To appear in Computers and Operations Research

Scheduling Arrivals to Queues for Minimum Average Blocking: the S(n)/M/C/C system

Christopher Rose & Roy D. Yates

April 20, 1995

SCOPE & PURPOSE

It is usually assumed that we can only partially control the arrival process to stochastic service center. As such, we usually design service systems around the arrival characteristics of the customers. However, in many situations (i.e., telecommunications networks) there may be a large customer base with diverse stochastic characterizations. Furthermore, the characterizations may be unknown *a priori*. This further complicates the problem.

We therefore consider an inverted approach. Rather than design the service system around the unknown customer characteristics, we assume that the customers have complete control over how jobs are presented to the service system. Given such control, and a stochastic description of the service center, we then ask how customers should schedule the job arrivals times so as to maximize performance. This formulation opens a new suite of problems in service system design such as optimal scheduling to specific service facilities, competitive scheduling and social/greedy optima among many users. This paper helps lay the foundation for such work in the context of blocking systems where service requests may be denied on the basis of server (or waiting room) unavailability.

ABSTRACT

Given a finite time interval $[0, T]$, a fixed number of customers N, a stochastic service system with C servers, and an initial occupancy distribution $\mathbf{\Pi}(0)$, we show how to calculate a schedule for the customer arrival times to minimize average blocking. We have in mind a facility user who infrequently receives a burst of N items to submit where N is a random variable. If such a "bursty" user can tolerate the delay associated scheduling items over an interval of duration T, then substantial improvements in blocking performance can be had over random scheduling (N Poisson arrivals on $[0, T]$) and periodic scheduling (constant interarrivals).

The matrix exponential method used herein can be applied to any stochastic service system whose state transitions can be modeled by linear differential equations and for which an arrival changes the system state in a predictable way. However, we here concentrate on the tractable C-server exponential service system.

1 Introduction

1.1 Motivation

Service system design usually proceeds by approximating or characterizing traffic source behavior and then devising means to meet a desired performance criterion: typically specified in terms of delay or blocking. These criteria are determined directly by the service-provider and indirectly by the user via demand.

In the context of telecommunications systems, this approach met with great success for homogeneous traffic types with simple stochastic characterizations such as voice. Unfortunately, modern telecommunications network designers are faced with a burgeoning number of traffic types (fax, voice, packetized voice, video, computer communications etc.) and often conflicting delay, peak and average bitrate, and fidelity requirements. Even were it possible to completely and compactly characterize every possible source, the issue of efficient network design becomes very difficult.

One possible approach to accommodating diversity is to impose restrictive standards on source behavior [1], [2], [3]. In this way, the number of possible different traffic types is reduced and a more orderly system design process can be undertaken. A difficulty with this approach is getting a large number of service providers to agree upon standards in an increasingly deregulated environment. In addition, standardization does not allow for the introduction of new source types which might evolve after design and deployment are complete.

Consider then that the sources in a telecommunications context may be individual users of the facility or even network nodes. As such, the users may have some control over how jobs are scheduled to the facility. Therefore, consider an inversion of the general problem:

- Rather than design a service system around the source characteristics, let the sources design their job presentation strategy around the service center characteristics.

The idea of empowering users is not new and is one of the basic ideas of distributed systems. The novelty lies in the *laissez faire* notion of allowing each user to optimize personal performance criteria via traffic scheduling. A similar approach to flow control using game theory was taken in [4].

Before considering whether such an idea is effective in real network design, we must first develop a framework. To this end, this paper lays the foundation for one approach to the problem. We develop a statespace formulation which is generalizable to any Markovian stochastic service facility. We then consider scheduling to a simple C-server exponential facility.

1.2 Prior Work

Prior research in the general area of arrival scheduling dates back to the 1960's and papers by Mercer [5] and Jansson [6]. Jansson studied optimal fixed interarrival time schedules and Mercer considered queue length in a pseudo-scheduled arrival process. Hajek [7] has shown that constant interarrival times minimize all moments of the long-term average delay and other related quantities. This earlier work considered non-blocking service facilities rather than blocking systems and was also fundamentally steady-state in nature. This latter feature precludes application of the results to so-called "bursty" users which use the system only infrequently and unpredictably.

Recent work [8], [9], [10] is more relevant and concerns finding optimal schedules on a finite interval subject to delay or system-cost optimization criteria. However, no previous work has considered blocking systems and there is no straightforward way to adapt the prior results to the blocking case. Since blocking is often an important service system attribute, especially in a telecommunications system, we fill this gap by 1) formulating the scheduling problem, 2) solving analytically for a few special cases, and 3) providing numerical results for more general cases.

1.3 Problem Specification

User traffic scheduling introduces a new suite of problems in the design of telecommunications systems ranging from optimal scheduling for particular service facilities to game theoretic studies of greedy versus social behavior. However, before complicated scenarios are considered, it is useful to understand the simpler problem of how a single user should submit traffic to a random service facility. This constitutes an egalitarian best-case scenario of sorts since on average, competing users can do no better than to fully cooperate.

This established, we may ask how much performance feedback can be provided to the user. With a minimum blocking performance criterion and perfect, timely feedback, the user would simply submit traffic only when the server could accept it. In practice, however, feedback is some combination of ambiguous and delayed. Consider, for example, a high-speed communication network where detailed network state information aged by one link transmission delay is virtually useless in determining the current state of the network.

Here, we circumvent this issue by assuming that the user is given absolutely no timely feedback. The user must schedule arrivals based on *a priori* knowledge of the service facility, such as the number and types of servers and an initial state occupancy probability distribution at time t_0. This scheduling of arrivals without service facility feedback, or *blind scheduling*, constitutes a worst case scenario.

In this paper we will consider blind scheduling of N items to an exponential service system with C servers (the S(n)/M/C/C system).[1] In addition, we only consider scheduling on a fixed duration interval $[0, T]$. We have in mind the infrequent (bursty) user whose idle time greatly exceeds his/her active time. Thus, each scheduling interval might be viewed as independent of the last. We view T as the user's capacity to endure delay during the active phase. This reasonably tractable system serves as a starting point for investigations of user-controlled scheduling to a stochastic server.

Let P_i be the blocking probability of the i^{th} arrival to a stochastic service facility on an interval $[0, T]$. We define the average blocking probability of N arrivals on this interval as,

$$\bar{P}_b(N) = \frac{1}{N}\sum_{i=1}^{N} P_i \tag{1}$$

Average blocking probability is often used as a measure of quality and fidelity for such diverse telecommunications traffic types as packetized voice and packet data transfers.[2] In particular, the blocking probability may be related to information-carrying capacity under suitable assumptions (see [12] and the binary erasure channel). We developed a procedure to find schedules (arrival times on $[0, T]$) which minimize $\bar{P}_b(N)$ for a given value of N. For a bursty user, we assume that scheduling intervals are effectively independent of each other.

This independence assumption allows consideration of the long-term $\bar{P}_b$ experienced by such a user when N is a random variable with mean λT. We examine the special case where N is a Poisson random variable and compare the result to a "bursty Poisson" user who infrequently submits a Poisson number of items, N, according to a random (Poisson) schedule on $[0, T]$. We also assume that the source knows only certain general characteristics of the system to which it submits arrivals; i.e., queue capacity and service rate, the scheduling interval, and possibly the initial state probability distribution.

[1]We follow the nomenclature of [8] and [10] for scheduled queueing systems. The S(n)/M/1/C system is treated in [11].

[2]Another useful measure, the maximum dropping probability (i.e., $P_i \leq P_{thresh}$) $i = 1, 2...N$) is considered in [11].

1.4 Paper Organization

First, we provide the matrix differential equations for the undriven C-server queue. This formulation permits a compact representation for the average blocking probability. and may be applied to any stochastic service system for which a state transition matrix exists for the probability distribution. [3] The state-space approach allows us to numerically find optimal schedules and identify trends in the form of the schedules as functions of queue capacity, number of arrivals, initial probability distribution etcetera, when no closed form can be found.

We then compare optimal scheduling to both fixed-rate Poisson arrivals and strictly periodic arrivals on the interval and close with a discussion of our results.

2 Schedules and Exponential Servers

The S(n)/M/C/C queue is described by a system of linear constant-coefficient differential equations. Normally we are interested only in the steady-state behavior. Here, however, the transient solutions are needed to calculate the blocking probability as a function of time.

Since we schedule arrivals for particular points in time, the rate function $\lambda(t)$ is identically zero between arrivals. We can therefore view the system as a succession of intervals over which the system behaves as a queue with no stochastic arrival process. These intervals are tied together by matching boundary/initial conditions imposed by the scheduled arrivals.

2.1 Matrix Formulation

A matrix formulation allows us to easily derive an explicit expression for the average blocking probability and may be used with any system which may be described by a state transition matrix. The appropriate matrix differential equation is

$$\dot{\mathbf{\Pi}} = \mathbf{A}\mathbf{\Pi} \tag{2}$$

[3] Closed-form optimal schedules for the special cases S(n)/M/1/1 and S(2)/M/C/C are derived in Appendix A.

where for the S(n)/M/C/C case

$$\mathbf{A} = \begin{bmatrix} 0 & \mu & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & -\mu & 2\mu & 0 & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & C\mu \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 & -C\mu \end{bmatrix} \tag{3}$$

This state space formulation allows us to calculate the state at any time t given the state at any time t_0 as

$$\mathbf{\Pi}(t) = \mathbf{\Phi}(t - t_0)\mathbf{\Pi}(t_0) \tag{4}$$

where $\mathbf{\Phi}(\tau)$, the state transition matrix, is a square matrix of the same dimensions as $\mathbf{A}$ which satisfies Equation (2); i.e., $\mathbf{\Phi}(\tau) = e^{\mathbf{A}\tau}$ [13]. Equally direct probabilistic arguments can also be used to derive $\mathbf{\Phi}(\tau)$.

2.2 Blocking Derivation

Suppose we wish to submit arrivals to the queue after time $t = 0$ but are given no information as to whether our submissions are blocked or actually gain access to the queue. For an arrival scheduled at $t = t_1$, the number in the queue the instant after t_1 is

$$N(t_1^+) = \begin{cases} N(t_1) + 1 & N(t_1) < C \\ C & N(t_1) = c \end{cases} \tag{5}$$

Thus, the state probability distribution after a scheduled arrival is

$$\Pi_k(t_1^+) = \begin{cases} 0 & k = 0 \\ \Pi_{k-1}(t_1^-) & k = 1, \ldots, C-1 \\ \Pi_{C-1}(t_1^-) + \Pi_C(t_1^-) & k = C \end{cases} \tag{6}$$

The evolution of the state after t_1 can be obtained using these new initial conditions and our derived solutions with time now referenced to the arrival time t_1.

Let arrivals occur at times $\{t_i\}$. Then for $t_{k-1} < t \leq t_k$ we have

$$\mathbf{\Pi}(t) = \mathbf{\Phi}(t, t_{k-1})\mathbf{S\Pi}(t_{k-1}) \tag{7}$$

where $\mathbf{\Phi}(t, t_{k-1})$ is the state transition matrix $\mathbf{e}^{\mathbf{A}(t-t_{k-1})}$ and $\mathbf{S}$ is the matrix which accomplishes the translation of state probabilities after an arrival defined by (6). For a linear time-invariant system $\mathbf{\Phi}(t_2, t_1) = \mathbf{\Phi}(t_2 - t_1)$. If we let $\tau_k = t_k - t_{k-1}$ then

$$\mathbf{\Pi}(t_k) = \left[\mathbf{\Phi}(\tau_k)\prod_{i=1}^{k-1}\mathbf{S}\mathbf{\Phi}(\tau_i)\right]\mathbf{\Pi}(0) \tag{8}$$

If we define $\mathbf{\Gamma}^T = [0 \cdots 0\ 1]$, then the probability of blocking at time t_k is $\mathbf{\Gamma}^T\mathbf{\Pi}(t_k)$ and the average probability of blocking is then

$$\bar{P}_b = \frac{\mathbf{\Gamma}^T}{n}\sum_{k=1}^{n}\mathbf{\Pi}(t_k) = \frac{\mathbf{\Gamma}^T}{n}\left[\sum_{k=1}^{n}\mathbf{\Phi}(\tau_k)\prod_{i=1}^{k-1}\mathbf{S}\mathbf{\Phi}(\tau_i)\right]\mathbf{\Pi}(0) \tag{9}$$

3 Analytic Schedules for Special Cases

3.1 The Optimal Final Arrival Time is T

All optimal schedules on $[0, T]$ must have the last arrival occur at $t_n = T$. This can be seen by noting that the probability of the queue being full, $\Pi_C(t)$, decreases monotonically after the penultimate arrival at t_{n-1}. For a putative optimal schedule $(t_1, t_2, ...t_n)$ with $t_n < T$, we could always set $t_n = T$ thereby obtaining a schedule with smaller average blocking. This, of course, means that the putative optimal schedule with $t_n < T$ is not optimal. Therefore all optimal schedules must have $t_n = T$.

Unfortunately, this simple argument cannot be used to recursively find optimal schedules on successive subintervals of $[0, T]$ since the boundaries of such subintervals cannot be determined *a priori*. We therefore derive expressions for average blocking probabilities and find schedules which minimize them for a few simple cases in appendix A. Following is a listing of these results

3.2 Case I: S(n)/M/1/1

The optimal schedule for n arrivals to a queue with exactly one server and no waiting room is given by,

$$\tau_1 = \begin{cases} 0 & \Pi_1^0 \le e^{-\mu T/(N-1)} \\ \dfrac{T}{N} + \dfrac{N-1}{N\mu}\ln \Pi_1^0 & \text{otherwise} \end{cases} \tag{10}$$

and

$$\tau_k = \frac{T - \tau_1}{N-1} \qquad k = 2, \ldots, N \tag{11}$$

where the τ_i are the interarrival times starting from $t = 0$.

$\bar{P}_b$ is

$$\bar{P}_b = \begin{cases} (\Pi_1^0)^{\frac{1}{N}} \mathbf{e}^{-\mu T/N} & \Pi_1^0 \geq \mathbf{e}^{-T\mu/(N-1)} \\ \frac{1}{N}\Pi_1^0 + \frac{N-1}{N}\mathbf{e}^{-\mu T/(N-1)} & \text{otherwise} \end{cases} \tag{12}$$

3.3 Case II: S(2)/M/C/C

Since the second arrival must occur at the end of the interval, we only need solve for the first arrival time t_1. From appendix A. we have

$$t_1 = \left(\frac{C}{C+1}\right) T + \frac{1}{(C+1)\mu} \log\left(\frac{C\Pi_C^0}{\Pi_{C-1}^0 + C\Pi_C^0}\right) \tag{13}$$

Since t_1 must be greater than zero, if

$$\Pi_C^0 \leq \Pi_{C-1}^0 e^{-C\mu T}/C(1 - e^{C\mu T}) \tag{14}$$

then $t_1 = 0$. Notice that t_1 is strongly skewed toward the end of the interval as C increases. For C very large, $t_1 \approx T$ unless Π_C^0 is extremely small.

$\bar{P}_b$ is

$$\bar{P}_b = \frac{\Pi_C^0}{2}\left[(C+1)\left(\frac{C\Pi_C^0}{C\Pi_C^0 + \Pi_{C-1}^0}\right)^{\frac{-C}{C+1}} e^{-\mu T \frac{C^2}{C+1}} - (C-1)e^{-C\mu T}\right] \tag{15}$$

4 Numerical Methods for the N-Arrival Case

It is difficult to show that the optimal schedule for the N-arrival case is unique. In an attempt to show convexity over the space of valid schedules,[4] several different avenues of attack were tried without success. In previous work[8], a similar problem proving convexity was also encountered. Thus, optimal schedule uniqueness is an open problem.

We therefore used numerical techniques to find optimal schedules. By taking the gradient of $\bar{P}_b$, we can apply a number of efficient known procedures [14] to mount a search for the minimum. We performed the minimizations many times for each derived schedule with randomized starting points in an attempt to avoid local minima. We found that over a variety of queue sizes, initial conditions and interval durations, the optimization surface seems to contain only one global minimum over the region of valid schedules; we have never obtained more than a single solution to the same problem, as might be expected were there multiple minima.

[4]If we consider a schedule as simply an n-tuple, $\mathbf{t}$ over $\Re^n$ with no constraints to ensure a valid schedule, then $P_b(\mathbf{t})$ is NOT convex. This can be shown by counter-example.

5 Empirical Properties of Optimal Schedules

Owing to the form of the governing matrix differential equation we can assume the exponential service rate μ to be fixed at 1 sec^{-1} with no loss of generality.

5.1 Dependence on Initial Conditions

The user must make some *a priori* assumptions about the initial state of the service facility. Since only partial information may be available, it would be useful to know the sensitivity of the optimal schedules on this initial state.

First consider the case where exactly $C-M$ servers are known to be busy at $t=0$. The blocking probability of the first M arrivals is identically zero. Furthermore, since delaying these first M arrivals past $t=0$ can only adversely affect the blocking probability for the remaining $N-M$ arrivals, the optimal schedule places these M arrivals at $t=0$. This argument also applies to the case where the scheduler knows only that *at least* M servers are free at $t=0$. In both cases the scheduling problem then becomes one of finding an optimal schedule for $N-M$ arrivals subject to the initial conditions imposed by the first M arrivals at $t=0$. For exactly M servers free at $t=0$, the state at $t=0^+$ is $\Pi_C^{0+}=1$. For at least M servers free, $\mathbf{\Pi}(0+)=\mathbf{S}^M\mathbf{\Pi}(0)$.

We now seek the sensitivity of schedules to random initial conditions with $\Pi_C^0>0$. To this end we generated an occupancy distribution by choosing $C+1$ numbers uniformly on $[0,1]$ and then normalizing. In Figure 1 and Figure 2 we show schedules for $N=4,20$, $C=8$ for various T. Each plot in the montage consists of five schedules found for such randomly chosen initial conditions.

Empirically it seems that the influence of randomly chosen initial conditions on schedules is weak. Notice the relatively small variability of the schedules with the exception of scheduling intervals small with respect to the service time of 1 second. In particular notice the small variability in the later scheduled arrivals in all cases.

This appears to be a typical feature of optimal schedules. A possible explanation can be offered by analogy to the analytic schedules for the $C=1$ case and the $N=2$ cases which show a *log* dependence on the initial state at $t=0$. We might therefore expect small variations in the form of the schedule even with relatively large variations in the initial conditions.

5.2 Initial Delay and Quasi-Constant Interarrivals

Since the form of the optimal schedule does not seem to depend strongly on initial conditions (other than the known-free servers described previously), assume for now an initially full system at $t = 0$. This assumption simplifies our study, and in addition, offers a pessimistic bound on the expected performance; the blocking performance can be no worse than to start with a full system at $t = 0$.

Figure 3 shows optimal schedules for $C = 4$ and $T = 8$ for various N. Notice the increasing linearity of the schedules with increasing N; i.e., the schedule is almost periodic after an initial delay (first arrival time). Figure 4 is a plot of the first arrival time for $N = 8$ arrivals and an interval size $T = 4$ as a function of the number of servers, C.

A semi-quantitative explanation for this initial delay may be had by considering the dependence of $\bar{P}_b$ on T (Figure 5) for $C = 4$ and $N = 8$. The blocking probability of the first arrival decays as e^{-Ct_1}. The variation of e^{-Ct} is much stronger than the variation of $\bar{P}_b$ with T. The optimal schedule therefore balances the penalty associated with effectively shortening the scheduling interval against the benefit of decreasing blocking for the first arrival.

5.3 $\bar{P}_b^*$ vs. T for Fixed Arrival Rate

Figure 6 shows the dependence of $\bar{P}_b^*$ on T for fixed arrival rate $\lambda = N/T = 1$ and $C = 4$. Three cases are considered:

- System initially full
- System initially empty
- System initially in steady state

The figure shows that longer scheduling intervals do not necessarily result in better performance. For the system initially full, average blocking decreases with increased T as the overloaded system empties toward the offered load. For the system initially empty, average blocking increases with increased T.. For the system initially in steady state, blocking stays more or less the same.

However, as T tends to ∞ the optimal schedules for the full and empty systems asymptotically produce the same average blocking. If we define, $P_f^*(T, \lambda)$ as the minimum average blocking probability of $n = \lambda T$ arrivals on $[0, T]$ assuming an initially full system. and define $P_e^*(T, \lambda)$ similarly only assuming the system is initially empty we have,

$$P_e^*(T, \lambda) = \frac{n - C}{n} P_f^*(T, \frac{n - C}{n} \lambda) \tag{16}$$

which in the limit of large n (large T) gives

$$P_e^*(T^\infty, \lambda) = P_f^*(T^\infty, \lambda) \tag{17}$$

as suggested by Figure 6.

6 A Comparison to Poisson Arrivals and Uniform Schedules

When N is a random variable it is useful to define *average blocking probability* or $\bar{P}$. This is the probability that a given arrival will be blocked. For steady state systems, $\bar{P}$ is defined as,

$$\bar{P} = \lim_{X \to \infty} \frac{B(X)}{N(X)} \tag{18}$$

where $B(X)$ is the number of blocked arrivals on an interval of size X and $N(X)$ is the total number of arrivals on the same interval. However, when we repeatedly consider a set of arrivals on a *finite* interval, a modified definition is more appropriate. Let

$$\bar{P}(T) = \lim_{K \to \infty} \frac{\sum_{k=1}^{K} B_k(T)}{\sum_{k=1}^{K} N_k(T)} \tag{19}$$

where on the kth interval, $N_k(T)$ is the number of arrivals and $B_k(T)$ is the number of blocked arrivals. Note that $\bar{P}(T)$ is well defined when we assume a random arrival process on each interval of length T. Furthermore, $\bar{P}(T)$ is well defined when at the start of each interval we are given a random number N of arrivals to schedule. When the arrival process is Poisson and the initial distribution $\mathbf{\Pi}(0)$ is chosen as the $M/M/C/C$ stationary distribution $\mathbf{\Pi}^{\text{ss}}$, we denote $\bar{P}(T)$ by $\bar{P}_E(T)$.

It is not obvious how $\bar{P}_b(n)$, the average blocking of n arrivals on an interval of length T, relates to $\bar{P}$ or $\bar{P}(T)$, the per-arrival blocking probabilities. Appendix B explores this relationship in detail. A synopsis follows.

To establish the relationship of $\bar{P}_b(n)$ to $\bar{P}$, we first define the random variable $\mathcal{B}(N,T)$ as the sum of the blocking probabilities for N arrivals on an interval of duration T. From Equation (B.14) we have,

$$\bar{\mathcal{B}}(T) = E[\mathcal{B}(N,T)] = \sum_n p_N(n) n \bar{P}_b(n,T)$$

and we can then show that (Appendix B)

$$\bar{P}(T) = \frac{\bar{\mathcal{B}}(T)}{\bar{N}(T)} \tag{20}$$

where $\bar{N}(T)$ is the expected number of arrivals on the interval.

For steady-state Poisson arrivals of intensity λ and exponential service rate μ, the stationary distribution $\mathbf{\Pi}^{ss}$ is Erlang ([15]) where

$$\mathbf{\Pi}_k^{ss} = \frac{\rho^k/k!}{\sum_{j=0}^{C} \rho^j/j!} \qquad (0 \leq k \leq C)$$

and $\rho = \lambda/\mu$. In this case, $\bar{P}$, the probability that an arbitrary arrival is blocked, is denoted by $\bar{P}_E = \mathbf{\Pi}_C^{ss}$.

6.1 Fixed Number of Arrivals

Consider a Poisson arrival stream of rate $\lambda = \bar{N}/T$ on an interval $[0, T]$ where the system is initially full. Let N be the number of arrivals. Given $N = n$, the Poisson arrival stream is specified by n arrival times distributed uniformly over the interval $[0, T]$. We averaged 50,000 such random schedules for each value of n to generate $\bar{P}_b(n)$, the mean blocking probability of n Poisson arrivals on $[0, T]$. The results, compared to the mean blocking probability of n arrivals for optimal schedules, are shown in Figure 7 for a representative case ($C = 4$, $T = 16$). In addition, we provide $\bar{P}_b(n)$ for uniform schedules: arrivals spaced uniformly over the interval.

As expected, optimal scheduling provides far lower blocking than the average Poisson schedules for each value of n. The uniform schedule performs comparatively poorly as well, except at higher loads where it approaches the performance of the optimal. This observation is consonant with the form of the optimal schedules. In particular, as n increases, the optimal schedule becomes more uniform (see Figure 3) in that the effect of the initial delay imposed by the optimal schedule becomes less important.

6.2 $\bar{P}$ versus λ

Suppose a user is repeatedly given N arrivals to schedule on identical intervals of duration T where N is chosen i.i.d from a distribution $p_N(n)$. The average arrival rate is $\lambda = \bar{N}/T$. It is useful to know how the average blocking probability $\bar{P}(T)$ varies as a function of λ. Here we assume a Poisson distribution,

$$p_N(n) = (\lambda T)^n \frac{e^{-\lambda T}}{n!} \qquad (n \geq 0) \tag{21}$$

and an initial distribution $\mathbf{\Pi}(0)$.

Given initial conditions, we calculated optimal schedules and computed $\bar{P}_b^*(n)$ for $n = 1, 2...40$. We also computed the corresponding $\bar{P}_b(n)$ for Poisson schedules and uniform schedules using identical initial conditions (see section 6.1). Using Equation (20) we then computed $\bar{P}(T)$ as a function of arrival rate λ.[5] In addition we calculated the blocking probability for the appropriate uniform schedules.

A set of typical results is shown in Figure 8 with $C = 4$, $T = 16$ and $\mathbf{\Pi}(0) = \mathbf{\Pi}^{\mathrm{ss}}$. The optimal schedule provides much lower blocking probability than the Poisson random schedules and the uniform schedule performs almost as well as the optimal. The analytic steady state Poisson blocking probability $\bar{P_E}$ is also shown for comparison. This is as might be predicted by the steady-state work of Mercer [5], Jansson [6] and Hajek [7] by stretching their results for non-blocking systems.

However, consider next the results for an initially full system as shown in Figure 9. Figure 9 shows $\bar{P}(T)$ for optimal schedules, uniform schedules and Poisson arrivals, assuming an initially full system. In this case, the average per-arrival blocking probability for optimal schedules is far lower than that for both Poisson schedules and uniform schedules. Strictly uniform schedules are therefore strongly suboptimal, counter to a loose (and incorrect) analogy to the "Folk Theorem" [7].

7 Discussion & Conclusions

The benefits of scheduling *vis a vis* blocking are obvious when a user has N items to submit and exclusive use of the stochastic service facility on an interval of length T. However, when a user receives items burstily and relatively infrequently, as might be the case in practice (i.e.; a real-time computation, packets converging on an ATM network node), a scheduling scenario implies buffering of the stochastically arriving items and a concomitant delay (of maximum duration T) in submitting arrivals to the downstream server.

For example, flow control of ATM networks requires that network nodes and/or users throttle their submission rates, resulting in buffering and delay [3], [2], [1].[6] Therefore, since such buffering and delay occur naturally in a number of contexts, the addition of scheduling can only improve performance.

In high-fidelity, delay-insensitive applications such as large file transfers, the extra delay may not be problematic, especially since it comes with greatly diminished dropping probability. And even in applications where delay is important, scheduling provides a means of minimizing blocking subject to a clearly defined delay parame-

[5] Since n was limited, we constrained λ such that $\sum_{k=1}^{40} p_N(k) > 0.98$ so that the approximation of Equation (B.14) would be accurate.

[6] Leaky bucket flow control, where a backlog of packets are disbursed to the network at a fixed rate, though suboptimal, is often similar to optimal schedules for larger N.

ter, T, the scheduling interval duration. The performance index which quantifies this tradeoff between delay and blocking can always be specified by the user.

8 Future Work

The service system considered here was purposely simplistic. Scheduling to more realistic service center models should be explored. The formulation provided here allows the consideration of arbitrary service models which have a statespace description. In addition, the effect of imperfect/delayed feedback from the service facility must be considered. A number of tools can be brought to bear on this problem such as state observers, Kalman filtering and the like [13].

We can also begin to consider the question of competition/cooperation. i.e., we might consider the competitive advantage conferred to the scheduler when the other users are Poisson streams. Preliminary results show that, as might be expected, the scheduled user achieves lower blocking. However, more importantly, the competing Poisson user achieves only slightly poorer performance than if its competitor were another Poisson source rather than a scheduler. This suggests that scheduling is not too "socially unacceptable" in a shared resource setting. The general question of competition/cooperation between two schedulers may also be raised. Game theory [4], [13], [16] and evolutionary algorithms [17] suggest well-traveled analytic and heuristic approaches to this problem.

A Appendix: Special Cases

A.1 Case I: S(n)/M/1/1

Here we consider scheduling n arrivals to a queue with exactly one server and no waiting room. A little thought will reveal that the system is full ($\Pi_1(t_k) = 1$) just after each arrival whether the arrival is blocked or not. The transient solution is therefore particularly simple on the interval $[0, T]$.

$$\Pi_1(t) = \begin{cases} \Pi_1^0 \mathbf{e}^{-\mu t} & 0 < t \leq t_1 \\ \mathbf{e}^{-\mu(t-t_i)} & t_i < t \leq t_{i+1} \end{cases} \tag{A.1}$$

The blocking probability for the first arrival at t_1 is $\Pi_1^0 \mathbf{e}^{-\mu t_1}$ and the blocking probability of the ith arrival is $\mathbf{e}^{-\mu(t_i - t_{i-1})}$. Expressed in terms of the time differences $\tau_1, \ldots, \tau_n$, we have

$$\bar{P}_b = \frac{1}{n} \left[\Pi_1^0 \mathbf{e}^{-\mu\tau_1} + \sum_{k=2}^{N} \mathbf{e}^{-\mu\tau_k} \right] \tag{A.2}$$

We observe that $\bar{P}_b$ is a convex function of $\tau_1, \ldots, \tau_n$. Minimizing $\bar{P}_b$ subject to $\tau_i \geq 0$ for $i = 1, \ldots, N$ and $\tau_1 + \cdots + \tau_N = T$ is easily solved using Lagrange multipliers yielding

$$\tau_1 = \begin{cases} 0 & \Pi_1^0 \leq e^{-\mu T/(N-1)} \\ \dfrac{T}{N} + \dfrac{N-1}{N\mu} \ln \Pi_1^0 & \text{otherwise} \end{cases} \tag{A.3}$$

and

$$\tau_k = \frac{T - \tau_1}{N-1} \qquad k = 2, \ldots, N \tag{A.4}$$

From this, $\bar{P}_b$ is found as

$$\bar{P}_b = \begin{cases} (\Pi_1^0)^{\frac{1}{N}} \mathbf{e}^{-\mu T/N} & \Pi_1^0 \geq \mathbf{e}^{-T\mu/(N-1)} \\ \frac{1}{N}\Pi_1^0 + \frac{N-1}{N} \mathbf{e}^{-\mu T/(N-1)} & \text{otherwise} \end{cases} \tag{A.5}$$

A.2 Case II: S(2)/M/C/C

The solution for $\Pi_C(t)$ for the $N = 2$ case is,

$$\Pi_C(t) = \begin{cases} \Pi_C^0 \mathbf{e}^{-C\mu t} & 0 < t \leq t_1 \\ \mathbf{e}^{-C\mu t}[(\Pi_{C-1}^0 + C\Pi_C^0)\mathbf{e}^{\mu t_1} - (C-1)\Pi_C^0] & t_1 < t \leq T \end{cases} \tag{A.6}$$

Since $t_2 = T$ this result leads to

$$\bar{P}_b = \frac{1}{2} \left(\Pi_C^0 \mathbf{e}^{-C\mu t_1} + \mathbf{e}^{-C\mu T}[(\Pi_{C-1}^0 + C\Pi_C^0)\mathbf{e}^{\mu t_1} - (C-1)\Pi_C^0] \right) \tag{A.7}$$

It is readily shown that $\bar{P}_b$ is a convex function of t_1. Differentiating $\bar{P}_b$ with respect to t_1 and setting the result equal to zero yields,

$$t_1 = \left(\frac{C}{C+1}\right) T + \frac{1}{(C+1)\mu} \log\left(\frac{C\Pi_C^0}{\Pi_{C-1}^0 + C\Pi_C^0}\right) \tag{A.8}$$

Once again, since $t_1 \geq 0$, if

$$\Pi_C^0 \leq \Pi_{C-1}^0 e^{-C\mu T}/C(1-e^{C\mu T}) \tag{A.9}$$

then $t_1 = 0$. Notice that t_1 is strongly skewed toward the end of the interval as C increases. For C very large, $t_1 \approx T$ unless Π_C^0 is extremely small.

After substitution we find $\bar{P}_b$ to be

$$\bar{P}_b = \frac{\Pi_C^0}{2}\left[(C+1)\left(\frac{C\Pi_C^0}{C\Pi_C^0 + \Pi_{C-1}^0}\right)^{\frac{-C}{C+1}} e^{-\mu T \frac{C^2}{C+1}} - (C-1)e^{-C\mu T}\right] \tag{A.10}$$

It is useful to note that Equation (A.8)and (A.10) reduce to Equations (A.3) and (A.5) when $N = 2$ and $C = 1$.

B Appendix: $\bar{P}$ and $\bar{P}_b(n)$

Here we relate the average per-arrival blocking probabilities $\bar{P}$ and $\bar{P}(T)$ to $\bar{P}_b(n)$, the average blocking probability for n arrivals on an interval of length T. $\bar{P}$ and $\bar{P}(T)$ are defined in Equations (18) and (19) respectively.

We assume a set of K independent intervals of length T with identical initial conditions $\mathbf{\Pi}(0)$. Following Equation (19), define

$$\bar{P}(K,T) = \frac{\sum_{k=1}^{K} B_k(T)}{\sum_{k=1}^{K} N_k(T)} \tag{B.11}$$

and

$$\bar{P}(T) = \lim_{K\to\infty} \bar{P}(K,T) \tag{B.12}$$

To relate $\bar{P}(T)$ to $\bar{P}_b(n,T)$, we define the random variable, $\mathcal{B}(N,T)$, as the sum of the blocking probabilities for N arrivals scheduled on an interval of duration T and observe that

$$E[\mathcal{B}(N,T)|N=n] = n\bar{P}_b(n,T), \tag{B.13}$$

Given there were n arrivals in an interval, $\bar{P}_b(n,T)$ is the conditional average blocking probability. Note that $\bar{P}_b(n,T)$ depends on whether those n arrivals were scheduled or whether the n arrivals were the result of a random arrival process. However, $\bar{P}_b(n,T)$ is well defined in either case. The expected value of $\mathcal{B}(N,T)$ over N is

$$\bar{\mathcal{B}}(T) = \sum_n p_N(n) n \bar{P}_b(n,T) \tag{B.14}$$

In Equation (B.11), we can group the $B_k(T)$ in the numerator according to the number of arrivals on the interval. Specifically, define

$$S_n(K,T) = \sum_{k:N_k(T)=n} B_k(T)$$

as the sum of blocked arrivals over those intervals for which there were n arrivals and define $\alpha_n(K,T)$ as the number of intervals on which there were n arrivals. In this case,

$$\frac{1}{K}\sum_{k=1}^{K} B_k(T) = \sum_{n=0}^{\infty} \frac{\alpha_n(K,T)}{K}\frac{S_n(K,T)}{\alpha_n(K,T)} \tag{B.15}$$

Now, consider that the individual terms in the summation of Equation (B.15) satisfy

$$\lim_{K\to\infty} \frac{\alpha_n(K,T)}{K}\frac{S_n(K,T)}{\alpha_n(K,T)} = p_N(n) n \bar{P}_b(n,T) \tag{B.16}$$

which implies that

$$\lim_{K\to\infty} \frac{1}{K}\sum_{k=1}^{K} B_k(T) = \bar{\mathcal{B}}(T) \tag{B.17}$$

Since

$$\lim_{K\to\infty} \frac{1}{K} \sum_{k=1}^{K} N_k(T) = \bar{N}(T) \tag{B.18}$$

we have by Equation (B.11),

$$\bar{P}(T) = \lim_{K\to\infty} \bar{P}(K,T) = \frac{\bar{\mathcal{B}}(T)}{\bar{N}(T)} \tag{B.19}$$

C Figures

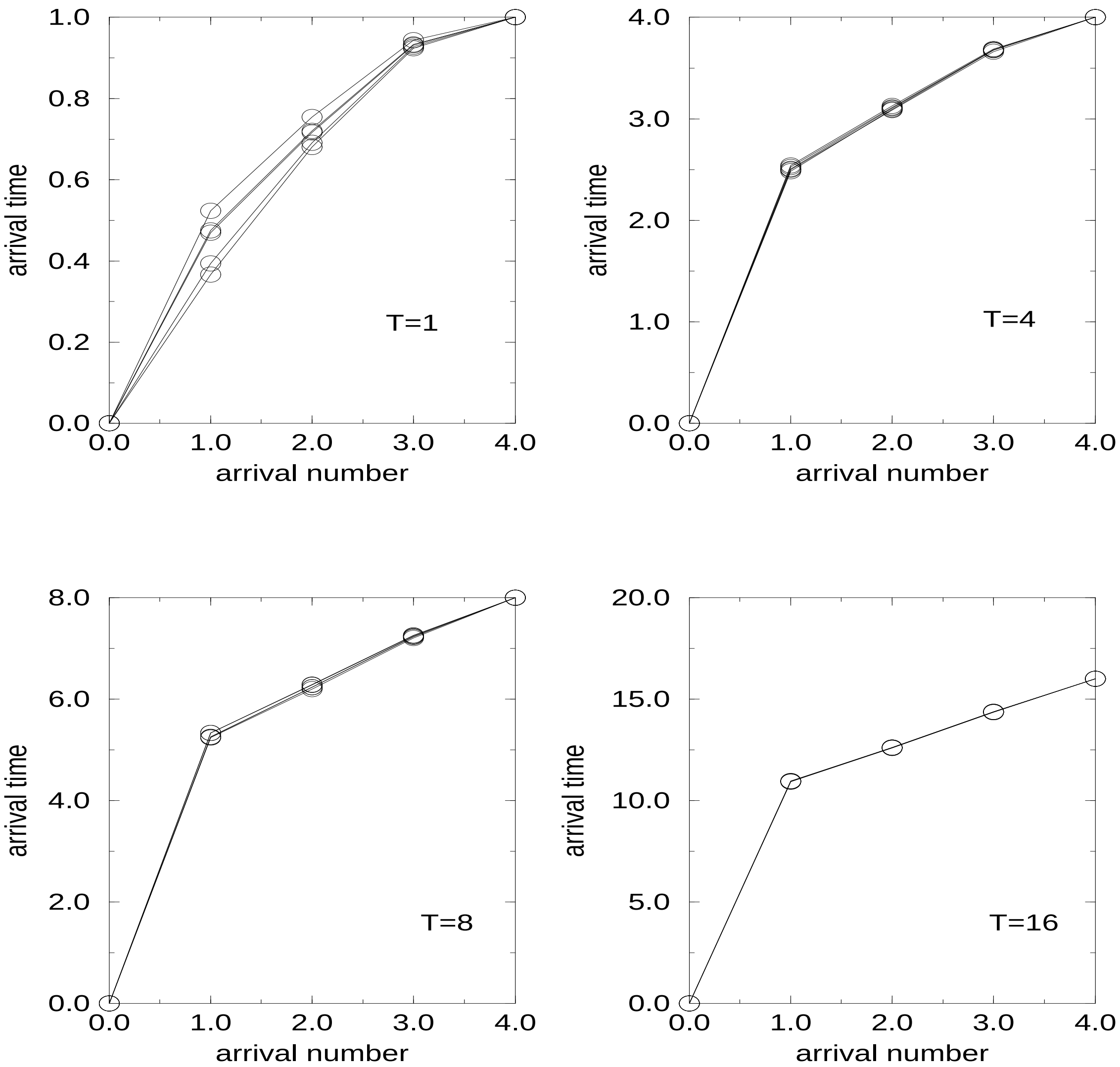

Figure 1: Schedules for random initial conditions with N=4, C=8.

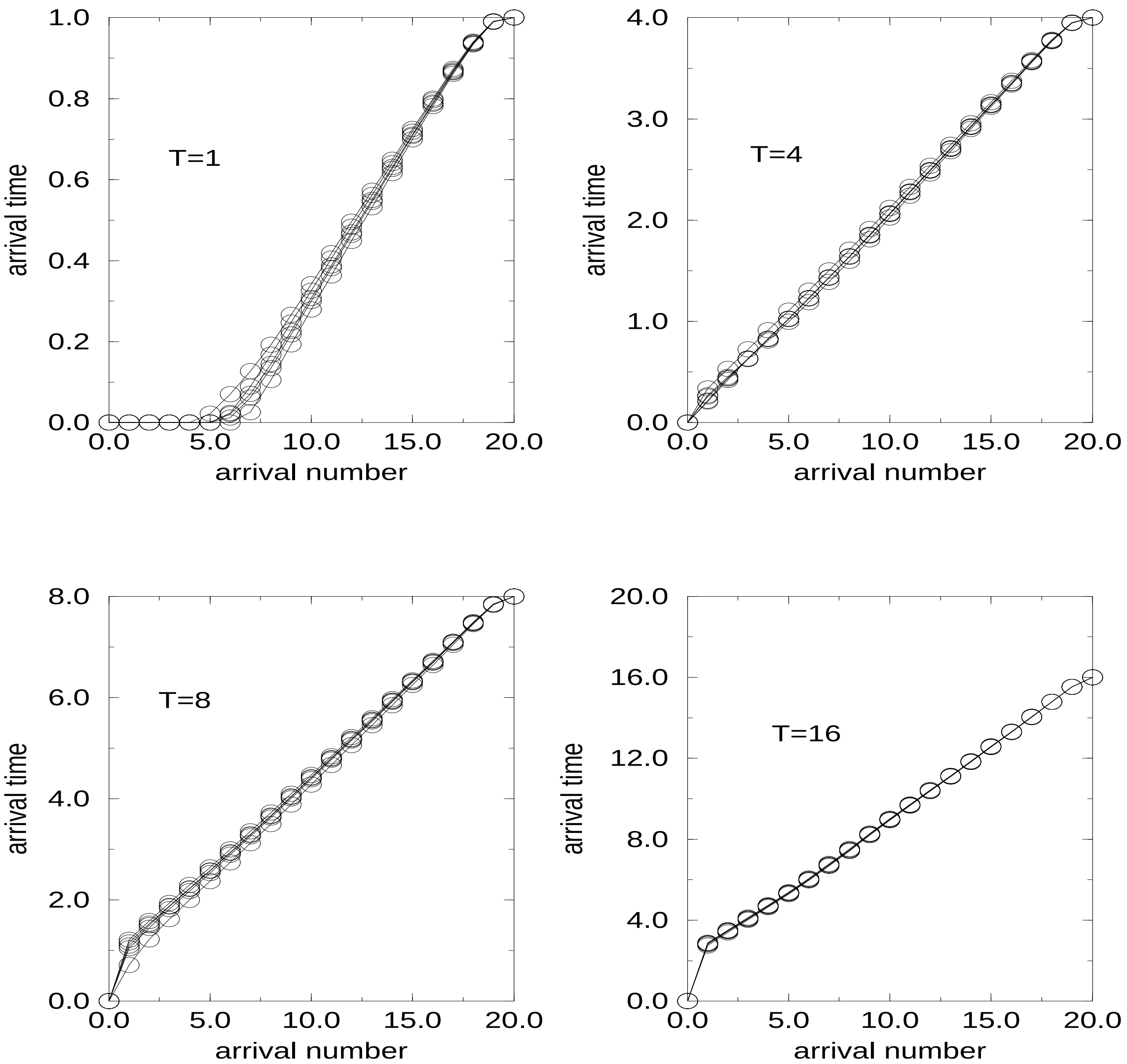

Figure 2: Schedules for random initial conditions with N=20, C=8.

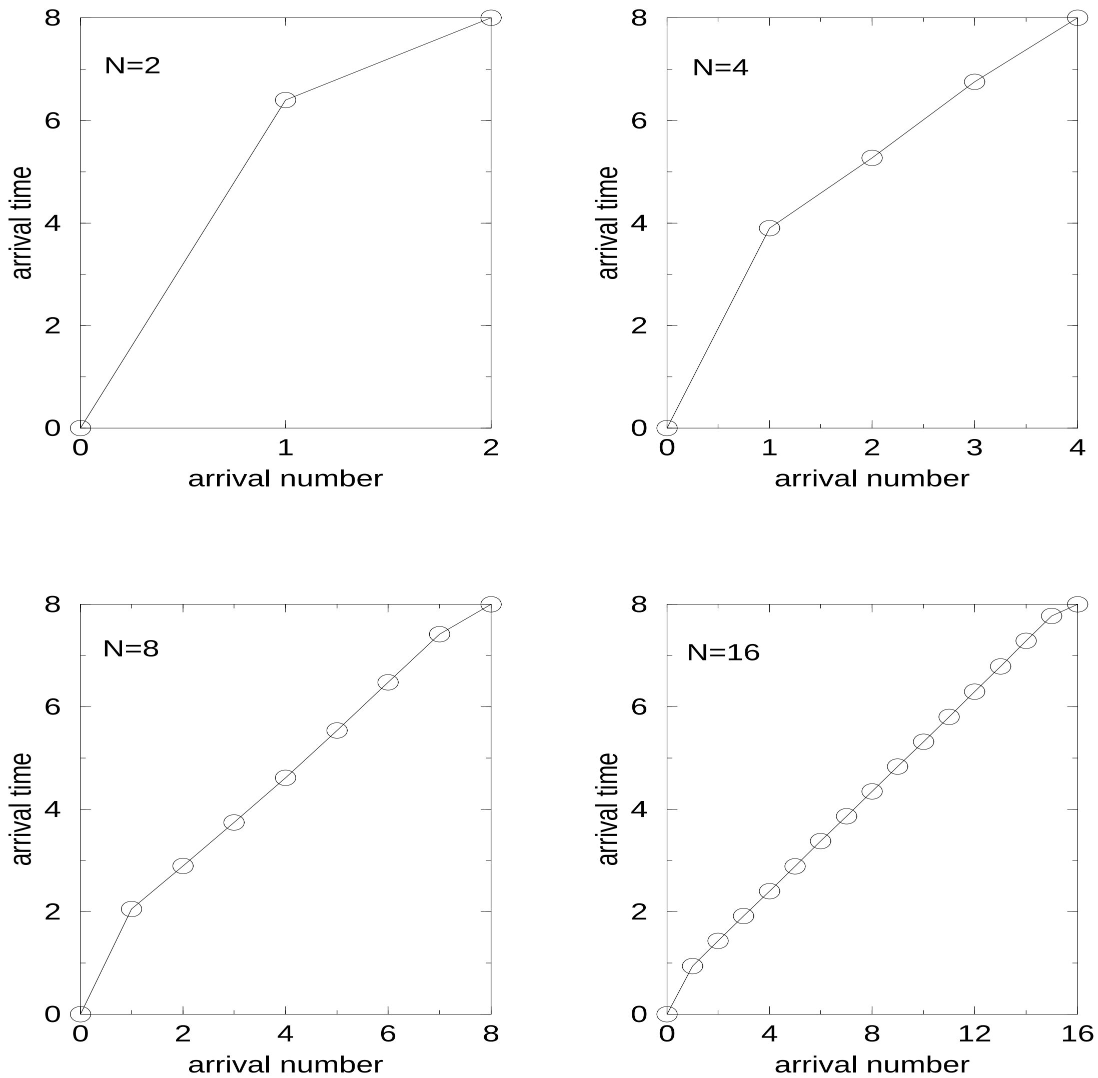

Figure 3: Montage of schedules for C=4, T=8 and various N.

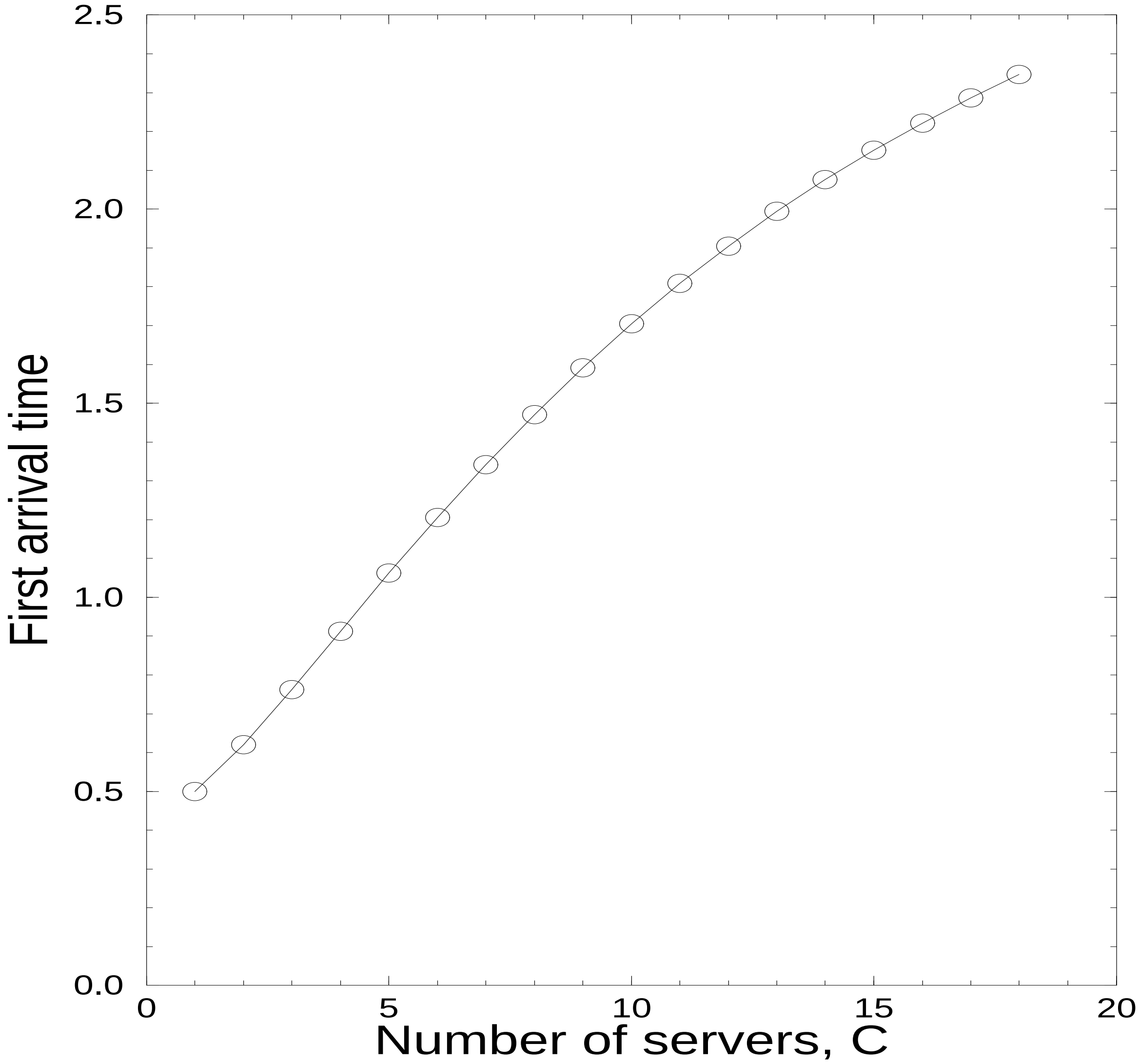

Figure 4: Dependence of first arrival time on C (N=8, T=4).

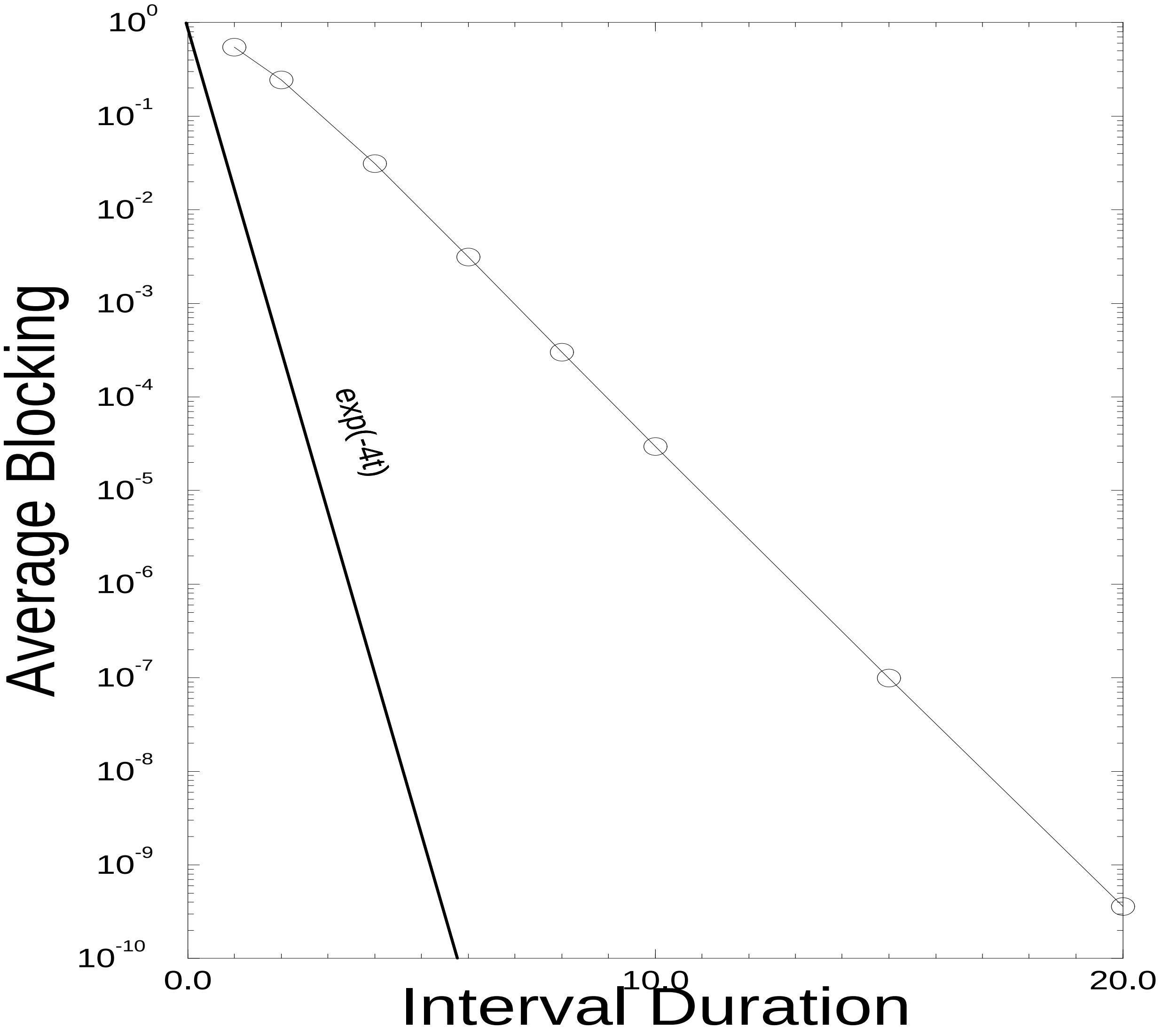

Figure 5: $\bar{P}_b$ as a function of T for fixed $N = 8$, $C = 4$. e^{-4t} also shown comparison.

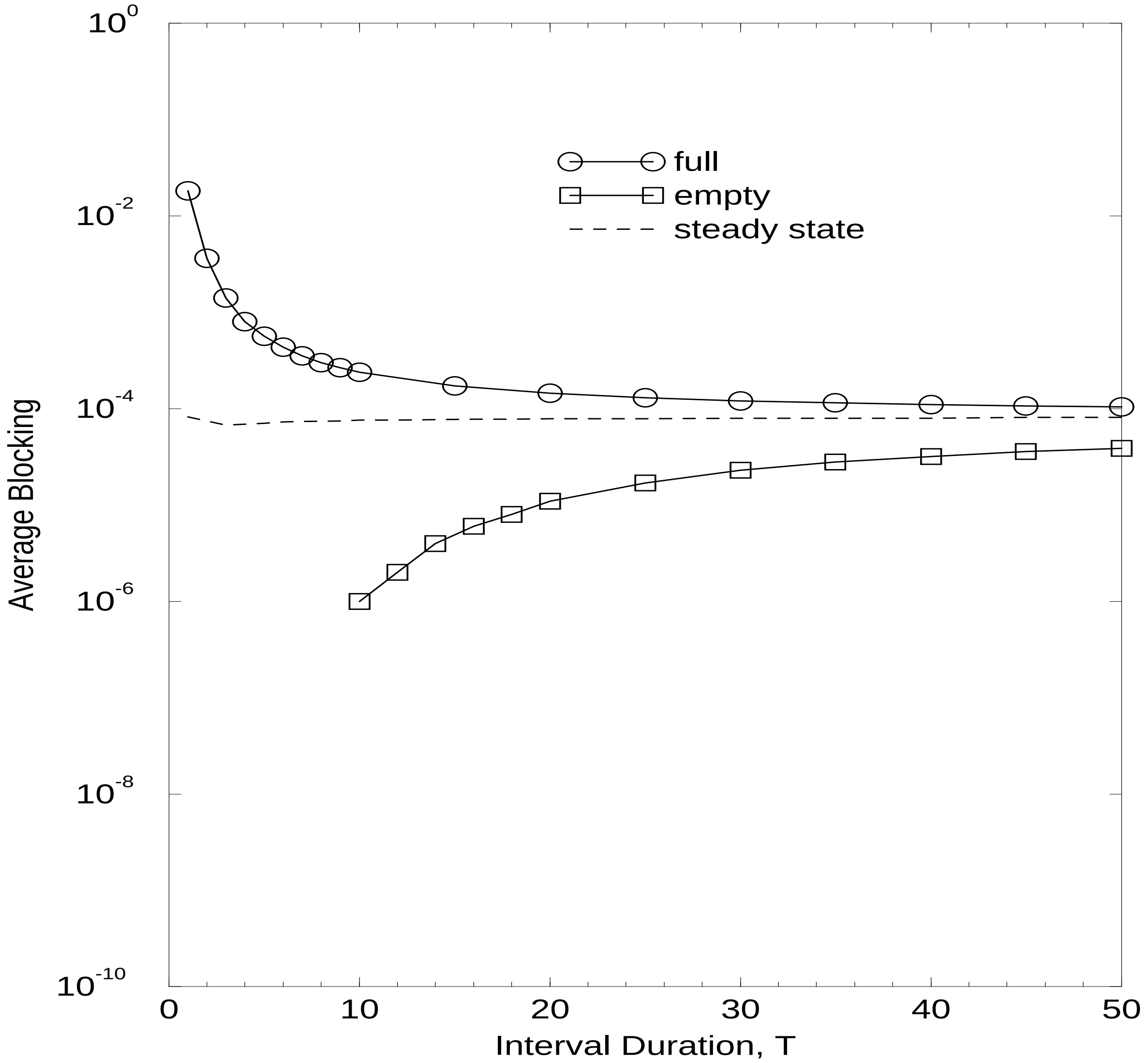

Figure 6: $\bar{P}_b$ as a function of T with fixed arrival rate of $\lambda = N/T = 1$ with $C = 4$. Curves shown for system initially full, initially empty and initially close to steady state.

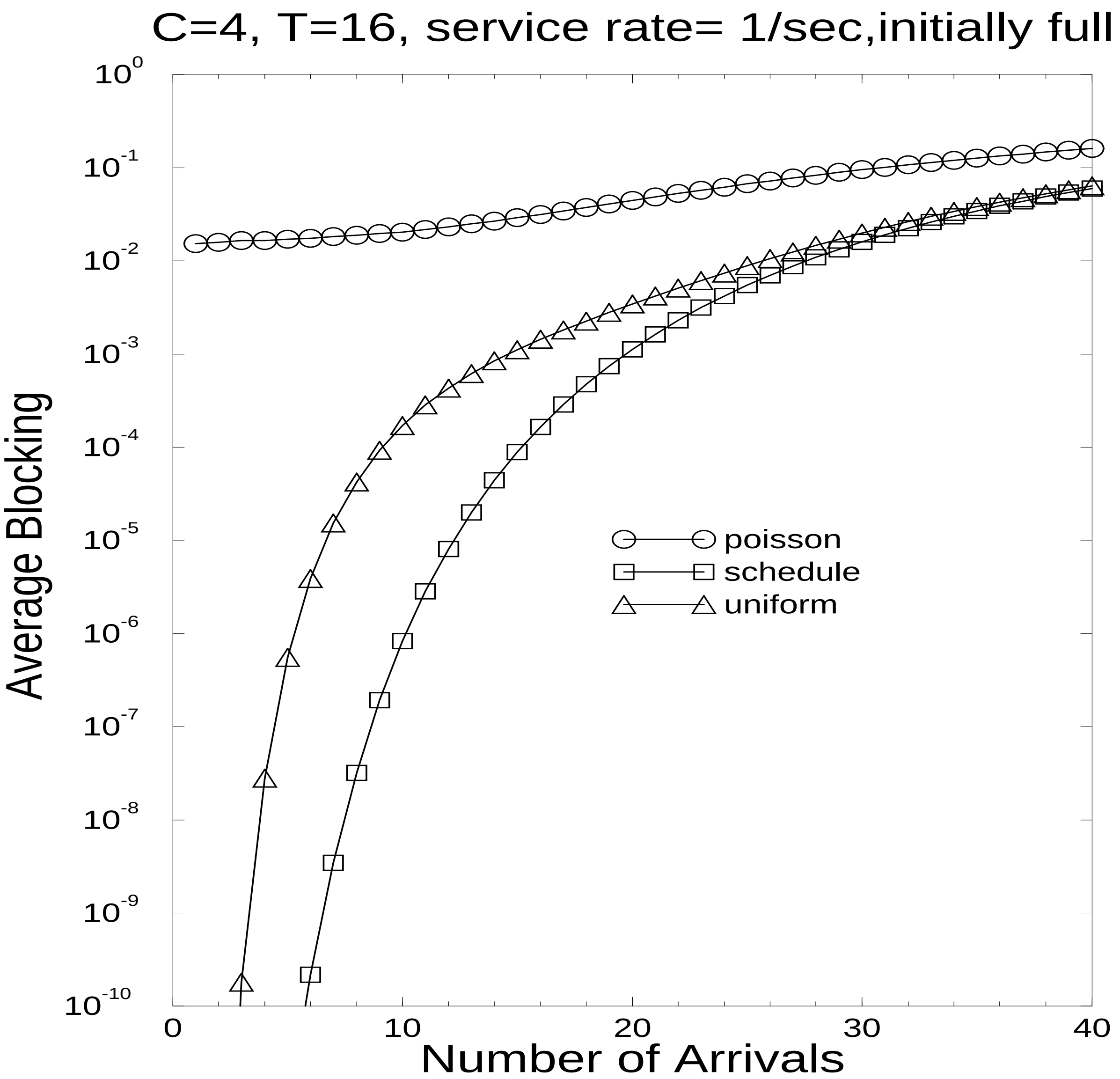

Figure 7: $\bar{P}_b$ as a function of number of arrivals for Poisson and optimally scheduled arrivals.

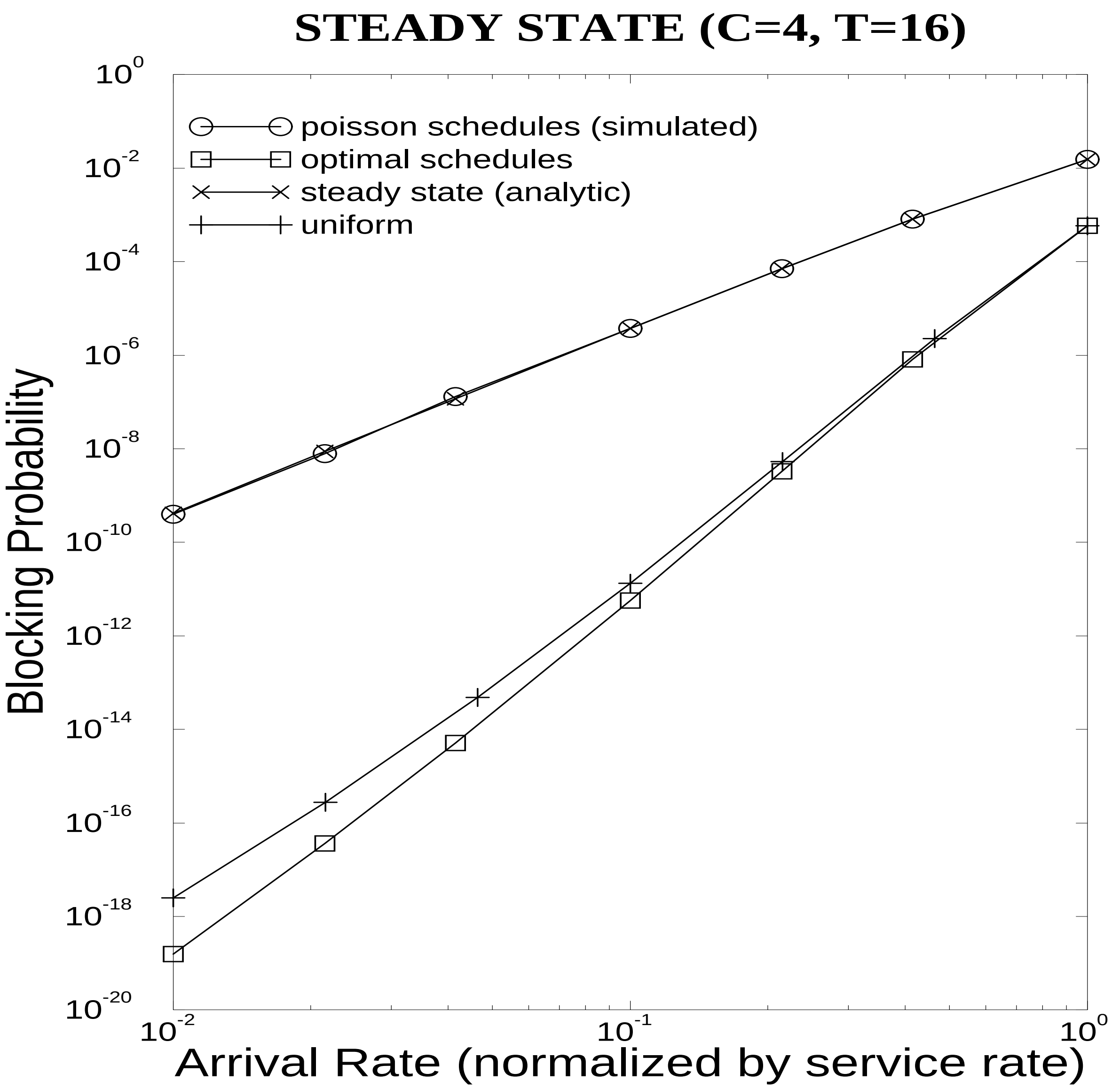

Figure 8: Comparison of per-arrival blocking probabilities for Poisson and scheduled arrivals (C=4, T=16). System assumed initially in steady state at specified rate. Steady state Poisson blocking probability as a function of rate also shown. Notice the agreement between the simulated and analytic Poisson results.

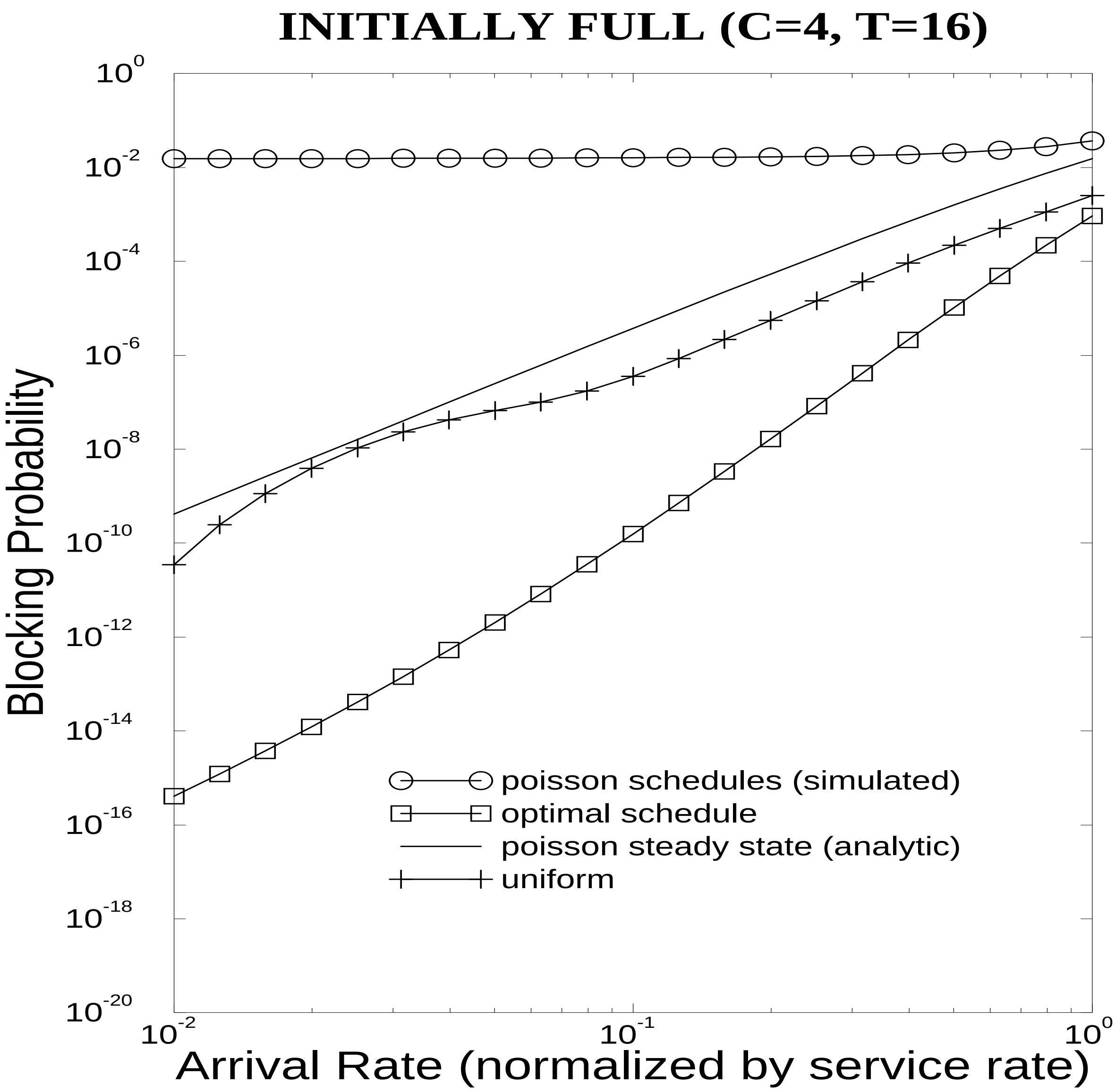

Figure 9: Comparison of average per-arrival blocking probabilities for Poisson and scheduled arrivals (C=4, T=16). System assumed initially full. Steady state Poisson blocking probability as a function of rate also shown.

References

[1] L. Dittman, B.J. Soren, and K. Moth. Flow Enforcement Algorithms for ATM Networks. *IEEE JSAC*, 9(3):343–350, April 1991.

[2] M. Butto, E. Cavallero, and A. Tonietti. Effectiveness of the "Leaky Bucket" Policing Mechanism in ATM Networks. *IEEE JSAC*, 9(3):335–342, April 1991.

[3] D.A. Hughes, G. Anido, and H.S. Bradlow. Performance of the Leaky Bucket Policing Mechanism for Small Bucket Depths. *Electronics Letters*, 26(16):1305–1307, 1990.

[4] C. Douligeris. *Optimal Flow control and Fairness in Communications Networks: a game theoretic perspective.* PhD thesis, Columbia University, Graduate School of Arts and Science, 1989.

[5] A. Mercer. Queues with Scheduled Arrivals: A Correction, Simplification, and Extension. *Journal of the Royal Statistical Society, Series B*, 35:104–116, 1973.

[6] B. Jansson. Choosing a Good Appointment System – A Study of Queues of the Type (D,M,1). *Operations Research*, pages 292–312, 1966.

[7] Bruce Hajec. The Proof of a Folk Theorem on Queueing Delay with Applications to Routing in Networks. *Journal of the Association for Computing Machinery*, 30(4):834–851, Oct. 1983.

[8] C.D. Pegden and M. Rosenshine. Scheduling Arrivals to Queues. *Computers and Operations Research*, 17(4):343–348, 1990.

[9] P.P. Wang. Static and Dynamic Scheduling of Customer Arrivals to a Single-Server System. *Naval Research Logistics*, 40:345–360, 1993.

[10] K.J. Healy. Scheduling Arrivals to a Stochastic Service Mechanism. *Queueing Systems*, 12:257–272, 1992.

[11] R.D. Yates and C. Rose. Scheduling Arrivals to Queues for Minmax Blocking: the S(n)/M/1/C system. *Princeton ICC*, March 1994. (also submitted to Queueing Systems).

[12] T. M. Cover and J. A. Thomas. *Elements of Information Theory.* Wiley-Interscience, 1991.

[13] A.E.Jr. Bryson and Yu-Chi Ho. *Applied Optimal Control; optimization, estimation and control.* Hemisphere Publishing Corp., 1975.

[14] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C.* Cambridge University Press, second edition, 1992.

[15] D. Gross and C.M. Harris. *Fundamentals of Queueing Theory.* Wiley, second edition, 1985.

[16] E.M. Valsbord and V.I. Zhukovski. *Introduction to Multi-Player Differential Games and Their Applications.* Gordon and Breach Science Publishers, 1988.

[17] D. E. Goldberg. *Genetic Algorithms: in search, optimization and machine learning.* Addison-Wesley, 1989.