KNOWING TRAFFIC CHARACTERISTICS ON MULTIPLE TIME SCALES IMPROVES EFFICIENCY OF CALL ADMISSION CONTROL POLICIES

Arnie Neidhardt
Bellcore
331 Newman Springs Road
Red Bank, NJ 07701
arnie@bellcore.com

Ljiljana Trajković
UC Berkeley
EECS Department
Berkeley, CA 94720-1770
ljilja@eecs.berkeley.edu

Abstract

We investigate the importance of understanding traffic characteristics for Call Admission Control (CAC) policies in an ATM network. We are particularly interested in whether it pays off to characterize traffic on more than one time scale. Specifically, we compare the network utilization that could be achieved with complete knowledge of traffic characteristics, against the utilization achieved with an admission policy based on traffic knowledge on only one time scale.

1 Introduction

A Call Admission Control (CAC) policy in an ATM network should, in principle, depend on the Quality of Service (QoS) that needs to be provided to admitted users. Under this ideal condition, a decision to admit a new call is based on a prediction of the new QoS that would result from the additional traffic. One of the important obstacles to an accurate QoS prediction is the lack of understanding of the traffic characteristics. Without accurate predictions of the resulting QoS, a network may adopt an overly conservative admission policy that, while assuring certain agreed upon QoS, lowers network utilization. Of course, if more detailed forecasts of network traffic were available, then QoS predictions could be sharpened and network utilization improved.

We want to distinguish here two obstacles to the creation of a CAC that ensures agreed upon QoS, while allowing the network to be used efficiently:

1. forecasting the relevant traffic features (traffic characterization), and
2. predicting QoS from characteristics of the admitted traffic.

In our article we address the advantage of including traffic information details in forecasting, but we do not solve the QoS prediction. Indeed, our investigation of forecasting is hindered by our inability to solve prediction.

2 Network Information and Call Admission Control Policies

Information about a network may include several sets of network and traffic parameters. The optimal network CAC policy will depend on how well we know these parameters.

Traffic may be described in many ways, but a natural approach is to describe the random amounts of traffic generated in various intervals [1], [2], [4], [5]. No engineer is likely to know in detail all the joint distributions of all these random amounts, but some of the less complex statistics are likely to be available. For instance, there may be one well studied time scale $s > 0$, for which one might know reasonably well the distribution of the random amount of traffic generated in a given interval of length s. With more ambition, one may attempt to learn about the distributions corresponding to more than one time scale s, or even about joint distributions. The subject of this investigation is whether such ambition pays off, i.e., this report is about how much more efficiently a network can be run if one knows more about traffic than only a single time scale.

One particularly relevant scale is the half-buffer work time. This is the time that the queue server requires to serve the amount of traffic that could be stored in half of the allocated buffer. This is the most relevant time scale for a conservative admission policy that relies on the knowledge of only one time scale. Basically, if excess arrivals on the half-buffer time scale almost never occur, then any excesses at smaller scales cannot cause an overflow.

Available traffic information will influence the choice and the performance of adopted CAC policy, and, ultimately, the achieved QoS parameters in a particular network. We begin by listing of network CAC policies, traffic information, and policy performance.

1. For systems in which a user's *line speed* is known, with no knowledge of the traffic that the line will carry, the *(physical) peak-rate allocation* admission policy is the only viable option. Naturally, this scheme often achieves very little efficiency.

2. If a user's *maximum traffic rate* on the half-buffer time scale is known, the *(traffic) peak-rate allocation* policy is justifiable. This approach yields an improved efficiency, even for links carrying only one call (i.e., links whose capacity is less than the user's physical peak can now serve the call). Nevertheless, with such a policy no further gains from statistical multiplexing are possible. Incidentally, this (traffic) peak-rate policy might be described as a "mean-rate" policy by fans of policing, because the traffic peak on the time scale of interest will be very close to the Sustained Cell Rate (SCR) parameter of a policing mechanism.

3. Knowing a user's *mean traffic rate*, in addition to the peak, would lead to some multiplexing gains, and would even allow link utilizations to approach unity at very large capacities.

4. Additional knowledge of the *variance* of a user's traffic on the half-buffer time scale would allow further gains. One quantitative indication of these gains is that, asymptotically, the reduction of idle time with capacity will approach an ideal value.

5. Knowing the *complete marginal distribution* of a user's traffic on the single scale of a half-buffer work time would allow further multiplexing gains, especially at the more practical capacities below the asymptotic regime.

6. Knowing the *traffic distribution at more than one time scale* should lead to further increases in utilization.

7. *Complete traffic information* would allow the use of an ideal policy. The "only" restriction would be the computational complexity in implementing the policy.

The improvements along the progression (1–7) are to be expected from the simple observation that greater knowledge opens the possibility of greater efficiency. For Cases 1–4, the possibilities for extreme levels of multiplexing are not quite obvious, but are, nevertheless, still theoretically derivable. To evaluate these cases quantitatively at intermediate levels of multiplexing, we have examined ten CAC policies with multiplexed realistic traffic traces.

3 Traffic Traces Used in Simulations

For our comparison of various CAC policies, we used genuine video and data traffic trace.

Video Traffic: The video-conference trace was the source of "real data" of a teleconference showing talking heads and shoulders, used in [3]. The trace lists the number of 64-octet cells in each of approximately 48,500 frames. It corresponds to approximately 30 minutes of a video at 24 frames/second.

Data Traffic: The data traffic trace was acquired by measuring TCP/IP traffic running over HIPPI network at the Pittsburgh Supercomputing Center used for storage and retrieval of data files. In addition to the data traffic, the trace also includes rather small signaling packets, and the acknowledgment (ACK) packets. The TCP MTU (Maximum Transmission Unit) was set to 64 Kbytes. The selected traffic session lasted 1.972×10^4 seconds (5.48 hours). The trace shows large variability in the packet inter-arrival times and packet sizes.

4 Simulation Results

We have examined the efficiency (occupancy) ρ of a link when it is "filled" with sessions admitted according to a general call-admission-control rule *CACname*, subject to particular features of the link, the traffic, and the service standard.

The link parameters are its speed C cells/second and its buffer size $2Ch$ cells, where h is roughly a half-buffer work time. The *TrafficDescription* of a single session is taken from the two available traffic traces. The service standard criterion is

$$P(queue\ overflow\ during\ a\ service\ monitoring\ interval\ D) \leq \epsilon.$$

We compared ten CAC policies. For each case, one can choose four parameters: *TrafficDescription*, h, D, and ϵ. They determine how many sessions ($Nmax$) of type *TrafficDescription* can be admitted simultaneously. This, in turns, determines the occupancy ρ of the link when the link is filled:

$$\rho = Nmax * meanrate\ (TrafficDescription)/C.$$

Simulation results for the video-conference and the HIPPI traces are shown in Figure 1 and Figure 2, respectively.

For a video-conference trace, we found that nearly all the multiplexing gains could be achieved if we knew the traffic peak, the mean, and the variance on a single time scale (Case 4).

For data traffic, however, further knowledge of the traffic was required in order to improve network utilization. For the HIPPI traffic we have examined, the ideal admission policy seemed to more than double the utilization achieved by the Case 5 policy. Specifically, an OC-24 link was able to handle 517 data streams, whereas the Case 5 policy would allow only 195 streams concurrently present on the link. In the case of 517 streams, while half-buffer work times, in which arrivals exceeded capacity, occurred often enough to "frighten" the Case 5 policy, the neighboring half-buffer work times usually encountered few arrivals. Realization of an efficient admission-control policy, therefore, requires quantitative knowledge of these "negative correlations" between arrivals in neighboring intervals. (The correlations referring to typical neighboring intervals are positive. Nevertheless, when dealing with very large excursions, the probability of maintaining a large excursion is quite small, even for positively correlated random variables.)

Tremendous burstiness distinguished the data trace from the video-conference trace. A 67% occupancy could be achieved with video conferencing merely by using an OC-6 link to multiplex 109 sessions with a buffer of only 1418 cells to handle bursts. In contrast, to achieve noticeable multiplexing gains with the data traffic, the higher-capacity OC-24 link was required (to allow more sessions in the statistical mix), and a buffer of 5,608,031 cells (2 seconds of work) was required to absorb the bursts. Even then, the occupancy achieved in multiplexing 517 sessions was only 55%. We emphasize in this article the imperative

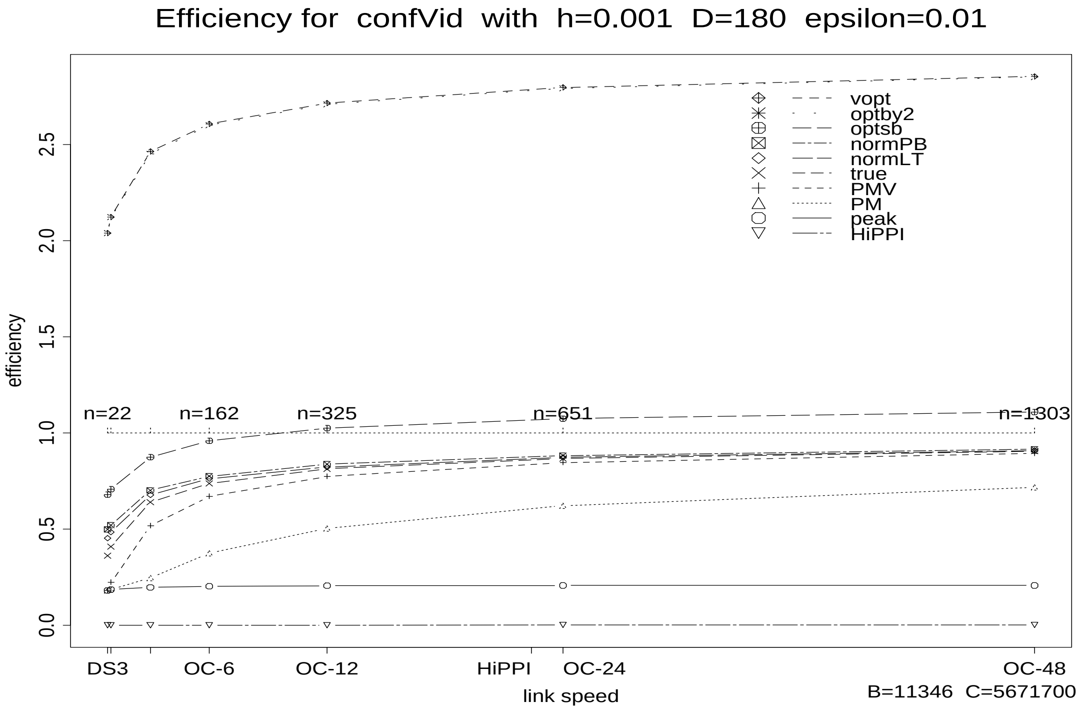

Figure 1: Efficiency for video-conferencing traffic.

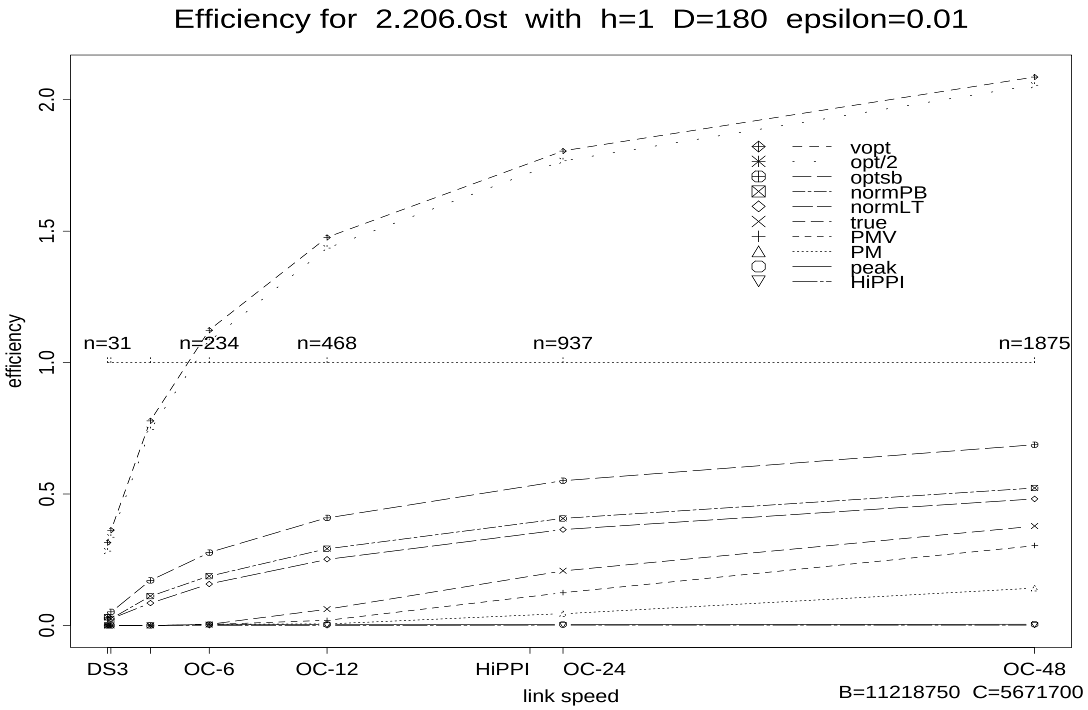

Figure 2: Efficiency for HIPPI traffic.

to understand traffic at more than one time scale when dealing with bursty traffic, in order to achieve even the 55% occupancy. In other words, even with a 5 Megacell size buffer for an OC-24 link, the occupancy would be only 20% if traffic characteristics on only one time scale were available.

Identifying relevant time scales is difficult, and we could only determine some rather loose bounds. The first time scale, naturally, is the time needed to serve half a buffer (1 second for the case of multiplexed data traffic that we considered). For other time scales, we could only determine that it was necessary to understand the traffic at some time scale between 10 seconds and 6 minutes. In particular, it was important to understand traffic over a range covering more than an order of magnitude (from 1 second to more than 10 seconds), and very likely close to two orders of magnitude (from 1 second to 6 minutes = 360 seconds).

How do we know that time scales longer than 10 seconds matter? First, only 410 simultaneous sessions could be justified by the single-time-scale method for a buffer size 10 times as large. Since this method is based on the probability of arrivals exceeding capacity for 10 seconds, the observation that only 410 sessions can be justified means that with 517 concurrent sessions there are many 10-second intervals in which arrivals exceed capacity. Second, 517 happens to be the number of sessions that could be justified by the single-time-scale method for a link 20% faster and a buffer 20% larger (so the half-buffer work time is the same as for the simulated case). In other words, with 517 concurrent sessions present, the 1-second intervals in which arrivals exceed the 20%–enlarged capacity are just barely rare enough to be tolerated. They would not have been tolerated with any smaller enhancement of capacity. Hence, the 1-second intervals in which arrivals exceed 110% (say) of capacity are too frequent to be tolerated on their own without some knowledge of isolation. Thus, because 517 sessions seem to be supportable, most of the relatively frequent 1-second intervals, in which arrivals exceed 110% of capacity, are surrounded by other intervals of light enough arrivals that the queue will be emptied. The first observation, however, says that on such occasions, the queue will usually not be emptied for at least 10 seconds. Thus, for a call-admission-control algorithm to justify 517 simultaneous sessions, it must be based on knowledge of traffic features that range from 1 second to more than 10 seconds.

How do we know that important correlations disappear beyond 6 minutes? In essence, we tried to look at whether a typical episode of queue overflow was brief or durable. We found that typical durations were of a few seconds, though with quite a few lasting longer than a minute, but none seem to have lasted more than 6 minutes.

Unfortunately, the examination of the duration of queue overflow episodes had to be indirect. Exact durations are not available, for the simple reason that two neighboring busy periods are usually part of a single larger episode of congestion events, and it is the length of the larger episode that is in question. Nevertheless, based on reports of neighboring 3-minute intervals with losses, one can estimate how long the episodes are. For instance, one aid to interpretation is that the number of lossy 3-minute intervals formed only 1% of the total. Hence, one is free to interpret almost any incident of neighboring lossy intervals as a single episode of long duration, unless the number of such incidents is only about 1% of the total number of lossy intervals. Similarly, if a typical lossy episode lasting for the fraction x of a minute is detected by reports generated every minute, then x is also the probability that the episode will be recorded in consecutive reports of lossy intervals. Based on considerations such as these, we found that a typical congestion episode lasted only 6 seconds, but quite a few lasted longer than a minute. As for the longest episodes, so few reports were consistent with episodes longer than 6 minutes, that we conclude that these reports were due to coincidences of separate episodes happening close in time. Simulation results are shown in Figure 3.

Another, somewhat separate, observation is that the particular service standard is somewhat irrelevant. Thus, a user's session is either entirely lossless or not, and, in the latter case, losses will disrupt the session severely. For instance, in all the simulation runs of an OC-24 link carrying 517 data sessions, only 0.94% of the various 3-minute intervals that we examined had any losses; yet the average cell-loss rate was 2×10^{-4}, which means that on the few occasions when losses occurred, the losses were horrendous. Moreover, for an average loss episode of the 517 concurrent sessions, 359 sessions had losses. Thus, so long as the admission-control policy attempts to be efficient, there will be some episodes of loss, and each episode will be horrendous. What will differentiate various policies will be how rare the policies make these episodes.

Loss Episodes

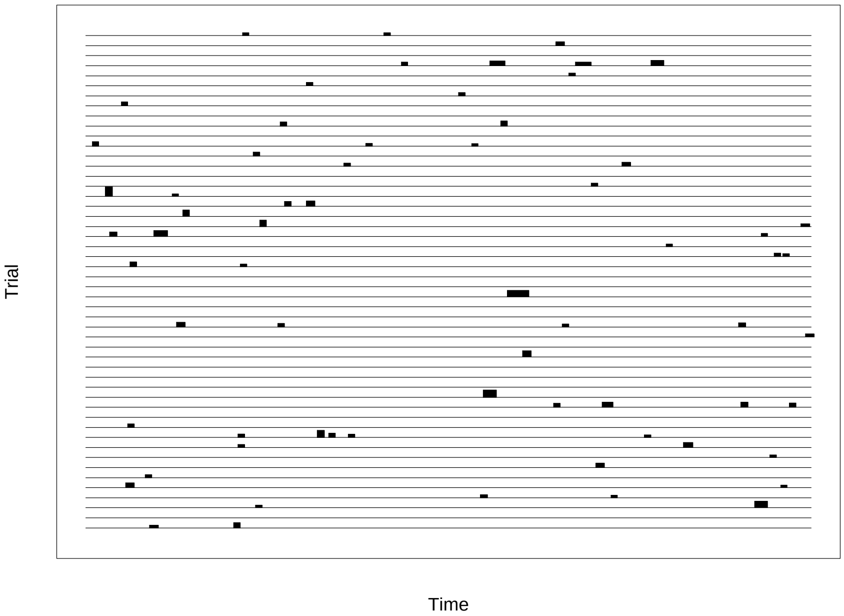

Figure 3: Loss episodes over the period of 5.5 hours in various trials.

5 Concluding Remarks

Under an ideal condition that a CAC policy in an ATM network should depend on the QoS that needs to be provided to admitted users, a decision to admit a new call is based on a prediction of the new QoS that would result from the additional traffic. One of the important obstacles to accurate QoS prediction is the lack of understanding of the traffic characteristics. We demonstrated that without accurate predictions of the resulting QoS, a network may adopt an overly conservative admission policy that, while assuring certain agreed upon QoS, lowers network utilization. If more detailed forecasts of network traffic were available, then QoS predictions could be sharpened and network utilization improved.

References

[1] D. R. Cox, "Long-Range Dependence: A Review," Statistics And Appraisal, *Proc. 50th Anniversary Conf.*, Iowa State Statistical Laboratory, H. A. David and H. T. David, Eds., 1984 The Iowa State University Press, pp. 55–74.

[2] M. W. Garrett and W. Willinger, "Analysis, Modeling and Generation of Self-Similar VBR Video Traffic" *Proc. ACM SIGCOMM '94,* London, UK, August 1994, pp. 269–280.

[3] D. P. Heyman, A. Tabatabai, and T. V. Lakshman, "Statistical Analysis and Simulation Study of Video Teleconference Traffic in ATM Networks." *IEEE Transactions on Circuits and Systems for Video Techonology*, vol 2, no. 1, March 1992, pp. 49-59.

[4] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, "On the Self-Similar Nature of Ethernet Traffic (Extended Version)," *IEEE/ACM Trans. Networking,* vol. 2, no. 1, February 1994, pp. 1–15.

[5] W. Willinger, D. V. Wilson, W. E. Leland, and M. S. Taqqu, "On Traffic Measurements that Defy Traffic Models (and vice versa): Self-Similar Traffic Modeling for High-Speed Networks," *ConneXions*, vol. 8, no. 11, November 1994, pp. 14–24.