Proceedings of the 29th Conference
on Decision and Control
Honolulu, Hawaii • December 1990

A MULTI-CLASS INPUT-REGULATION THROTTLE

Arthur W. Berger
Room 3H-601
AT&T Bell Laboratories
Holmdel, N.J. 07733

Ward Whitt
Room 2C-178
AT&T Bell Laboratories
Murray Hill, N.J. 07974-2070

ABSTRACT

We present an input-regulation throttle to allocate capacity among multiple classes of jobs, where the allocation is enforced only when the total offered load is beyond capacity. The design uses a rate control throttle with a dedicated token bank for each class and a single, shared overflow bank. The per-class blocking and throughput are computed via alternative, approximate analyses. For the important, special case of two priority classes of Poisson job arrivals, we present an exact analysis.

1. INTRODUCTION

In this paper we introduce and investigate a rate-control throttle for regulating the admission to a system of jobs of multiple classes. The jobs arrive randomly in an unknown and possibly nonstationary manner. We must decide, immediately upon arrival, whether or not to admit each job. We assume that the jobs that are not admitted are lost (do not enter the system) without affecting future arrivals.

We want a policy or mechanism for deciding whether or not to admit each job. We have several goals. First, we want to be able to guarantee a specified input rate to each class, which may differ from class to class. However, if the total offered load is less than system capacity, then we want all jobs admitted, regardless of class allocations. We want a simple control, which requires minimal monitoring of the arrival streams and minimal data processing. Also, it should not need real-time feedback from the system that the jobs enter. The control should be robust; e.g., over a range of offered load, it should perform well with constant parameter settings. Finally, it should respond relatively quickly to changes in the offered loads.

The proposed control is a rate-control throttle based on token banks. The token banks are counters that increment up periodically and decrement at job arrivals. A throttle based on token banks for a single class of jobs was proposed and analyzed by Doshi and Heffes [1]. More recently, the single-class token-bank throttle and the closely related leaky-bucket throttle have been investigated as potential policing mechanisms in broadband networks, e.g. Eckberg et al. [2] and Sidi et al. [3]. The purpose of the present paper is to introduce and analyze a token-bank throttle for multiple classes. We thank our colleague R. Milito for suggesting the use of the token-bank throttle for regulating the admission of multiple classes.

The throttle is defined as follows. Upon arrival of a class-i job, $i = 1,\ldots, N$, if bank i contains a token, then the job is admitted and the bank is decremented by 1 token. If bank i is empty, then the job overflows to a shared, overflow bank, bank 0, where it gets a second chance to be admitted. If bank 0 contains a token, then the job is admitted and bank 0 is decremented by 1 token. If both banks i and 0 are empty, then the job is rejected by the throttle. (Herein, we consider the job to be lost, though alternatively it could be queued or marked and admitted.) As for the tokens, class-i tokens arrive deterministically, evenly spaced, at a rate r_i, to bank i of finite capacity C_i. If bank i is full, then the token overflows to bank 0 of capacity C_0. If both banks i and 0 are full, then the token is dropped and lost.

The dedicated banks provide the allocation for each class, and the overflow bank provides a sharing of excess capacity. The decision maker chooses the parameters: $\{r_1, \ldots, r_N\}$ and $\{C_0, C_1, \ldots, C_N\}$. The r_i's determine the maximum, sustained, admission rate for each class. The C_i's limit the instantaneous burst of arrivals that may be admitted. Given the stochastic nature of the job arrival processes, both the token rates and the bank capacities influence the steady-state per-class blocking and throughput, as well as the transient response. This paper begins the analysis of this input-regulation throttle by computing the per-class blocking and throughput as a function of the design parameters, assuming stationary job processes.

2. SPECIAL CASE OF TWO CLASSES WITH PRIORITIES: EXACT ANALYSIS

This section considers the important, special case in which one class (or group of classes) is to be given preference (priority) over another class. The input regulation throttle can give priority to class 1 by setting the token arrival rate at bank 1 equal to the desired maximum sustained admittance rate from both classes and setting the token arrival rate at bank 2 equal to zero. Thus, class 2 jobs are admitted only via tokens from class 1 that overflow to bank 0.

We assume in this section that the job arrival processes are Poisson. Let $X^i(t)$ be the number of tokens in bank i at time t, $i = 0,1$. Let T_n be the epoch of arrival of the n^{th} token. Lastly, let $X_n^i = X^i(T_n^-)$, the number of tokens seen in bank i by the n^{th} token arrival, i=0,1. Given Poisson job arrivals, $\{X_n^0, X_n^1\}$ forms an embedded Markov chain.

CH2917-3/90/0000-2106$1.00 © 1990 IEEE

Define $P[(i,j),(k,l),x]$ = Prob($X^0(T_n+x) = k$ and $X^1(T_n+x) = l$, given that $X_n^0 = i$ and $X_n^1 = j$ and $0 < x \le 1/r_1$). When $x = 1/r_1$, we interpret $T_n + x$ as T_n^-, in which case, $P[(i,j),(k,l),x]$ is an element of the transition matrix of the embedded Markov chain $\{X_n^0, X_n^1\}$. (For x less than $1/r_1$, $P[(i,j),(k,l),x]$ is used in the calculation of the number in system at arbitrary time epochs, equation (2).) The following expressions for $P[(i,j),(k,l),x]$ are given in terms of: $A^i(n,x)$ = Prob(n jobs of class i arrive to the throttle over an interval of length x) and $B^i(n,x)$ = Prob(at least n jobs of class i arrive to the throttle over an interval of length x), i=1,2. $A^i(n,x)$ and $B^i(n,x)$ are respectively point mass and tail probabilities from Poisson distributions. The derivation follows from the definition of the throttle and the assumption of Poisson job arrivals. The details are tedious and are omitted. For brevity, the argument x is suppressed in $A^i(n,x)$, $B^i(n,x)$ and $P[(i,j),(k,l),x]$.

For $i = k = 0$: $P[(i,j),(k,l)] =$ (1a)

$$A^1(\min(j+1,C_1)-l)\cdot B^2((j+1-C_1)^+) + B^1(\min(j+1,C_1)+1)\cdot 1_l\{0\}$$

For $i = 1,\ldots, C_0-1$, $k = 0$: $P[(i,j),(k,l)] =$ (1b)

$$A^1(\min(j+1,C_1)-l)\cdot B^2(i+(j+1-C_1)^+) + [B^1(j+1+i) + \sum_{m=1}^{i-1+(j+1-C_1)^+} A^1(j+1+i-m)\cdot B^2(m)]\cdot 1_l\{0\}$$

For $i = C_0$, $k = 0$: $P[(i,j),(k,l)] =$ (1c)

$$A^1(\min(j+1,C_1)-l)\cdot B^2(C_0) + [B^1(\min(j+1,C_1)+C_0) + \sum_{m=1}^{C_0-1} A^1(\min(j+1,C_1)+C_0-m)\cdot B^2(m)]\cdot 1_l\{0\}$$

For i=0,...,C_0−1, k=1,...,i+1, let $n = i - k$: (1d)

$$P[(i,j),(k,l)] = A^1(\min(j+1,C_1)-l)\cdot A^2(n+(j+1-C_1)^+) + [\sum_{m=1}^{n-1+(j+1-C_1)^+} A^1(j+1+n-m)\cdot A^2(m)]\cdot 1_l\{0\}$$

For $i = C_0$, k=1,...,C_0: $P[(i,j),(k,l)] =$ (1e)

$$A^1(\min(j+1,C_1)-l)\cdot A^2(C_0-k) + [\sum_{m=0}^{C_0-k-1} A^1(\min(j+1,C_1)+C_0-k-m)\cdot A^2(m)]\cdot 1_l\{0\}$$

For i=0,...,C_0−2, k=i+2,...,C_0: $P[(i,j),(k,l)] = 0$ (1f)

In equation (1), j and l vary from 0 to C_1 and $A^1(\cdot)$ is taken to be zero when its argument is negative. Likewise, the sums are zero when the summation is vacuous. Lastly, $1_z\{\cdot\}$ is the indicator function, which equals 1 if the argument equals z and otherwise equals 0.

The Markov chain is irreducible, aperiodic and positive recurrent. Denote its limiting distribution as: $\upsilon(i,j)$, i = 0,...,C_0 and j = 0,...,C_1. To obtain the steady state distribution, at an arbitrary time t, $p(k,l)$ = Prob($X^0(t) = k$, $X^1(t) = l$), condition on the age since the last token arrival and on the state of the Markov chain seen by that token arrival. This yields:

$$p(k,l) = r_1 \int_0^{1/r_1} \sum_{i=0}^{C_0}\sum_{j=0}^{C_1} P[(i,j),(k,l),x]\cdot\upsilon(i,j)\,dx \qquad (2)$$

Since the jobs are assumed to arrive as Poisson processes, and Poisson arrivals see time averages (PASTA), [4], then, $p(l,k)$ is also the distribution for number in system seen by job arrivals. A class 1 job is blocked iff banks 0 and 1 are empty and a class 2 job is blocked iff bank 0 is empty (as bank 2 is always empty). Denoting the probability a class i job is blocked as β_i and the throughput of class i jobs admitted by the throttle as θ_i, then

$$\beta_1 = p(0,0) \qquad \beta_2 = \sum_{l=0}^{C_1} p(0,l). \qquad (3a)$$

$$\theta_1 = \lambda_1(1-\beta_1) \qquad \theta_2 = \lambda_2(1-\beta_2) \qquad (3b)$$

3. GENERAL CASE: APPROXIMATE ANALYSIS

In this section, we return to the general case of Section 1 with N classes of jobs, each with a dedicated token bank and associated token arrival stream. We assume that all the token arrival streams are deterministic with constant spacing, but the rates may be different. We assume that the job arrival streams are N independent renewal processes partially characterized by the first two moments of the inter-renewal time or, equivalently, the rate (the reciprocal of the mean) and the squared coefficient of variation (SCV, variance divided by the square of the mean) of the inter-renewal time. We are primarily interested in the case of Poisson job arrival streams, but we include the more general renewal case to provide approximations for non-Poisson streams. Since there are multiple token arrival streams with possibly different rates, even with Poisson job arrival streams there are not convenient regeneration points; i.e., we cannot do the embedded Markov chain exact analysis that we did for two priority classes in Section 2.

As before, our goal is to determine the job blocking probabilities β_i and throughputs θ_i for each class. We develop approximations using the familiar parametric-decomposition approach [5]. In particular, we treat the dedicated banks and the overflow bank separately. We act as if all the token and job overflow streams coming into the overflow bank are mutually independent, and partially characterize each stream by one or more parameters. We then analyze the overflow bank given such inputs.

Even though the dedicated banks are assumed to be independent, the overflow independence is an approximation,

because the token and job overflow streams associated with any one class are clearly dependent. However, overflow independence is a natural approximation because one of the two overflow streams for each class will often dominate. Moreover, as the number of dedicated banks increases, the effect of the dependence obviously decreases.

We first analyze the class-i dedicated bank with token arrival rate r_i, job arrival rate λ_i, and job arrival SCV c_i^2. We calculate, either approximately or exactly, the job overflow rate λ_i' and the token overflow rate r_i', plus any additional parameters to characterize partially these overflow streams. (In this paper, we only consider SCVs of inter-overflow intervals in renewal process approximations, but other parameters could also be considered with this approach.)

We then analyze the overflow bank with mutually independent job and token arrival streams having rates λ_i' and r_i', $1 \le i \le N$ (possibly plus additional SCV parameters for each stream). We calculate an approximate class-i job overflow rate from the overflow bank λ_i'' for each class. The class-i job blocking probability is then $\beta_i = \lambda_i'' / \lambda_i$ and the class i throughput is $\theta_i = \lambda_i(1 - \beta_i)$.

We have developed several, different, separate approximations for the dedicated banks and the overflow bank, which can be put together in various ways to make composite approximations. We now describe three composite approximations.

3.1 A Deterministic Fluid Approximation

The first scheme is a simple deterministic fluid model: We act as if both jobs and tokens arrive at each bank continuously and deterministically like a fluid with the given rates, i.e., we let

$$\theta_i = \min\left(\lambda_i,\ r_i + \frac{(\lambda_i - r_i)^+}{\sum_{j=1}^{N} (\lambda_j - r_j)^+} \cdot \sum_{j=1}^{N} (r_j - \lambda_j)^+ \right). \qquad (4)$$

This is the most elementary approximation, because it does not represent any stochastic features.

3.2 Markov-Chain-Poisson Approximation

In our second scheme, we first fit the job arrival stream parameters at each dedicated bank to a specific renewal process and then approximate the stochastic process representing the number of tokens in each dedicated bank by the queue length process in a D/G/1/C model. We use this model to calculate the overflow rates from the dedicated banks. We then act as if the overflow streams are mutually independent Poisson processes and analyze the overflow bank by using an M/M/1/C model. In the case of Poisson job arrival streams, this Markov-Chain-Poisson approximation is a one-parameter method that attempts to capture the stochastic behavior.

When the job arrival stream is Poisson or batch Poisson, the D/G/1/C model for the dedicated bank is an exact analysis, but otherwise it is an approximation, because in the token bank the job arrival process keeps running when the token bank is empty, whereas in the D/G/1/C model service begins when there is an arrival to an empty system.

Since the D/G/1/C analysis is exact when the job arrival process is batch Poisson (BP), it is natural to use a BP renewal process, i.e., a BP process with a geometric batch-size distribution. This BP process is a two-parameter renewal process, characterized by the batch arrival rate λ_i^b and the mean batch size m_i^b or, equivalently, the geometric parameter q_i; i.e., the batch-size probability mass function (pmf) is $b_i(n)=(1-q_i)q_i^{n-1}$, $n \ge 1$, where $m_i^b=1/(1-q_i)$ and $q_i=(m_i^b-1)/m_i^b$. We relate these parameters to the specified mean λ_i^{-1} and SCV c_i^2 of an interarrival time by

$$\lambda_i = \lambda_i^b m_i^b \quad \text{and} \quad c_i^2 = 2m_i^b - 1. \qquad (5)$$

Given the D/BP/1/C model, we obtain the token overflow rate by solving for the equilibrium vector $\pi_i \equiv (\pi_i(0), \ldots, \pi_i(C_i))$ of the embedded discrete-time Markov chain (MC) describing the queue length process just prior to token arrivals, as in [6], from which we obtain the exact token blocking probability at the dedicated bank

$$r_i' / r_i = \pi_i(C_i). \qquad (6)$$

Since the rate of accepted tokens at the dedicated bank $r_i - r_i'$ must equal the rate of accepted jobs $\lambda_i - \lambda_i'$, the associated job blocking probability at the dedicated bank is

$$\lambda_i' / \lambda_i = 1 - (r_i - r_i') / \lambda_i. \qquad (7)$$

If $c_i^2 < 1$ or if another renewal process is deemed more realistic with $c_i^2 > 1$, then we can use a phase-type renewal process and calculate the overflow rates as in (6) and (7) after applying a MC analysis to a D/PH/1/C model [7]. (Now the phase is part of the state of the MC and $\pi_i(C_i)$ is the probability of having C_i tokens in the bank, obtained by summing the probabilities of all possible service-phase states.)

Given the job overflow rates λ_i' and token overflow rates r_i' from the dedicated banks, we analyze the overflow bank using an M/M/1/C model. We use the fact that the superposition of independent Poisson processes is again Poisson. Let $\Lambda' = \lambda_1' + \ldots + \lambda_N'$, $R' = r_1' + \ldots + r_N'$, $\rho' = R'/\Lambda'$, $\Lambda'' = \lambda_1'' + \ldots + \lambda_N''$ and $R'' = r_1'' + \ldots + r_N''$, where the double prime indicates overflows from the overflow bank. The exact blocking probabilities under the Poisson assumption are

$$\frac{\lambda_i''}{\lambda_i'} = \frac{\Lambda''}{\Lambda'} = \frac{1-\rho'}{1-(\rho')^{C_0+1}}, \quad \frac{r_i''}{r_i'} = \frac{R''}{R'} = \frac{1-(\rho')^{-1}}{1-(\rho')^{-(C_0+1)}} \qquad (8)$$

where as before C_0 is the capacity of the overflow bank. Combining (6)–(8) gives λ_i''.

3.3 A Full Two-Parameter Procedure

Our most elaborate approximation procedure determines SCVs partially characterizing the overflow streams and uses them to determine the blocking at the overflow bank. We

first use the D/BP/1/C or D/PH/1/C analysis in Section 3.2 to determine the overflow rates from the dedicated banks.

To determine an approximate SCV c_{Ti}^2 for the class-i token overflow stream from the dedicated bank, we use the D/BP/1/C model. Then the token overflow process is a renewal process and the inter-overflow time is distributed exactly as the first passage time from state C_i to state C_i in the ergodic discrete-time MC with states $\{0, 1,...,C_i\}$. The mean inter-overflow time is $1/r_i'$, which we can obtain from (6). From [8], we obtain

$$c_{Ti}^2 = [W_i]_{C_i \times C_i} \cdot (\pi_i(C_i))^2 - 1 , \tag{9}$$

where $[W_i]_{C_i \times C_i}$ is the $(C_i \times C_i)^{\text{th}}$ element of the matrix of second moments of the first passage times (measured in number of steps of the Markov chain) given by

$$W = M(2Z_{dg}\Delta - I) + 2(ZM - E(ZM)_{dg}) \tag{10}$$

with M the matrix of mean first passage times and Z the fundamental matrix, i.e.,

$$M = (I - Z + EZ_{dg})\Delta , \quad Z = (I - (P - A))^{-1} , \tag{11}$$

P the one-step transition matrix of the Markov chain, A the square matrix with each row being the equilibrium vector of P, Δ the diagonal matrix whose diagonal elements are the reciprocal of the equilibrium probabilities, I the identity matrix, E the square matrix with all entries equal to 1 and $(\cdot)_{dg}$ a diagonal matrix whose diagonal equals that of the argument matrix.

The job overflow process associated with a D/BP/1/C model is not a renewal process, so it is not easy to analyze or partially characterize. To obtain an approximating SCV c_{Ji}^2 for the class-i overflow stream from the dedicated bank, we use an M/M/1/C model for the dedicated bank; i.e., we approximate both the token and job arrival processes at the dedicated bank by Poisson processes. The job overflow process then becomes a renewal process and the SCV is

$$c_{Ji}^2 = \begin{cases} \dfrac{2C_i^2 + 4C_i + 3}{3C_i + 3} & \text{if } \rho_i = 1 \\ \dfrac{(1+\rho)(1-\rho^{2C_i+2}) - 4(C_i+1)(1-\rho_i)\rho_i^{C_i+1}}{(1-\rho)(1-\rho^{C_i+1})^2} & \text{if } \rho_i \neq 1 . \end{cases} \tag{12}$$

To derive (12) and higher moments, let J_i be the time between successive job overflows and let B_i be the length of a busy period for tokens (the interval beginning when a token arrives at an empty token bank until the bank is next empty again). Then

$$J_i \overset{d}{=} X_i + (1 - I_i)(B_i + J_i) , \tag{13}$$

where $\overset{d}{=}$ denotes equality in distribution, the four random variables on the right in (13) are independent, X_i is exponential with mean $1/(\lambda_i + r_i)$ and $P(I_i=1) = 1-P(I_i=0) = \lambda_i/(\lambda_i + r_i)$. We obtain (13) by considering what happens after a job overflow. The time until the next event is X_i. Then $I_i=1$ if the next event is a job arrival, in which case $J_i=X_i$. If $I=0$, then the next event is a token arrival. Then the remaining time until the next job overflow is the sum of a token busy period plus the time until the next job overflow after the system becomes empty (which is distributed the same as J_i given Poisson-job arrivals).

We are now ready to analyze the overflow bank. We first partially characterize the superposition token arrival process with rate $R' = r_1' + ... + r_N'$. A simple approximation for the associated SCV is the asymptotic method approximation from [9], i.e.,

$$c_{TA}^2 = \sum_{i=1}^{N} (r_i'/R')c_{Ti}^2 , \tag{14}$$

but to reflect the convergence to Poisson as N gets large, we suggest the refinement in (4.16) of [9], i.e.,

$$c_T^2 = 1 + (c_{TA}^2 - 1)\sum_{i=1}^{N} (r_i'/R')^2 . \tag{15}$$

We partially characterize the token superposition arrival process by its rate R' and its SCV in (15). Then we fit a renewal process to these parameters. In particular, we make the first two moments of the inter-renewal interval be $1/R'$ and $(c_T^2+1)/R'^2$. If $c_T^2 > 1$, then we use a mixture of two exponentials (hyperexponential, H_2) with balanced means as in (3.7) of [9]; if $1/2 < c_T^2 < 1$, then we use a convolution of two exponentials with different means as in (3.10) of [9]; if $0 < c_T^2 < 1/2$, then we use an Erlang (E_2) distribution with $c_T^2 = 1/2$ or we use a convolution of more exponential distributions to match c_T^2 exactly.

In order to be able to calculate easily the per-class blocking, we approximate the per-class job overflow processes by batch Poisson processes. (This approach was used for multi-server loss systems by van Doorn [10].) To obtain a two-parameter renewal process, we use geometric batch sizes, just as in Section 3.2.

However, prior to applying the analog of (5), we first deflate the SCVs to reflect the smoothing due to superposition. Moreover, the variability in streams with lower rates should have less impact. Hence, we replace c_{Ji}^2 by

$$\bar{c}_{Ji}^2 = 1 + \frac{\lambda_i'}{\Lambda'}(c_{Ji}^2 - 1) , \quad 1 \le i \le N . \tag{16}$$

We are now ready to calculate the approximate distribution of the number of tokens in the overflow bank at an arbitrary time. We use the fact that a superposition of independent BP processes is again a BP process. The superposition batch-size distribution is a mixture of the component batch-size distributions, where the weights are proportional to the rates, i.e., the final batch-size pmf at the overflow bank is

$$b'(n) = \sum_{i=1}^{N} (\lambda_i'/\Lambda')b_i'(n) , \quad n \ge 1 . \tag{17}$$

Assuming that tokens arrive to the overflow bank (bank 0) as a phase-type renewal process and that jobs arrive as a batch

Poisson process, the vector $\{X^0(t), J(t)\}$ is the state of a continuous-time Markov chain, where $X^0(t)$ is the number of tokens in bank 0 at time t, and $J(t)$ is the state at time t of the transient Markov chain associated with the phase-type distribution. The equilibrium vector can be solved for numerically, and then the steady-state number of tokens in bank 0, p_0, is obtained by summing over the states associated with the phases. The class-i overflow rate from bank 0, λ_i'', is the rate λ_i^b that class-i batches arrive at the overflow bank times the expected number of overflows from an arbitrary class-i batch. Let B_i be the number of overflows from an arbitrary class-i batch. The class-i batch-size distribution is $b_i'(n) = (1-q_i')(q_i')^{n-1}$, $n \geq 1$, where $q_i' = (m_i^b - 1)/m_i^b$. Hence

$$P(B_i = n) = \sum_{k=0}^{C_0} (1-q_i')(q_i')^{n+k-1} p_0(k), \quad n \geq 1, \tag{18}$$

$$E[B_i] = \sum_{n=1}^{\infty} nP(B_i = n) = (1-q_i')^{-1} \sum_{k=0}^{C_0} q_i'^k p_0(k) \tag{19}$$

and $\lambda_i'' = \lambda_i^b E[B_i]$.

4. ILLUSTRATIVE RESULTS

In this section, we that assume that the job arrival processes are Poisson. As a first comparison of the accuracy of the three approximations in Section 3.3, consider the special case of only one class of jobs and thus only one dedicated bank. In this case, the overflow bank is artificial and the epochs of blocking and admittance equal those from a throttle with a single bank of capacity $C_0 + C_1$. This case is potentially stressful for the approximations since they do not explicitly capture the negative correlation of token and job overflows. That is, when tokens are overflowing from a dedicated bank, jobs can not be, and vice versa. Also, because of the equivalence to a single-bank throttle, the true blocking is easily calculated, using equations (6) and (7).

Table 1 compares the blocking from the three approximations with the exact blocking for a range of normalized job arrival rates, λ_1/r_1. The three approximations of Section 3.3 are labeled respectively "fluid," "MC-Poisson" and "two-parameter." Note that for λ_1/r_1 outside of the interval (.9, 1.1), even the simple fluid approximation gives reasonably accurate blockings (in terms of the absolute, not relative, error). However, as the bank capacities are decreased towards 1, the accuracy of the fluid approximation declines. For $\lambda_1/r_1 \approx 1$, none of the models are exceptionally accurate. However, if an absolute error in blocking probability of 0.005 is tolerable for an initial design, then one could use the two-parameter approximation to explore the parameter space and then use simulation for fine tuning. As for the relative error, note that for light loads where the blocking is small, the relative error is orders of magnitude. Lastly, note that the stochastic MC-Poisson approximation is not much of an improvement over the deterministic fluid approximation. To obtain reasonable estimates for blocking when $\lambda_1/r_1 \approx 1$, one needs to use the two-parameter, or more accurate, approximations.

FRACTION BLOCKED, β_1
(One class of jobs, $C_0 = C_1 = 10$)

	APPROXIMATIONS			
λ_1/r_1	fluid	MC-Poisson	Two-parameter	EXACT
.50	.00000	.99 e-62	.21 e-8	.81 e-11
.75	.00000	.23 e-27	.31 e-4	.46 e-5
.90	.00000	.66 e-11	.002458	.001671
.95	.00000	.50 e-6	.009877	.007614
1.00	.00000	.00440	.02495	.02459
1.05	.04762	.04762	.05134	.05507
1.10	.09091	.09091	.09096	.09271
1.50	.33333	.33333	.33333	.33333
2.00	.50000	.50000	.50000	.50000

TABLE 1.

As a second comparison, consider the more interesting scenario of three classes of jobs where the decision maker wishes to guarantee each class a minimum admission rate but not restrict any class when the total arrival rate is below capacity. In particular, suppose the guaranteed admission rate is the same for all three classes; let $r_1 = r_2 = r_3 = 1$, and let all bank capacities be 20. Suppose class 1 is over its allocation, $\lambda_1 = 1.2$, and class 2 is under, $\lambda_2 = 0.5$, and let λ_3 vary.

PER-CLASS BLOCKINGS, β_i

Three classes of jobs: $\lambda_1 = 1.2$, $\lambda_2 = 0.5$;
all token arrival rates = 1.0, & all bank capacities = 20

		APPROXIMATIONS			
λ_3	Class	fluid	MC-Poisson	Two-parameter	SIMULATION
1.2	1	.0000	.00039	.0114	.0082
1.2	3	.0000	.00039	.0114	.0082
1.3	1	.0000	.00794	.0198	.0200
1.3	3	.0000	.0110	.0295	.0251
1.4	1	.0278	.0284	.0326	.0365
1.4	3	.0476	.0487	.0599	.0554
2.0	1	.0972	.0973	.0950	.103
2.0	3	.292	.292	.293	.288
3.0	1	.129	.129	.128	.134
3.0	3	.515	.515	.516	.513

TABLE 2.

The per-class blocking of jobs is given in Table 2, where the blocking for class 2 is omitted since it is less than 10^{-10}. The approximations are compared with a discrete event simulation written in FORTRAN. For each case, the

simulation is run for 5 million time units; the 95% confidence intervals are around ± 0.0006. In Table 2, significant digits from the simulation are displayed. (Note that for $\lambda_3 = 1.3$, the total arrival rate of jobs equals the total arrival rate of tokens.) Again, if initial or exploratory designs only require blockings to within 0.005, then the two-parameter model is adequate. Note that for $\lambda_3 = 2$ and 3, all three models predict similar, though biased, blockings. The overall blocking, however, from all three models agrees with that from the simulation to 3 significant digits: .189 and .362 respectively for λ_3 equal to 2 and 3.

For this scenario of three classes, and using the two-parameter approximation, the throughput for each class versus λ_3 is shown in Figure 1. These throughputs illustrate the general behavior of the throttle: as desired, the excess tokens from class 2 are shared by classes 1 and 3.

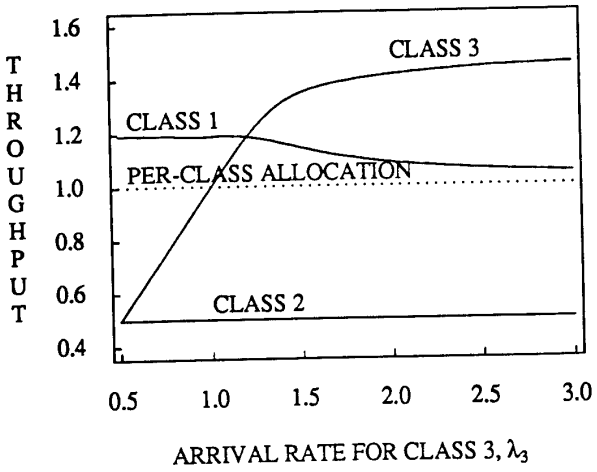

Fig. 1. Throughput versus arrival rate for class 3, given three classes where $\lambda_1 = 1.2$, $\lambda_2 = 0.5$, all token arrival rates are 1.0 and all bank capacities are 20.

2. *SUMMARY AND FUTURE WORK*

We introduced a multi-class input-regulation throttle and presented approximate analyses for the per-class blocking and throughput and an exact analysis for the special case of two priority classes of Poisson job arrivals. The simple fluid approximation does surprisingly well, except in the case of roughly equal token and job arrival rates and small bank capacities. Interestingly, the stochastic Markov-chain-Poisson approximation yields only modest improvement over the fluid approximation. Thus, approximations that reasonably capture at least the second moment of the overflow processes are needed for accurate estimates of blocking for the important case when the offered load is near the regulated limit. The two-parameter approximation of Section 3.3, plus simulation for fine tuning, seems adequate for most engineering purposes. Additional approximations are under study that use the interrupted Poisson Process.

The input-regulation throttle seems to be a promising candidate for meeting the goals stated in the introduction. Moreover, the design presented herein could be enhanced in obvious ways. If jobs of different classes use different amounts of system resources, then the tokens and jobs overflowing to bank 0 could respectively deposit and withdraw a class dependent number of credits. Also, the decision maker would have more control over the allocation of the excess capacity if there were more than one overflow bank and/or tokens could overflow to a dedicated bank of another class.

REFERENCES

[1] B. T. Doshi and H. Heffes, "Analysis of Overload Control Schemes for a Class of Distributed Switching Machines," *10th International Teletraffic Congress*, Montreal, Canada, 1983, Paper No. 5.2.2.

[2] A. E. Eckberg, D. T. Luan & D. M. Lucantoni "Bandwidth Management: A Congestion Contro Strategy for Broadband Packet Networks Characterizing the Throughput-Burstiness Filter,' *International Teletraffic Congress Specialist Seminar* Adelaide, Australia, September 1989, Paper No. 4.4.

[3] M. Sidi, W. Z. Liu, I. Cidon & I. Gopal. "Congestio Control Through Input Rate Regulation,' *GLOBECOM '89*, Dallas, Texas, November 1989 pp. 1764-1768.

[4] R. W. Wolff, "Poisson Arrivals See Time Averages,' *Operations Research*, vol. 30, 1982, pp. 223-231.

[5] W. Whitt, "The Queueing Network Analyzer," *Bel System Technical Journal*, vol. 10, 1983, pp. 2779 2815.

[6] A. W. Berger, "Overload Control Using a Rate Control Throttle: Selecting Token Bank Capacity foı Robustness to Arrival Rates," *Proceedings of the 28th IEEE Conference on Decision and Control*, 1989, pp. 2527-2529.

[7] A. W. Berger, "Performance Analysis of a Rate Control Throttle Where Tokens and Jobs Queue," *IEEE INFOCOM*, June, 1990, pp.30-38.

[8] J. G. Kemeny and J. L. Snell, *Finite Markov Chains*. Princeton: D. Van Nostrand.

[9] W. Whitt, "Approximating a Point Process by a Renewal Process, I: Two Basic Methods," *Operations Research*, vol. 30, 1982, pp. 125-147.

[10] E. A. Van Doorn, "A Note on Delbrouck's Approximate Solution to the Heterogeneous Blocking Problem," *IEEE Transactions on Communications*, vol. 32, 1984, pp. 1210-1211.