeably with "acceptance region."

**Equivalent Bandwidth.** The third measurement-based algorithm computes the equivalent bandwidth for a set of flows using the Hoeffding bounds. The equivalent bandwidth of a set of flows is defined in references [7, 10] as the bandwidth $C(\epsilon)$ such that the stationary bandwidth requirement of the set of flows exceeds this value with probability at most $\epsilon$. We call $\epsilon$ the "loss rate" in the remainder of the paper; however, in an environment where large portion of traffic is best-effort traffic, realtime traffic rate exceeding its equivalent bandwidth is not lost but simply encroaches upon best-effort traffic. In reference [7] the measurement-based equivalent bandwidth based on Hoeffding bounds ($\widehat{C}_H$) assuming peak rate ($p$) policing of $n$ flows is given by:

$$\widehat{C}_H(\widehat{\nu}, \{p_i\}_{1 \leq i \leq n}, \epsilon) = \widehat{\nu} + \sqrt{\frac{ln(1/\epsilon)\sum_{i=1}^{n}(p_i)^2}{2}}, \quad (4)$$

where $\widehat{\nu}$ is the measured average arrival rate of existing traffic and $\epsilon$ is the probability that arrival rate exceeds the link capacity. The author of [7] indicates that the measured average arrival rate may be approximated by measured average load. We use measured arrival rate in our study, but keep the same notation ($\widehat{\nu}$) to refer to both measured arrival rate and measured load for the sake of simplicity. The admission control check when a new flow $\alpha$ requests admission is then:

$$\widehat{C}_H + p^{\alpha} \leq \mu. \quad (5)$$

Upon admission of a new flow, the load estimate is increased using $\widehat{\nu}' = \widehat{\nu} + p^{\alpha}$. Again, if a flow's peak rate is unknown, it is derived from its token bucket filter parameters $(r, b)$ using Eqn. 3. Similar to the algorithm in [9], if a flow is denied admission, no other flow of similar type will be admitted until an existing one departs.

Before moving on to discuss the measurement mechanism in the next section, recall that while the admission control algorithms described here are based on meeting quality of service constraints of either loss rate or delay bound, the specific values used by the admission control algorithms are not advertised to the users of controlled-load service.

## 3 Three Measurement Mechanisms

We now describe the measurement mechanisms used in our study. We note that these may not be the most efficient nor the most rigorous measurement mechanisms. They are however, very simple, which help us isolate admission patterns caused by particular admission control algorithm from those caused by the measurement mechanism. We refer the interested readers to references [17, 6, 4] for alternate treatments on measurement mechanisms.

**Time-window.** Following [11], we use a simple time-window measurement mechanism to measure network load with the "Measured Sum" algorithm. As shown in Fig. 1, we compute an average load every $S$ sampling period. At the end of a measurement window $T$, we use the highest average from the just ended $T$ as the load estimate for the next $T$ window. When a new flow is admitted to the network, the estimate is increased by the parameters of the new request as explained in the previous section. If a newly computed average is above the estimate, the estimate is immediately raised to the new average. At the end of every $T$, the estimate is adjusted to the actual load measured in the previous $T$. A smaller $S$ gives us higher maximal averages, resulting in a more conservative admission control algorithm; a larger $T$ keeps longer measurement history, again resulting in a more conservative admission control algorithm. To get a statistically meaningful number of samples, we try to keep $T/S \geq 10$.

**Point Samples.** The measurement mechanism used with the acceptance region algorithm takes an average load sample every $S'$ period [12].

**Exponential Averaging.** Following reference [7] we use an estimate of the average arrival rate, instead of instantaneous bandwidth, to compute admission decisions with the equivalent bandwidth approach. The average arrival rate ($\widehat{\nu}^S$) is measured once every $S$ sampling period. The average arrival rate is then computed using an infinite impulse response function with weight $w$, which we set to 2e-3 in this study:

$$\widehat{\nu}' = (1 - w) * \widehat{\nu} + w * \widehat{\nu}^S. \quad (6)$$

If the traffic arrival rate changes abruptly from 0 to 1 and then remains at 1, a $w$ of 2e-3 allows the estimate to reach 75% of the new rate after 10 sampling periods. A larger $w$ makes the averaging process more adaptive to load changes; a smaller $w$ gives a smoother average by keeping a longer history. Recall that the equivalent bandwidth based admission control algorithm requires peak rate policing and

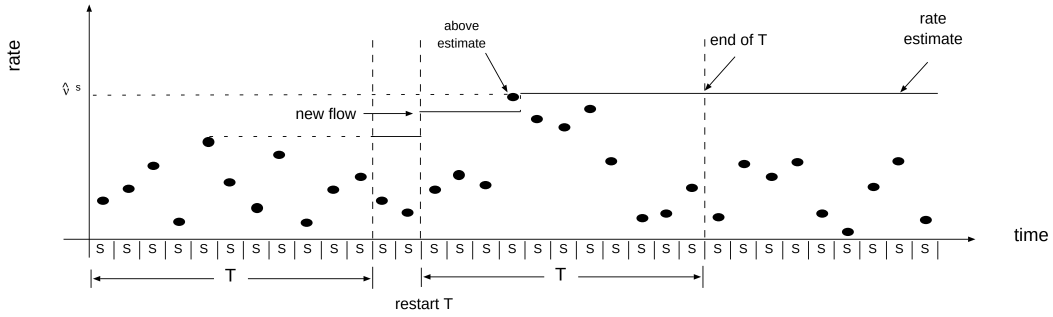


Figure 1: Time-window measurement of network load.

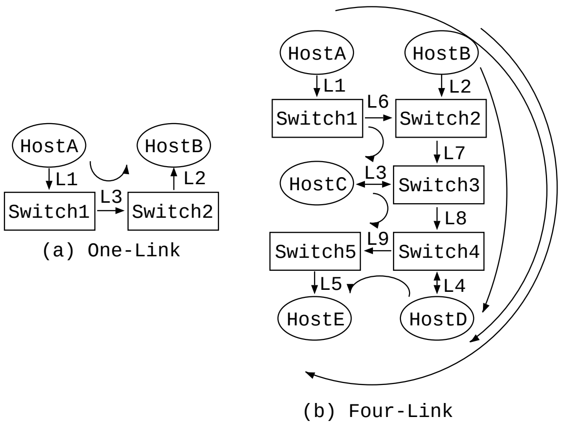


Figure 2: The ONE-LINK and FOUR-LINK topologies

derives a flow's peak rate from its token bucket parameters using Eqn. 3 when the peak rate is not explicitly specified. We set $U = S$ to reflect the peak rate seen by the measurement mechanism. A smaller $S$ not only makes the measurement mechanism more sensitive to bursts, it also makes the peak rate derivation more conservative. A larger $S$ may result in lower averages, however it also means that the measurement mechanism keeps a longer history because the averaging process (Eqn. 6) is invoked less often.

# 4 Simulation Scenarios

For this paper, we run our simulations on two topologies: the ONE-LINK and FOUR-LINK topologies depicted in Figure 2. In both topologies, each host is connected to a switch by an infinite bandwidth link. The connection between switches are all 10 Mbps links. Buffer space at the switches are shared by all admitted flows. Buffer space of the switches connected to the bottleneck links are sized differently for each simulation, as we explain below. In the ONE-LINK topology, traffic flows from HostA to HostB. In the FOUR-LINK topologies, traffic flows between six host pairs: HostA–HostC, HostB–HostD, HostC–HostE, HostA–HostD, HostB–HostE, HostD–HostE; flow creations are distributed uniformly among the six host pairs. In Figure 2, these host pairs and the paths their packets traverse are indicated by the directed curve lines.

We use two kinds of source model in our simulations. The first is an ON/OFF model with exponentially distributed ON and OFF times. During each ON period, an exponentially distributed random number of packets, with average $N$, are generated at fixed rate $p$ packet/sec. Let $I$ milliseconds be the average of the exponentially distributed OFF times, then the average packet generation rate $a$ is given by $1/a = I/N + 1/p$.

Recent studies ([13, 8, 2]) have shown that network traffic often exhibits long-range dependence (LRD), with the implications that congested periods can be quite long and a slight increase in the number of active connections can result in large increase in packet loss rate [16]. Reference [16, 7] further call attention to the possibly damaging effect long-range dependent traffic might have on measurement-based admission control algorithms. To investigate this and other LRD related questions, we augment our simulation study with LRD source models. Following [18], our next model is an ON/OFF process with Pareto distributed ON and OFF times (for ease of reference, we call this the *Pareto-ON/OFF* (POO) model). Pareto distribution is a heavy-tailed distribution that can be described by two parameters: its location and shape. A Pareto shape parameter less than 1 gives data with infinite mean; shape parameter less than 2 results in data with infinite variance. The Pareto location parameter can be computed from the formula: $mean * (shape - 1)/shape$. During each ON period of the Pareto-ON/OFF model, a Pareto distributed number of packets, with mean $N$ and Pareto shape parameter $\beta$, are generated at some peak rate $p$ packet/sec. The OFF times are also Pareto distributed with mean $I$ milliseconds and shape parameter $\gamma$. Each Pareto-ON/OFF source by itself does not generate LRD series; the aggregation of them does.

In addition to each source's burstiness, network traffic dynamics is also effected by the arrival pattern and duration of flows. We use exponentially distributed lifetimes with the Markov-ON/OFF source model, following [14]. The duration of Pareto-ON/OFF sources, however, are taken from a lognormal distribution, following [3, 5]. The interarrival times of all flows are exponentially distributed [16].

We use six instantiations of the above source models as summarized in Table 1. In the table, $p = \infty$ means that after each OFF time, packets for the next ON period are transmitted back to back. The shape parameter of the Pareto distributed ON time ($\beta$) of the Pareto-ON/OFF sources are selected following the observations in [18]. According to the same reference, the shape parameter of the Pareto distributed OFF time ($\gamma$) stays mostly below 1.5; in this paper we use $\gamma = 1.1$ for all Pareto-ON/OFF sources. For the POO1 and POO2 models, we use a token bucket rate equal to the source's peak rate so that the token bucket filter does not reshape the traffic. For the POO3 model, some of the generated packets were queued; this means during some of the source's alleged "OFF" times, it may actually still be draining its data queue onto the network.

In the same table, we also list the settings of the token bucket parameters assigned to each source. In this study, we assign each flow a data queue with infinite length (i.e. packets that arrive at an empty token bucket are queued, and the queue never overflows). Column 7 of the table, labeled *max qlen*, shows the maximum data queue length a flow can expect to see. Our packets are of fixed size (1 Kbits) and each of our token is worth 1 packet of data.

When a flow with token bucket parameters $(r, b)$ is served with WFQ, the maximal queueing delay (ignoring terms proportional to a single packet time) is given by $b/r$ [15]. Column 8 of the table, labeled $D^*$, lists the maximal delay for each source given its assigned token bucket filter. This is also the "burst time" queueing delay acceptable under the definition of controlled-load service. Column 9, labeled $D$, lists the delay bound we assigned to each source. We have chosen the token bucket parameters such that, in most cases, the delay bounds given to a flow will be the same as its "burst time" queueing delay. This facilitates analyzing the performance of the algorithms under controlled-load service. In the few cases where the delays are not the same, such as in the POO3 case, the utilization comparison is less meaningful. For each simulation with measurement-based admission control algorithm, we size the buffer at the switches with enough space to accommodate the delay bound ($D$). For example, simulations with EXP1 source, given a link speed of 10 Mbps, use a buffer size of 160 packets. In simulations with multiple source models having different delay bound requirements, we use the maximum of the required buffer sizes; for example, in a simulation with both EXP1 and EXP2 models, we use a buffer size of 1600 packets. Simulations with the parameter-based admission control algorithm assume infinite buffer size. Column 10, labeled $\Upsilon$, contains the utilization threshold used to simulate the acceptance region algorithm. This utilization threshold should not be confused with the utilization target used with the "Measured Sum" scheme, where the value is set to 90% link bandwidth. The acceptance region is computed in [9] for $\epsilon$ of 1e-12. For simulations with the equivalent bandwidth algorithm, we use $\epsilon$ of 1e-7. The next column, labeled $S$, gives the sampling period used with the measurement mechanisms in packet transmission time (*ptime*). For the time-window mechanism, the window size is $10 * S$. The sampling period used with the acceptance region algorithm is $S' = 5 * S$. The last column, labeled $\widehat{p}$, contains the derived peak rates computed with Eqn. 3. Note that for source POO3, the derived peak rate is larger than the actual peak rate. In this paper, we also consider using the token rate ($r$) as the peak rate.

For each simulation, flow interarrival times are exponentially distributed with an average of 400 milliseconds. The average holding time of all Markov-ON/OFF sources is 300 seconds. The Pareto-ON/OFF sources have lognormal distributed holding times with median of 300 seconds and shape parameter 2.5. We run simulations with Markov-ON/OFF sources for 3000 seconds simulated time, serving 1e8 packets. The data presented are obtained from the later half of each simulation. By visual inspection, we determined that 1500 simulated seconds is sufficient time for the simulations to warm up. Simulations involving Pareto-ON/OFF sources require a longer warmup period and a longer simulation time for the LRD effect to be seen, thus we run them for 5.5 hours simulation time, serving close to 1e9 packets, with reported data taken from the later 10000 seconds.

# 5 Simulation Results

In the following three subsections, we present results from simulations on the ONE-LINK topology. A summary of the results is presented in Table 2. Each row of the table contains results from up to six simulations using the source model named at the leftmost column and the admission control algorithm indicated at the head of the columns. The "%Util" columns list the average utilization achieved at the bottleneck link of the ONE-LINK topology. The "#Active" columns list the average number of concurrently running flows in steady state. We have repeated some of the simulations reported in the table with up to ten different random seeds; the 99% confidence intervals are less than one of the least significant digit in most cases. We close this section by presenting results from simulations on the FOUR-LINK topology.

## 5.1 Single-hop Results

The first two columns of Table 2 show results from simulation using the "Simple Sum" parameter-based admission control algorithm. There are no lost packets. The second set of two columns show results from the "Measured Sum" algorithm. Except for the POO1 cases, where the "Mea-

Table 1: Six Instantiations of the Two Source Models

| Model Name | Model Parameters | | | | TB Filter | | | Switch Parameters | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $p$ pkt/sec | $I$ msec | $N$ pkts | $p/a$ | $r$ tkn/sec | $b$ tkns | max qlen | $D^*$ msec | $D$ msec | $\Upsilon$ (%) | $S$ ptime | $\widehat{p}$ pkt/sec |
| EXP1 | 64 | 325 | 20 | 2 | 64 | 1 | 0 | 16 | 16 | 98 | 5e3 | – |
| EXP2 | 1024 | 90 | 10 | 10 | 320 | 50 | 17 | 160 | 160 | 40 | 1e3 | 832 |
| EXP3 | $\infty$ | 684 | 9 | $\infty$ | 512 | 80 | 1 | 160 | 160 | – | 5e2 | 2150 |
| | | | | $\beta$ | | | | | | | | |
| POO1 | 64 | 325 | 20 | 1.2 | 64 | 1 | 0 | 16 | 16 | 98 | 5e3 | – |
| POO2 | 64 | 2925 | 20 | 1.2 | 64 | 1 | 0 | 16 | 16 | 98 | 5e3 | – |
| POO3 | 256 | 360 | 10 | 1.9 | 240 | 60 | 220 | 256 | 160 | 44 | 5e3 | 363 |

Table 2: Single-hop Homogeneous Sources Simulation Results. Entries in *italics* indicate simulations with non-zero loss rate.

| Model Name | Simple Sum | | Measured Sum | | Acc. Rgn. ($\widehat{p}$) | | Eqv. Bw. ($\widehat{p}$) | | Eqv. Bw. ($r$) | | Eqv. Bw. ($r$, $\epsilon$=1e-1) | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | %Util | #Actv | %Util | #Actv | %Util | #Actv | %Util | #Actv | %Util | #Actv | %Util | #Actv |
| EXP1 | 46 | 144 | 79 | 250 | *97* | *308* | 71 | 223 | – | – | – | – |
| EXP2 | 28 | 28 | 75 | 74 | 38 | 37 | 12 | 12 | 41 | 40 | 68 | 67 |
| EXP3 | 2 | 18 | 54 | 406 | – | – | 0.3 | 2 | 5 | 40 | 23 | 171 |
| POO1 | 39 | 144 | *86* | *330* | *99* | *381* | 70 | 270 | – | – | – | – |
| POO2 | 7 | 144 | 78 | 1539 | *97* | *1912* | 45 | 885 | – | – | – | – |
| POO3 | 3 | 38 | 72 | 965 | 43 | 582 | 2 | 27 | 12 | 159 | 37 | 505 |

sured Sum" algorithm gives a loss rate on the order of 1e-7, simulations with other source models using this algorithm do not result in any loss. We can achieve no loss with POO1 sources if we reduce the utilization target to 80% of link bandwidth, in which case the average link utilization achieved is 77% and the average number of concurrently served flows is 297.

The next two columns, under the heading "Acc. Rgn. ($\widehat{p}$)" give the results of simulations with the acceptance region algorithm; here the peak rate of sources with token bucket greater than 1 is derived from their token bucket parameters using Eqn. 3. We do not study the performance of this algorithm for EXP3 source because the utilization threshold for this model is 0 [9]. The loss rate for sources EXP1, POO1, and POO2, from three simulations of each with different random seeds, are 6e-3, 2e-2, and 1e-2 respectively. Since EXP1 and POO1 sources have mostly the same characteristics except for the distributions of their ON and OFF times, we conjecture that by not taking these into account, the acceptance region algorithm is overly optimistic for sources having heavy-tailed ON and OFF times distributions. As the grain size of flows, i.e. the $p/\mu$ ratio, enlarges, this algorithm becomes more conservative. For the EXP2 and POO3 sources, achievable utilization is about half of the utilization under "Measured Sum." Since the acceptance region algorithm does not artificially adjust its measurement values upon admittance of a new flow, and given its point sampling measurement process, we do not see any significant difference in performance between simulations using derived peak rates and token rates as peak rates.

The two columns of Table 2 under the heading "Eqv. Bw. ($\widehat{p}$)" show results from simulations using the equivalent bandwidth based admission control algorithm. In the case of simulations with EXP2, EXP3 and POO3 sources, the flows' peak rates are derived from their token bucket parameters using Eqn. 3. We showed the derived peak rates for the three sources in Table 1 and pointed out that in the POO3 case the derived peak rate is higher than the actual peak rate. To see how a less conservative peak rate effects the performance of the algorithm on the EXP2, EXP3, and POO3 sources, we simulate them with their token bucket rates as their peak rates, ignoring the token bucket depths. The two columns of Table 2 under the "Eqv. Bw. ($r$)" heading show results from these simulations. The performance of the algorithm improves significantly, but still lags behind the other measurement-based algorithms. Next we experiment with $\epsilon$ = 1e-1, using the token rate as peak rate, for sources EXP2, EXP3, and POO3. The results are presented under the "Eqv. Bw. ($r$, $\epsilon$=1e-1)" heading. Note that these numbers are still lower than those achieved with the "Measured Sum" algorithm. Due to its conservativeness, none of the simulations with the equivalent bandwidth algorithm results in packet loss.

## 5.2 Multiple-hop Results

Table 3 contains the average link utilization and average number of connections of the four links in the FOUR-LINK topology from simulations where we run all six sources, with the choice of sources uniformly distributed. Simulations with the "Measured Sum" and equivalent bandwidth algorithms use a sampling period of 1e3 packet transmission times while that of acceptance region use a sampling period five times as large. The switches have buffer space for 1600 packets in all these simulations. The table shows that the equivalent bandwidth algorithm is, again, rather conservative in this scenario. The "Acceptance Rgn. 1" scenario uses a utilization threshold of 40%, whereas the "Acceptance Rgn. 2" scenario uses 98%. With a 98% utilization threshold, the acceptance region algorithm results in 5e-5, 6e-5, and 8e-7 loss rate at links L7, L8, and L9 respectively. Under both the acceptance region and equivalent bandwidth algorithms, sources that do not declare their peak rates are admitted based on their derived peak rates per Eqn. 3.

Note that link L6 consistently achieves lower utilization than the other links. We call this the *under-representation* phenomenon in reference [11], and attribute its cause to unconsummated reservations when multi-hop flows admitted by the switch attached to L6 are rejected by one of the downstream switches. Table 4 shows the composition of the type of admitted flows, in percentages. It confirms our earlier observation in [11] that more resource demanding sources, namely EXP2, EXP3, and POO3, can suffer from another form of *under-representation*, where they are discriminated against by the network.

While the utilization achieved under the acceptance region algorithm is high when the utilization threshold is 98%, the choice of the utilization threshold is not from computation in [9], rather it is a "best case," but *ad-hoc*, choice for this scenario—hence does not allow the load estimation error to be quantified and assessed any more more rigorously than under the "Measured Sum" method. To compute the acceptance region, one must know the source characteristics *a priori*. On environments such as the Internet, where new applications are introduced at a high rate, and source characteristics depend not only on the applications but also their use, one cannot make *a priori* characterization of sources with any degree of certainty.

# 6 Conclusion

We have presented some preliminary results from the comparison of four admission control algorithms for controlled-load service. We hope that this paper will start a dialog with others in the field on a suitable method to conduct such study. We ourselves plan to do further, more systematic, comparative analysis of the various algorithms presented here.


# Acknowledgments

We have benefited from various discussions with Lixia Zhang on measurement-based admission control, probability and statistics, running simulations, and writing papers. We thank Sally Floyd for many fruitful discussions on long-range dependent traffic and equivalent bandwidth. Frank Kelly and Richard Gibbens assisted us in understanding their scheme. This paper has also been improved by comments from Lee Breslau, Sally Floyd, and Lixia Zhang. We also thank the anonymous referees for their comments.


# References


[1] F. Baker, R. Guérin, and D. Kandlur. *Specification of the Committed Rate Quality of Service*. Internet-Draft, Jun. 1996.

[2] J. Beran, R. Sherman, M.S. Taqqu, and W. Willinger. "Long-range Dependence in Variable-Bit-Rate Video Traffic". *IEEE Transactions on Communications*, 43:1566–1579, 1995.

[3] V.A. Bolotin. "Modeling Call Holding Time Distributions for CCS Network Design and Performance Analysis". *IEEE Journal on Selected Areas in Communications*, 12(3):433–438, Apr. 1994.

[4] C. Casetti, J. Kurose, and D. Towsley. "An Adaptive Algorithm for Measurement-based Admission Control in Integrated Services Packet Networks". *Proc. of the Protocols for High Speed Networks Workshop*, Oct. 1996.

[5] D.E. Duffy, A.A. McIntosh, M. Rosenstein, and W. Willinger. "Statistical Analysis of CCSN/SS7 Traffic Data from Working CCS Subnetworks". *IEEE Journal on Selected Areas in Communications*, 12(3):544–551, Apr. 1994.

[6] Z. Dziong, M. Juda, and L.G. Mason. "A Framework for Bandwidth Management in ATM Networks — Aggregate Equivalent Bandwidth Estimation Approach". *Submitted for publication*, 1996.

[7] S. Floyd. "Comments on Measurement-based Admissions Control for Controlled-Load Service". Submitted to *Computer Communication Review*, 1996. **URL** ftp://ftp.ee.lbl.gov/papers/admit.ps.Z.

[8] M. Garrett and W. Willinger. "Analysis, Modeling and Generation of Self-Similar VBR Video Traffic". *Proc. of ACM SIGCOMM '94*, pages 269–279, Sept. 1994.

[9] R.J. Gibbens, F.P. Kelly, and P.B. Key. "A Decision-Theoretic Approach to Call Admission Control in ATM Networks.". *IEEE Journal on Selected Areas in Communications*, 13(6):1101–1114, Aug. 1995.

[10] R. Guérin, H. Ahmadi, and M. Naghshineh. "Equivalent Capacity and Its Application to Bandwidth Allocation in High-Speed Networks". *IEEE Journal on Selected Areas in Communications*, 9(7):968–981, Sept. 1991.

Table 3: Multiple-hop All Sources Simulation Results. Entries in *italics* indicate simulations with non-zero loss rate.

| Link | Measured Sum | | Acceptance Rgn. 1 | | Acceptance Rgn. 2 | | Equivalent Bw. | |
|---|---|---|---|---|---|---|---|---|
| | %Util | #Actv | %Util | #Actv | %Util | #Actv | %Util | #Actv |
| L6 | 47 | 282 | 23 | 145 | 54 | 332 | 9 | 64 |
| L7 | 79 | 485 | 38 | 242 | *94* | *563* | 16 | 104 |
| L8 | 77 | 469 | 36 | 228 | *93* | *547* | 18 | 116 |
| L9 | 77 | 469 | 38 | 234 | *93* | *532* | 25 | 158 |

Table 4: Percentage Composition of Type of Admitted Flows

| Algorithm | EXP1 | EXP2 | EXP3 | POO1 | POO2 | POO3 |
|---|---|---|---|---|---|---|
| Measured Sum | 21% | 9 | 12 | 21 | 22 | 15 |
| Acceptance Rgn. 1 | 21 | 9 | 13 | 21 | 21 | 16 |
| Acceptance Rgn. 2 | 19 | 13 | 15 | 19 | 19 | 16 |
| Equivalent Bw. | 26 | 3 | 7 | 26 | 27 | 12 |


[11] S. Jamin, P. B. Danzig, S. J. Shenker, and L. Zhang. "A Measurement-based Admission Control Algorithm for Integrated Services Packet Networks (Extended Version)". *ACM/IEEE Transactions on Networking*, Feb. 1997. **URL** http://netweb.usc.edu/jamin/admctl/ton96.ps.Z.

[12] F. Kelly. Personal communication. Workshop on System and Control Issues in Communication Networks, Airlie, VA, Aug. 10, 1996.

[13] W.E. Leland, M.S. Taqqu, W. Willinger, and D.V. Wilson. "On the Self-Similar Nature of Ethernet Traffic (Extended Version)". *ACM/IEEE Transactions on Networking*, 2(1):1–15, Feb. 1994.

[14] E.C Molina. "Application of the Theory of Probability to Telephone Trunking Problems". *The Bell System Technical Journal*, 6:461–494, 1927.

[15] A.K. Parekh. *A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks*. PhD thesis, MIT, Lab. for Information and Decision Systems, Tech. Report LIDS-TR-2089 1992. Parts of this thesis were also published with R.G. Gallager in the *ACM/IEEE Transactions on Networking*, 1(3):344-357 and 2(2):137-150.

[16] V. Paxson and S. Floyd. "Wide-Area Traffic: The Failure of Poisson Modeling". *Proc. of ACM SIGCOMM '94*, pages 257–268, Aug, 1994. An extended version of this paper is available as **URL** ftp://ftp.ee.lbl.gov/papers/poisson.ps.Z.

[17] R. Warfield, S. Chan, A. Konheim, and A. Guillaume. "Real-Time Traffic Esitmation in ATM Networks". *International Teletraffic Congress*, June 1994.

[18] W. Willinger, M.S. Taqqu, R. Sherman, and D.V. Wilson. "Self-Similarity Through High-Variability: Statistical Analysis of Ethernet LAN Traffic at the Source Level". *Proc. of ACM SIGCOMM '95*, pages 100–113, Aug. 1995.

[19] J. Wroclawski. *Specification of the Controlled-Load Network Element Service*. Internet-Draft, Nov. 1995.