The Pros and Cons of a Job Buffer in a Token-Bank Rate-Control Throttle

Arthur W. Berger and Ward Whitt

Abstract— **Rate-control throttles with token banks or leaky buckets have been used for overload control in telecommunication systems and have been recommended for traffic policing in Broadband Integrated Services Digital Networks (B-ISDN's). Enhancing the token-bank throttle with a buffer to shape the admitted traffic has been suggested. Researchers have shown that the presence of the buffer can dramatically reduce the squared coefficient of variation of the interadmission time. However, we show that the impact of the buffer on longer-time-scale characteristics of the admitted traffic is much less dramatic. In particular, we show (primarily through simulations) that the job buffer has much less impact on higher values of the index of dispersion for intervals and on small tail probabilities for the steady-state number in system at a downstream queue (with only this one arrival stream). Indeed, the smoothing benefit of the job buffer decreases as longer-time-scale characteristics become more important. However, if the downstream queue is fed by many sources with throttles, as would be the case in most applications, then the relevant time scale at the downstream queue indeed becomes relatively short. Our simulation results show that then the benefit of traffic shaping can be much greater. The benefit gained in reduced buffer requirements at the downstream queue, though, is typically significantly less than the sum of all job buffers added to the throttles. A full cost/benefit analysis depends on the relative cost of buffer space in the two places and on details of the relevant application.**

Keywords—**rate-control throttles, leaky buckets, traffic policing, traffic shaping, buffers, B-ISDN, ATM, overload control**

I. Introduction

Rate-control throttles based on token banks have been used for the regulation of call-setup requests in telecommunication switching systems [1] and have been proposed for policing in Broadband Integrated Services Digital Networks (B-ISDNs) using Asynchronous Transfer Mode (ATM) [2]–[5]. Researchers have examined the addition of a buffer in which jobs queue when no token is available [6]–[12]. The operation of such a throttle is illustrated in Fig. 1. The throttle contains two finite buffers, one for tokens and one for jobs, where the jobs may be cells in a B-ISDN/ATM, packets in a general high-speed packet network, or call-setup requests to a telecommunications switching system. The tokens arrive deterministically and evenly spaced from an infinite source. Tokens that arrive to a full bank are blocked and lost. (The token bank can be implemented by a counter with a cap.) If the bank contains a token when a job arrives to the throttle, then the job is allowed to pass through, and the bank is decremented by one token. If the bank does not contain a token when a job arrives, then the job queues in the job buffer if the buffer is not full. If the job arrives to a full buffer, then the job is said to have "overflowed." In packet networks, the overflowed frame or cell may be discarded or may be marked and later treated as a lower priority class. (If marking is used and if job sequence is to be maintained, then a job arriving to a full job buffer could be queued and the job at the head of the queue could be marked and admitted.) Herein, our primary interest is in the non-overflow jobs, and for simplicity we assume the overflow jobs are discarded.

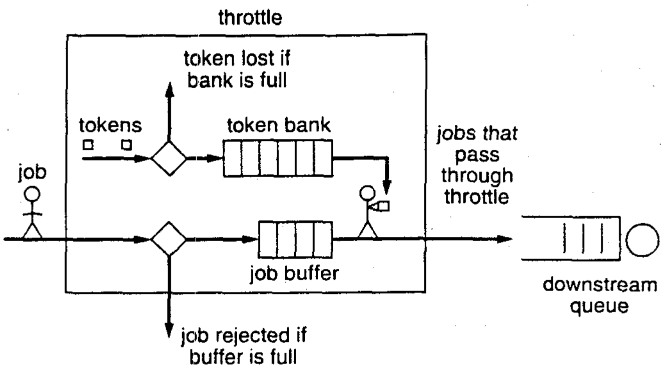
throttle
token lost if bank is full
tokens
token bank
jobs that pass through throttle
job
job buffer
downstream queue
job rejected if buffer is full

Fig. 1. Diagram of token-bank, job-buffer throttle and downstream queue.

The primary reason for the job buffer is traffic shaping (or smoothing) in order to reduce subsequent congestion experienced by jobs admitted by the throttle and by other jobs sharing the same subsequent resources. Our question is: *How effective is the job buffer, first, in shaping the traffic and, second, in reducing congestion?*

The following result from [9] and [13] helps put this question in proper perspective:

Overflow Invariance Property: Except for a finite initial period to account for initial conditions, the

Paper approved by Moshe Sidi, the Editor for Communication Networks of the IEEE Communications Society. Manuscript received May 1, 1992; revised February 24, 1993. The authors are with AT&T Bell Laboratories; the remaining postal addresses are, respectively, Room 1K-211, Holmdel, NJ 07733-3030 and Room 2C-178, Murray Hill, NJ 07974-0636.

IEEE Log Number 9400977.

> *job overflow process depends on the (finite) capacity of the token bank, C_T, and the (finite) capacity of the job buffer, C_J, only via their sum $C_T + C_J$.*

The overflow invariance property implies that we can decompose the traffic shaping by the rate-control throttle into two separate parts: First, there is the shaping caused by job rejections, which depends only on the total capacity $C_T + C_J$. Second, there is the traffic shaping provided by the job buffer given a fixed total capacity $C_T + C_J$. It is this second effect that we primarily want to understand: *For given total capacity $C_T + C_J$, what are the costs and benefits of including a job buffer of capacity C_J?* We begin to investigate this question in [13] wherein we prove sample-path inequalities and limits under very general conditions. In this letter, we continue the investigation, via simulation experiments, and illustrate the magnitude of phenomena and convergence rates for typical cases of interest. Additional results from the simulations are reported in [14].

II. The Benefits of Extra Delays

Theorem 5.1 of [13] shows that increasing C_J for fixed $C_T + C_J$ actually increases the time spent in the throttle by each admitted job (again modulo the initial conditions). A very general question, then, is: *What are the pros and cons of adding extra delays to some jobs upon arrival before they enter a service system?*

For the case in which a single arrival process comes to a single-server queue with unlimited waiting space and the first-in first-out (FIFO) discipline, it is easy to see *each job's total time in system necessarily increases if we reshape the arrival process by introducing delays in any way prior to admission to the queue;* see Theorem 12 of [15] and p. 358 of [16].

Nevertheless, the extra delays may be worthwhile because time spent in the subsequent service system may be much more costly than time spent waiting externally. Then we might be content to know that judicious delays added upstream would reduce the delays downstream [17], [18]. There also is the issue of multiple classes and fairness. Adding delays to one class of jobs may reduce delays for another class of jobs. Fairness obviously is an important aspect, but we do not consider it here.

III. Traffic Shaping by a Job Buffer

It is easiest to contrast an all-job-buffer throttle ($C_T = 0$) with an all-token-bank throttle ($C_J = 0$), assuming that $C_T + C_J$ is fixed. In the all-job-buffer throttle, the admission epochs of successive jobs are always separated by at least the deterministic interval between successive token arrivals, whereas in the all-token-bank throttle the jobs can be admitted in batches equal to the token bank capacity C_T. Hence, it is evident that the job buffer has the potential for traffic shaping.

Assuming Poisson job arrivals, Sidi et al. [7] showed that a token-bank throttle with an infinite-capacity job buffer can smooth the admitted stream, in the sense of reducing the squared coefficient of variation (SCV) of the interdeparture (interadmission) times, c_d^2. This effect also occurs when the arrival process is a Markov-modulated fluid source, as demonstrated in [12]. Herein, we examine the admission process in greater detail, and as a first step we estimate the *index of dispersion for intervals* (IDI) of the job admission process, [19, pp. 70-72] and [20, §III], assuming a bursty on-off renewal process with parameters chosen to give a rough representation of current local-area-network (LAN-to-LAN) traffic on contemplated access links of a B-ISDN/ATM (with mean number of cells per burst ranging from 10 to 100). Let $\{D_n, n \geq 1\}$ be the sequence of job interadmission times from the throttle. Assume that $\{D_n, n \geq 1\}$ is stationary, i.e., the joint distribution of $\{D_n, D_{n+1}, \ldots, D_{n+m}\}$ is independent of n for all m. Let $S_n = D_1 + \ldots + D_n$. The IDI is the sequence $\{IDI(n),\ n \geq 1\}$ defined by: $IDI(n) = n\text{Var}(S_n)/[E(S_n)]^2$, which equals the variance of S_n normalized by what the variance would have been if the process were Poisson.

Our main conclusion about the influence of the job buffer on the IDI is that the impact on $IDI(n)$ decreases as n increases. Fig. 2 plots simulation estimates of IDI(n) for n from 1 to 1,000 for three cases: (A) $C_T = 50$, $C_J = 0$; (B) $C_T = C_J = 25$; and (C) $C_T = 0$, $C_J = 50$, where the job arrival process has a mean rate of 3 (jobs/ms), has a geometrically distributed number of jobs in a burst with a mean of 10 and a spacing of 0.05 ms, and has an exponentially distributed off-period, and where the tokens arrive equally spaced at a rate of 4 (tokens/ms). Additional details of the simulation are given in [14]. Also shown is the IDI of the arrival process to the throttle. The arrival-process IDI is constant and equal to c_a^2 since the arrival process is renewal. *In summary, the traffic smoothing by the job buffer is less than one would expect from looking only at the interadmission-time SCV $c_d^2 \equiv IDI(1)$.* Indeed, Theorem 5.2 of [13] shows that IDI(∞) is unaffected by the job buffer. Moreover, from Fig. 2 we have an illustration of the rate at which the three sequences of IDI's approach their common limit. In particular, for $n = 1$, the ratio of the IDI of the admission process from the all-token-bank throttle (case A) to that from the all-job-buffer throttle (case C) is substantial, 2.9, and for $n = 10$ is still substantial, 2.4, though when $n = 100$, the ratio has declined significantly, to 1.3, and for $n = 1{,}000$ is 1.03. Similar results are obtained for other example job-arrival processes [14]. Expressed differently, we conclude that the traffic smoothing depends on the time scale [20]–[22]. We conclude that *the job buffer smooths the traffic dramatically in a short time scale, but much less so in a long time scale.*

IV. The Impact on a Downstream Queue

Of course, the IDI is not of much importance in itself. Reduced values are only important as indications that the admitted stream will be easier for the remaining system to handle.

Theorem 5.4 of [13] shows that *the heavy-traffic limiting behavior of a downstream queue (having a single-server queue, the first-come first-served discipline, unlimited waiting space and deterministic service times) as traffic inten-*

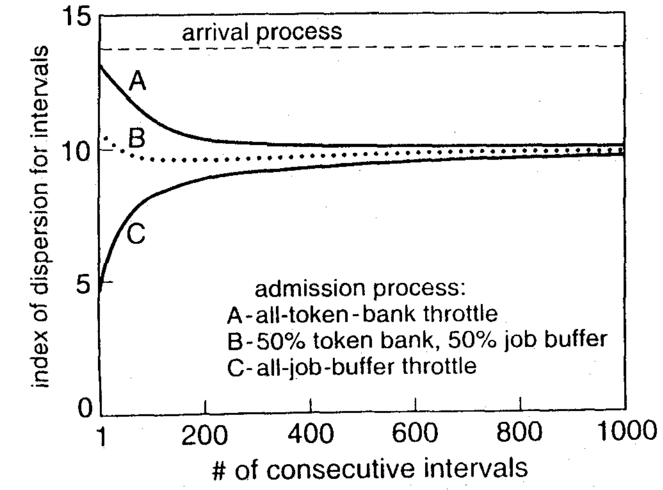

Fig. 2. Index of dispersion for intervals for the arrival and admission job processes: base-case parameter values.

sity approaches its critical value from below by increasing the service times is independent of the job buffer. To see the impact of the job buffer on a downstream queue at non-critical loads, we simulate a downstream queue with an arrival process consisting of the stream of admitted jobs from the rate-control throttle. (Here we consider only *one* throttle feeding the downstream queue; in §V we consider more than one.)

We choose the (deterministic) service times of the downstream queue so that the server occupancy is around 85%–95%. However, we are *not* suggesting that a real B-ISDN/ ATM device would likely operate at such an *average* load. In B-ISDN/ATM we are interested in extremely small overflow probabilities from finite buffers. Such overflows would occur during times of relatively high congestion, periods over a time scale of milliseconds to seconds when the load is high. Thus, when the downstream queue is viewed as a model of a real buffer in a B-ISDN/ATM device, we are considering the above congestion situation and have modeled it via the tail distribution of the steady-state number in an infinite-capacity queue at high occupancy.

We focus on the tail probabilities of the steady-state number, Q, in the downstream queue (number in system at an arbitrary time); i.e., we look at the levels n_k such that $P(Q \geq n_k) = 10^{-k}$ as a function of k. Consistent with the conclusion about the IDI above, we find that the job buffer has a greater impact for small k than for large k. For the same job and token arrival processes as in Fig. 2, for $C_J + C_T$ again equal to 50 and for a server occupancy of 0.86, going from the all-token-bank throttle to the all-job-buffer throttle reduced n_1 from 70 to 48 but reduced n_4 only from 225 to 201. We thus conclude that *a job buffer in a rate-control throttle would have relatively little benefit in reducing the buffer size required at a downstream queue (fed by only one stream) sized according to a criterion of a very small blocking probability.* An exception though, is when the deterministic service time at the downstream queue is less than the token interarrival time. Then, with the all-job-buffer throttle, the minimum interarrival time at the downstream queue is greater than the service time and the maximum number in the downstream queue is just one. In contrast, with the all-token-bank throttle, the interarrival time at the downstream queue can still be less than the service time, and queues "significantly" greater than one (on a percentage basis) can occur. However, the reduction in number in system at the downstream queue due to the change from an all-token-bank throttle to an all-job-buffer throttle is bounded by $C_J + C_T$. This bound is proved in the Appendix of [14]. Moreover, there we give a (somewhat pathological) example in which the steady-state number in the downstream queue is *stochastically larger with an all-job-buffer throttle than with an all-token-bank throttle.*

V. Multiple Sources

The analysis above is based on traffic from a single source. Most systems, however, would actually have traffic from many throttles entering the downstream system. In order for a downstream server to handle traffic from n nearly-equal-rate sources, it must have a service rate approximately n times faster than it would if there were only one source. With many sources, the mean interarrival time at the downstream queue must be much shorter than the mean interadmission time from a single throttle. Consequently, with many sources, the relevant time scale at the downstream queue is much shorter than it would be with only one source. In other words, as in [22], what is a short time scale for the source (e.g., a few interadmission times) will likely be a long time scale for the downstream queue. Consistent with this observation, *our simulation results show that the job buffer provides a dramatic smoothing benefit when 100 identical independent throttles feed the downstream queue.* In particular, with 100 sources, each with the same parameters as in §IV, going from an all-token-bank throttle to an all-job-buffer throttle reduces IDI (1000) from 12.6 to 2.5, reduces n_4 from 350 to 32, and reduces the maximum number in the queue during the simulation from 400 to 58. Again, similar results are obtained for other example arrival processes, [14].

We discovered another significant effect associated with multiple sources that we had not anticipated, but which is intuitively clear upon reflection. *With multiple sources, the congestion at the downstream queue is significantly affected by synchronization of the deterministic token arrival streams.* In particular, if multiple throttles have identical token rates and if the token arrival times are synchronized, then the delays at the downstream queue are much greater than if they are not synchronized, i.e., if they are staggered or chosen randomly. We assume that this synchronization effect is not present.

In §II we noted that, even though the addition of the job buffer tends to reduce the delay at the downstream queue, it necessarily increases the total delay for each job. For the previous example with 100 sources, the total mean delay is 4.45 ms with all-job-buffer throttles, but is only 0.06 ms with all-token-bank throttles.

VI. Comparison of Buffer Capacities

In order to achieve the significant smoothing benefit at the downstream queue with multiple sources, we must add a job buffer to each throttle. *Our simulation results show that the reduction in required buffer space at the downstream queue is far less than the sum of the job buffer capacities added to all the throttles.*

This comparison will be of little consequence if buffer capacity at the downstream queue is much more costly than the job buffers, but in some contexts the cost of the buffers may be comparable. For example, in a B-ISDN/ATM switching system, the job-buffer throttle may be in access line cards and the downstream queue herein may represent an output-port buffer also on a line card, in which case the buffer costs may be comparable. In any case, the relative buffer costs are clearly application and architecture dependent, and thus herein we simply compare the reduction in the number of buffer spaces required at the downstream queue with the number of added buffer spaces in the throttles. (This comparison is readily translated to dollars given the relevant buffer costs.)

For the 100-source example in §V, the reduction of maximum queue occupancy from 400 to 58 was achieved at the expense of adding 5000 (i.e., 100 × 50) job buffer spaces in the throttles. By these figures, downstream buffer space would thus have to be more than 15 times as expensive as job buffer space to prefer job buffers. As another example, if the individual job-arrival processes had a mean burst size of 100 and a spacing during the burst of 3μsecs. (roughly corresponding to back-to-back ATM cells on a 150 Mbps link) and if $C_J + C_T = 1,000$, then the reduction in maximum queue occupancy observed over a simulation run is from 5,510 to 485, though at the expense of adding 100,000 job-buffer spaces in the throttles.

Theorem 6.1 of [13] also provides a significant theoretical reference point in the case of a finite-capacity downstream queue. It states that *under very general conditions the throughput is greater when job buffers associated with all-job-buffer throttles are removed (holding $C_T + C_J$ fixed) from some of the throttles feeding a finite-capacity downstream queue and the eliminated job-buffer capacity is added to the downstream queue.* In other words, a fixed buffer capacity is more efficiently used at the downstream queue. This result is consistent with intuition, expressing the well known advantage of statistical multiplexing.

Of course, the choice may well not be all or none. Contrary to what one might have hoped, *our simulation results also show that a disproportionate portion of the benefit of job buffers comes from the final job buffer capacity;* e.g., having a proportion p of the total throttle capacity as job buffer provides *less* than a proportion p of the potential shaping benefit of the all-job-buffer throttle (where "benefit" is measured in terms of reduction in mean number or tail number, n_k, in the downstream queue) [14].

VII. Conclusions

For an archetype model where a-priori exogenous job arrival processes are regulated by token-bank throttles and where the admitted processes enter a downstream queue, we have examined (in greater detail than the SCV of the inter-admission times) the impact of including a job buffer in the throttles.

In summary, the traffic-shaping benefit of job buffers in token-bank rate-control throttles depends on several factors, one being the time scale. A job buffer smooths the traffic more in a short time scale than a long one. The relevant time scale in turn depends on the system, e.g., if a downstream queue is fed by a few sources or many. More sources means a smaller time scale, and a smaller time scale means more smoothing benefit. Even with many sources, a proper cost-benefit tradeoff depends on the relative cost of buffer capacity in the throttle versus downstream. If the cost of job buffer space is negligible compared to the cost of downstream buffer space and if there are many sources, then significant traffic-shaping benefits can be obtained from job buffers. More specific judgments of the efficacy of including a job buffer depend on details of the application and are not captured by the archetype model herein. For example, in the application to B-ISDN/ATM, where a job arrival process corresponds to a cell flow of an ATM connection and where the throttles correspond to policing mechanisms in the switching systems of a public ATM network, there may be opportunity to share the job-buffer space across multiple sources, particularly if multiple sources happen to be on the same access link.

Note that in the application to B-ISDN/ATM, the throttles have the important function of protecting the network and other users' traffic from malicious users and sputtering terminals. The addition of a job buffer in the throttle is not needed to limit this excess traffic since it does not impact the portion of a user's traffic detected as excessive, for given total $C_J + C_T$. However, a possible concern is that a group of k malicious users could simultaneously submit bursts of $C_J + C_T$ cells, which the all-token-bank throttles would admit as bursts and which the all-job-buffer throttles would space out before admitting. However, for the all-token-bank throttles, the buffer space downstream needed to avoid cell loss (assuming cell loss is to be avoided even in this extreme case) is bounded by $k \cdot (C_J + C_T)$, which is the amount of buffer space used in the all-job-buffer throttles.

References

[1] B. T. Doshi and H. Heffes, "Analysis of overload control schemes for a class of distributed switching machines," *Proceedings Tenth Int. Teletraffic Congress,* Montreal, Canada, Paper No. 5.2.2, 1983.

[2] J. Turner, "New directions in communications (or which way to the information age?)," *IEEE Communications Magazine,* vol. 24, pp. 8-15, 1986.

[3] A. E. Eckberg, D. T. Luan and D. M. Lucantoni, "Bandwidth management: A congestion control strategy for broad-

band packet networks, characterizing the throughput-burstiness filter," *Int. Teletraffic Congress Specialist Seminar,* Adelaide, Australia, Paper No. 4.4, September 1989.

[4] E. P. Rathgeb, "Modeling and performance comparison of policing mechanisms for ATM networks," *IEEE J. of Selected Areas in Communications,* vol. 9, pp. 325-334, 1991.

[5] F. Guillemin, P. Boyer, A. Dupuis, and L. Romoeuf, "Peak rate enforcement in ATM networks," *INFOCOM '92,* Florence, Italy, pp. 753-758, 1992.

[6] I. Cidon and I. S. Gopal "PARIS: An approach to integrated high-speed private networks," *Int. J. of Digital and Analog Cabled Systems,* vol. 1, pp. 77-86, 1988.

[7] M. Sidi, W. Z. Liu, I. Cidon and I. Gopal, "Congestion control through input rate regulation," *GLOBECOM '89,* Dallas, Texas, pp. 1764-1768, November 1989.

[8] M. C. Chuah and R. L. Cruz, "Approximate analysis of average performance of (σ, ρ) regulators," *Proc. of IEEE INFOCOM,* San Francisco, CA, pp. 874-881, 1990.

[9] A. W. Berger, "Performance analysis of a rate-control throttle where tokens and jobs queue," *IEEE J. Sel. Areas Commun.,* vol. 9, pp. 165-170, 1991.

[10] K. C. Budka and D. D. Yao, "Monotonicity and convexity properties of rate control throttles," *29th IEEE Conf. on Decision and Control,* Honolulu, Hawaii, pp. 883-884, 1990.

[11] K. Sohraby and M. Sidi, "On the performance of bursty and correlated sources subject to leaky bucket rate-based access control schemes," *IEEE INFOCOM '91,* Bal Harbour, Florida, pp. 426-434, April 1991.

[12] A. Elwalid and D. Mitra, "Analysis and design of rate-based congestion control of high-speed networks, I: stochastic fluid models, access regulation," *Queueing systems,* vol. 9, pp. 29-63, 1991.

[13] A. W. Berger and W. Whitt, "The impact of a job buffer in a token-bank rate-control throttle," *Stochastic Models,* vol. 8, pp. 685-717, 1992.

[14] A. W. Berger and W. Whitt, "Traffic shaping by a job buffer in a token-bank rate-control throttle," AT&T Bell Laboratories, 1992.

[15] W. Whitt, "Comparing counting processes and queues," *Adv. Appl. Prob.,* vol. 13, pp. 207-220, 1981.

[16] S. Suresh and W. Whitt, "The heavy-traffic bottleneck phenomenon in open queueing networks," *Oper. Res. Letters,* vol. 9, pp. 355-362, 1990.

[17] S. Niu, "Bounds for the expected delays in some tandem queues," *J. Appl. Prob.,* vol. 17, pp. 831-838, 1980.

[18] B. S. Greenberg, "On Niu's conjecture for tandem queues," *Adv. Appl. Prob.,* vol. 19, pp. 751-753, 1987.

[19] D. R. Cox and P. A. W. Lewis, *The Statistical Analysis of Series of Events,* London: Methuen, 1966.

[20] K. W. Fendick and W. Whitt, "Measurements and approximations to describe the offered traffic and predict the average workload in a single-server queue," *Proceedings of the IEEE,* vol. 77, pp. 171-194, 1989.

[21] J. W. Roberts, "Traffic control in B-ISDN," *Int. Teletraffic Congress Seminar,* Cracow Poland, 1991.

[22] W. Whitt, "A light-traffic approximation for single-class departure processes from multi-class queues," *Management Sci.,* vol. 34, pp. 1333-1346, 1988.