Traffic Characterization and Time Scales for Designing Efficient Network Control Policies

Ljiljana Trajković * and Arnie Neidhardt **

∗ *EECS Department, University of California, Berkeley, CA 94720-1770, ljilja@eecs.berkeley.edu*
∗∗ *Bellcore, 331 Newman Springs Road, Red Bank, NJ 07701, arnie@bellcore.com*

Abstract— Knowing traffic characteristics in packet networks is important for designing algorithms for managing and controlling packet networks. We demonstrate that for realistic traffic traces knowing few elementary traffic characteristics can substantially improve network utilization. Nevertheless, this improvement is often modest in comparison with the improvement that could be achieved if traffic characterization on more than one time scale were available.

I. INTRODUCTION

Packet networks are used to transmit traffic generated by applications that often generate complex, bursty, traffic streams. These streams may carry data, voice, image, or video information. Characterizing such traffic streams requires knowing the statistical properties of packet arrivals in time intervals of various durations called *time scales*. Knowing traffic on a time scale s means knowing the marginal distribution of the arrivals in a time interval of length s.

Researchers have adopted various approaches to designing Connection Admission Control (CAC) algorithms [4]. They include algorithms based on the notion of effective bandwidth [5], [10], [14], a decision-theoretic approach [8], and service-provisioning algorithms based on price-adjusting schemes [16]. Certain schemes were designed for bursty traffic and in-call re-negotiation [1], [12], [20], or to explore statistical multiplexing of voice and data [17] and of video sources [6], the use of a neural-networks paradigm for on-line prediction [2], and real-time traffic measurements [13]. One such optimal re-negotiating scheme is based on large-deviations theory [19] and takes into account traffic structure on multiple time scales [9].

In this study we illustrate the importance of traffic knowledge on more than one time scale in designing CAC policies in packet networks. The difference in the number of connections that a CAC policy admits (a relatively easy calculation) and the number of connections that the ideal policy $CACname = optsb$ would admit based on complete knowledge of joint distributions (the difficult task often approximated with heuristic approach or achieved via simulations) can be attributed to the additional traffic knowledge. Thus, the benefit of knowing traffic on more than one time scale is in the possible improvement of the CAC policy that is based on a single time scale marginal distribution. This study investigates the magnitude of this improvement.

II. TRAFFIC CHARACTERIZATION

A natural approach to describe traffic is to measure the random amounts of traffic generated in various time intervals [3], [7], [15], [21]. A traffic process is characterized by the family of random variables $A([s,t))$, representing the amount of traffic arriving in the interval $[s,t)$. An ideal traffic characterization would identify the joint distribution of these random variables. Traffic models often prescribe such joint distributions fully, but they are usually based on matching the observations of only a few features of the genuine traffic. We assume that the traffic-generation processes are stationary, i.e., $A_r([s,t)) = A([s+r, t+r))$. In particular, the amount of arrivals in any interval of length s has the same distribution as $A(s) = A([s,0))$. Thus, in our study, traffic knowledge is a description of the joint distribution of all the random variables in $\{A(s) | 0 \le s < \infty\}$. In contrast, traffic knowledge on a single time scale s refers to the distribution of only one of these random variables.

For the comparison of various CAC policies we used genuine traffic traces: a video-conferencing traffic trace [11] shown in Figure 1, and a data traffic trace collected from a HIPPI network [18].

III. CONNECTION ADMISSION CONTROL POLICIES

A network admission control policy should take traffic knowledge into account, while accounting for network capacities and the QoS standard to be met. Our service standard is that for any admitted connection:

$$P(\text{loss during a service-monitoring interval } D) \le \epsilon,$$

where the lost traffic refers to the given connection. Since the issue here is congestion, the only traffic losses that are relevant are those from queue overflows.

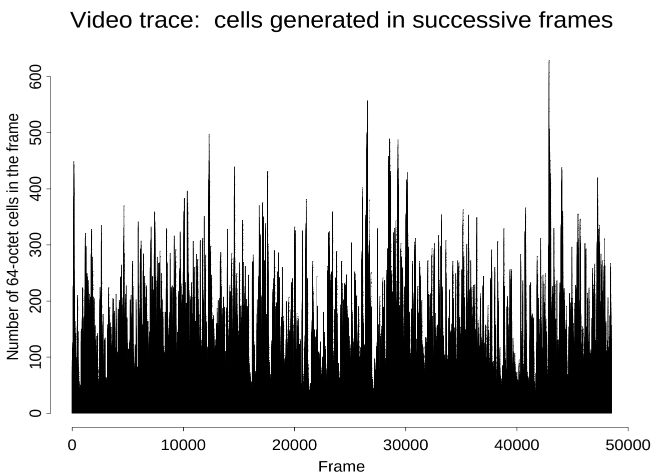

Figure 1: Traffic pattern of the video trace (time scale = 1/24 second). The trace lists the number of 64-octet (64-byte) cells in each of approximately 48 500 frames. It corresponds to approximately 30 minutes of a video at 24 frames/second.

The link parameters are its speed C (cells/second) and its buffer size $B \approx 2Ch$, where h is approximately the half-buffer work time. A general admission policy will admit $Nmax$ simultaneous homogeneous sessions, where the policy's determination of $Nmax$ will be affected by five parameters C, D, ϵ, and h, and the details of *TrafficDescription* that the policy is "prepared to understand." We examined various $CACname$ policies that differed in how much detail of the traffic description was used to calculate $Nmax$.

We compared ten CAC policies shown in Table 1, reflecting various degrees of traffic knowledge. The first five CAC policies (Cases 1–5) are conservative (meeting the service standard) and take at least some advantage of the available traffic information. The remaining CAC policies (Cases 6–10) are not conservative (not meeting the service standard) and intended to estimate the behavior of a CAC policy with complete traffic information.

The admission policies we considered all correspond to variations on the following calculation:

$$
\begin{aligned}
& \Pr(\ \exists t \in [0,D), \text{ the queue overflows } B \text{ at time } t) \\
\le\ & \Pr\Big(\bigcup_{i=1}^{L} \{A([p_i, p_{i-1})) > ch\}\Big) && (1)\\
\le\ & L\ \Pr(A(h) > ch) && (2)\\
\le\ & L\ \mathrm{E}e^{r(A(h)-ch)} && (3)\\
=\ & L\ (\mathrm{E}e^{r(A_1(h)-ch/n)})^n && (4)\\
\le\ & L\ (\mathrm{E}e^{r(\bar{A}_1(h)-ch/n)})^n && (5)\\
\le\ & \epsilon, && (6)
\end{aligned}
$$

where:

ϵ is the tolerable overflow probability given by the service criterion.

D is the length of a service-monitoring interval given by the service criterion.

B is the known buffer size in cells.

C is the known link service rate.

n is the known number of sessions contributing traffic.

X, c, h, and r are arbitrary positive numbers that an admission controller may select in an attempt to satisfy the inequalities above. In particular:

- X is the length of a prolog $[-X, 0)$ preceding the service-monitoring interval $[0, D)$.
- c is slightly smaller than C.
- h is the time scale on which features of the traffic process affect the calculation. We choose it to be slightly smaller than the half-buffer work time.

$L = \lceil (X+D)/h \rceil$ is the number of intervals of length h required to cover $[-X, D)$, the service-monitoring interval together with the prolog.

$p_i = D - ih$ for each integer i, yielding points in a sequence that provides one way of identifying a partition into intervals of length h. The intervals $[p_i, p_{i-1})$ for $1 \le i \le L$ form a partition of $[p_L, D) \supset [-X, D)$.

$A([p_i, p_{i-1}))$ is the aggregate traffic arriving during $[p_i, p_{i-1})$ from all n streams, which is only partially known to the admission controller.

$A(h)$ is the similarly unknown aggregate traffic arriving during $[0, h)$.

$A_k(h)$ is the unknown traffic from stream k.

$\bar{A}_1(h)$ is a random variable whose distribution can be selected by the admission controller. This selection should reflect traffic knowledge in the sense that the controller cannot rely on the comparison (5) between the true distribution of $A_1(h)$ and the controller's selection of $\bar{A}_1(h)$ without some knowledge of the true distribution. The admission controller's traffic knowledge affects the calculation only through this choice of $\bar{A}_1(h)$ and the comparison (5). Hence, h is the only time scale at which traffic knowledge affects this calculation.

Calculation (1) always holds if X, c, and h satisfy elementary constraints: $c < C$, $2ch \le B$, and $(C-c)X \ge B$. The comparisons (2)–(4) always hold if $r \ge 0$. The inequality (5) may always be satisfied,

Connection Admission Control policies		
Case	*CACname*	*Traffic knowledge or its treatment*
1	*HIPPI*	physical link peak rate
2	*peak*	traffic peak rate
3	*PM*	traffic peak and mean rate
4	*PMV*	traffic peak, mean, and the variance
5	*true*	the true marginal distribution
6	*normLT*	a normal approximation (via Laplace Transform)
7	*normPB*	a normal approximation (with probabilities)
8	*optsb*	optimistic, using a small buffer size
9	*opt/2*	optimistic, with a factor of 2
10	*vopt*	very optimistic

Table 1: Ten CAC policies used in comparison, all relying on traffic knowledge on the half-buffer time scale.

but the selection of $\bar{A}_1(h)$ must account for the limited traffic information available. Accordingly, in our selections various choices of $\bar{A}_1(h)$ will correspond to various CAC policies. The policy $CACname = true$ is obtained from (1)–(6) by choosing $\bar{A}_1(h) = A_1(h)$. For the policy $CACname = PMV$, $\bar{A}_1(h)$ is chosen to have the distribution Y that maximizes the expression Ee^{rY} among distributions with peak p, mean m, and variance v. Finally, the inequality (6) is not expected to hold for all values of n. Instead, it determines $Nmax$ as the largest value of n for which the inequality holds. The remaining policies (Cases 6–10) are not conservative because not all inequalities (1)–(6) are preserved.

IV. EFFICIENCY OF CAC POLICIES

We have examined efficiencies (occupancies) ρ of a link when it is "filled" with sessions admitted according to various CAC policies, where

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

An interesting example is shown in Figure 2. The values $D = 180$ seconds and $\epsilon = 0.01$ were inspired by a traditional standard of 1% (0.01) blocking for ordinary phone calls that last for an average of 3 minutes (180 seconds). As expected, greater efficiency is achieved with larger link speed C and with CAC policies employing greater traffic knowledge.

For a video-conference trace, respectable multiplexing gains can be achieved on a relatively low-speed OC-6 link (162 times the trace's mean rate), with a small 2-millisecond buffer (1,418 cells), and with policy $CACname = PMV$. This CAC policy admits 109 connections (67% occupancy). The knowledge of variance is important, as shown by the fact that $CACname = PM$ yields the much lower occupancy of 61 connections (38% occupancy). The more knowledgeable policy $CACname = true$ allows 120 connections (slightly greater occupancy of 74%).

In contrast, to achieve noticeable multiplexing gains with the data traffic, the higher-capacity OC-24 link was required (937 times the trace's mean rate) with a far larger 2-second buffer (5,608,031 cells). Even then, the simulations indicated that the ideal CAC policy would admit only 517 connections (55 % occupancy). In other words, the data traffic is tremendously bursty and even an ideal CAC policy cannot handle it as efficiently as the video traffic. Moreover, even the best among the conservative CAC policies based on a single time scale $CACname = true$ admits only 195 connections (21% occupancy). In other words, when dealing with bursty traffic, it is imperative to understand traffic on more than one time scale.

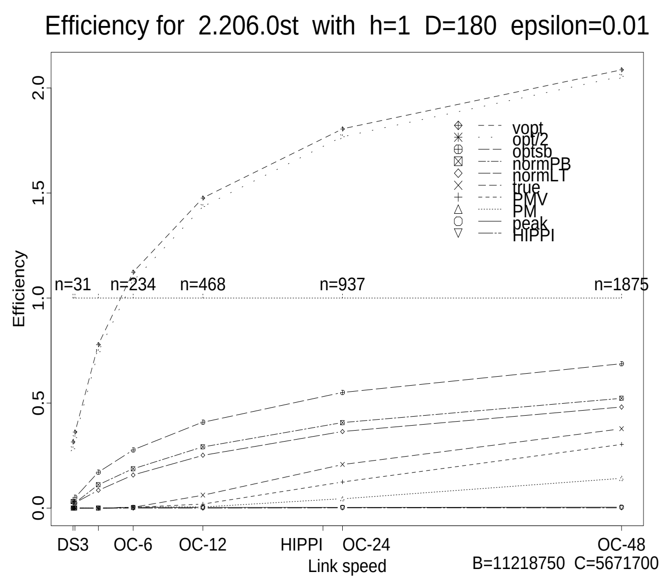

Figure 2: Efficiency vs. link speed for data traffic. Each curve corresponds to a separate CAC policy. The conservative CAC policies all yield rather small efficiencies well below both the ideal efficiency of 100% and the efficiencies produced by the nonconservative CAC policies.

V. RELEVANT TIME SCALES

Having concluded that significantly greater efficiency can be obtained only with traffic information on more than one time scale, the question now becomes, "What other time scales are most relevant?" 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 in 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 on some time scales between 10 seconds and 6 minutes, including at least one time scale in the range of 10–20 seconds. In particular, it is 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).

VI. CONCLUDING REMARKS

We investigated the effectiveness of incorporating various traffic-information details into the design of CAC policies. We found that, compared to a peak-rate-based policy, substantial efficiency could be gained by the additional knowledge of the mean and the variance on the time scale of the half-buffer work time. For some types of realistic traffic, such as the video-conference trace, the efficiency produced by this elementary knowledge of the traffic peak, mean, and variance on a single time scale, was quite satisfying. Yet, for other traffic types, such as the data trace, this simple single-time-scale information can yield some disappointing efficiencies. Indeed, for such traffic trace, an efficient policy would have to employ traffic information on multiple time scales that range over more than an order of magnitude.

References

[1] P. E. Boyer and D. P. Tranchier, *Computer Networks and ISDN Systems*, vol. 24, no. 4, pp. 321–334, May 1992.

[2] S. Chong, S. Q. Li, and J. Ghosh, *IEEE J. Select. Areas Commun.*, vol. 13, no. 1, pp. 12–23, Jan. 1995.

[3] D. R. Cox, *Statistics and Appraisal, Proc. 50th Anniversary Conf.*, Iowa State Statistical Laboratory, H. A. David and H. T. David, Eds., The Iowa State University Press, 1984, pp. 55–74.

[4] J. Curose, *Computer Communication Reviews*," vol. 23, no. 1, pp. 6–15, Jan. 1993.

[5] A. Elwalid and D. Mitra, *IEEE/ACM Transactions on Networking*. vol. 1, no. 3, pp. 329–343, June 1993.

[6] A. Elwalid, D. Heyman, T. V. Lakshman, and D. Mitra, *IEEE J. Select. Areas Commun.*, vol. 13, no. 6, pp. 1004–1016, Aug. 1995.

[7] M. W. Garrett and W. Willinger, *Proc. ACM SIGCOMM '94,* London, UK, Aug. 1994, pp. 269–280.

[8] R. J. Gibbens, F. P. Kelly, and P. B. Key, *IEEE J. Select. Areas Commun.*, vol. 13, no. 6, pp. 1101–1114, Aug. 1995.

[9] M. Grossglauser, S. Keshav, and D. Tse, *Proc. ACM SIGCOMM '95,* Cambridge, MA, Sept. 1995, pp. 219–230.

[10] R. Guerin, H. Ahmadi, and M. Nagshineh, *IEEE J. Select. Areas Commun.*, vol. 9, no. 7, pp. 968–981, Sept. 1991.

[11] D. P. Heyman, A. Tabatabai, and T. V. Lakshman, *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 2, no. 1, pp. 49–59, Mar. 1992.

[12] J. Hui, *IEEE J. Select. Areas Commun.*, vol. 6, no. 9, pp. 1598–1608, Dec. 1988.

[13] S. Jamin, P. Danzig, S. Shenker, and L. Zhang, *IEEE/ACM Trans. Networking*, vol 5, no. 1, pp. 56–70, Feb. 1997.

[14] G. Kesidis, J. Walrand, and C.-C. Chang, *IEEE/ACM Trans. Networking*, vol. 1, no. 4, pp. 424–428, Aug. 1993.

[15] 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, pp. 1–15, Feb. 1994.

[16] S. H. Low and P. P. Varaiya, *IEEE/ACM Trans. Networking*, vol. 1 no. 5, pp. 547–553, Oct. 1993.

[17] K. Sriram, *Computer Networks and ISDN Systems*, vol. 26, pp. 43–59, Sept. 1993.

[18] D. Tolmie and J. Renwick, *IEEE Network Magazine*, vol. 7, no. 1, pp. 28–32, Jan. 1993.

[19] D. Tse, R. G. Gallager, and J. N. Tsitsiklis, *IEEE J. Select. Areas Commun.*, vol. 13, no. 6, pp. 1028–1038, Aug. 1995.

[20] J. S. Turner, *IEEE Network Magazine*, vol. 6, no. 5, pp. 50–58, Sept. 1992.

[21] W. Willinger, D. V. Wilson, W. E. Leland, and M. S. Taqqu, *ConneXions*, vol. 8, no. 11, pp. 14–24, Nov. 1994.