

An Analytical Model for a Book-ahead Bandwidth Scheduler

Xiangfei Zhu, Mark E. McGinley, Tao Li, Malathi Veeraraghavan
151 Engineer's Way, PO 400740
University of Virginia
Charlottesville, VA 22904, USA
Email: {xz4p, mem5qf, taoli, mv5g}@virginia.edu

Abstract—Optical networks with book-ahead bandwidth schedulers are being deployed to meet the high-speed and predictable-service networking requirements of applications in the scientific research community. We present an analytical model for a single-link book-ahead bandwidth scheduler, which responds to advance reservation requests with the first-available time interval in which a channel is available. The link is assumed to be divided into m channels, and time is discretized into intervals. Our proposed model is a non-homogeneous continuous-time Markov chain, which has an embedded discrete-time Markov chain. We solve the model for call congestion, mean scheduling delay, and link utilization. This model can be used by network designers to select the size of the reservation window, K , in time-interval units, corresponding to a desired set of values for the output metrics, for a given value of m . For example, when m is 8, increasing the reservation window beyond 4 intervals does not affect the call congestion or system utilization, but causes the mean scheduling delay to increase. We show with comparative simulations that our analytical model can be used as a solution for an $M/D/m/p$ queueing system at moderate-to-high loads.

I. INTRODUCTION

There is an increasing interest in optical circuit-switched/virtual-circuit (VC) networks to support the high-speed and predictable-service requirements of applications in the scientific research community. The High Energy Physics (HEP) research project Atlas [1] is an example of a scientific effort that is characterized by both large datasets and large numbers of geographically distributed collaborators. Networking engineers involved in this project have expressed a need to reserve in advance high-bandwidth circuits/virtual circuits (VCs) dedicated for their exclusive use for data-transfer sessions.

Other scientific projects, such as remote visualization and remote instrument control, also benefit from advance reservations of dedicated high-bandwidth circuits/VCs. The extremely large datasets being visualized are often not collocated with the displays, e.g., [2], necessitating reserved high-bandwidth network connectivity for these sessions. Likewise, reserving network resources during remote control sessions of scientific instruments, such as the Advanced Photon Source (APS) at Argonne National Laboratory (ANL), enables scientists to have real-time closed-loop control of the equipment, and video monitoring of the experiment.

To meet the needs of these scientists for dedicated high-bandwidth circuits/VCs, several networking testbeds have been

deployed in many countries [3]–[6]. The key United States networks that support these scientific projects are the Department of Energy's ESnet4 project [5] and the new Internet2 Network. The new Internet2 Network [6] is comprised of a (circuit-switched) SONET network with dynamic circuit services, as well as an IP backbone, which continues and enhances the services provided by the Abilene backbone network. The ESnet4 network is being built on the new Internet2 optical infrastructure (currently under deployment). ESnet4 will consist of an IP core to handle general IP traffic, but also has the Science Data Network (SDN) core to provide dynamically provisioned, guaranteed-bandwidth high-speed circuits to accommodate eScience applications. Likewise, the Internet2 Network is designed to provide dynamically provisioned circuits in a hybrid environment, where IP and circuit-based capabilities are viewed as complementary.

Both ESnet and Internet2 recognize the need for supporting dynamic circuit services in addition to static dedicated circuit services, i.e., leased lines. This is because of the large numbers of sites requiring connectivity; the Atlas project alone involves hundreds of geographically distributed collaborators. To support dynamic circuit services, networking testbeds are developing advance reservation systems for network bandwidth, e.g., [7], [8], and [9]. We refer to advance reservation systems as **book-ahead (BA)** schedulers.

The goal of this work is to create a model for book-ahead schedulers for use in the design and operation of these networks. For example, with this model, a designer can select the size of the *reservation window*, which is the set of future time intervals for which the scheduler accepts reservations, and explore its impact on the system performance.

The contribution of this paper is a **Markov chain model** for a specific type of book-ahead scheduler. It is a single-link system, consisting of m channels, in which all advance reservation requests are for a single channel for a single time interval. The scheduler responds with an allocation in the earliest interval in which a channel is available. We call this the *BA-First* scheme. It is suitable for reservations for file transfers, where a computer application is the end-user of the network resource rather than a human user, as would be the case for remote-visualization type applications. In the latter case, other BA systems are more appropriate; for a listing of different types of BA schedulers, see [10].

The output metrics of our model are call congestion, mean scheduling delay, and link utilization. We show the effect of the reservation window size, K , on these metrics. We also show that our model can be used by network designers to select the design parameters corresponding to a desired set of values for the output metrics.

The remainder of this paper is structured as follows. Section II surveys related work on advance reservations. Section III describes the specific *BA-First* mechanism and our model. Section IV presents numerical results, showing the impact of various parameters, and comparing the model results with simulation results of an $M/D/m/p$ queueing system. Section V concludes the paper.

II. RELATED WORK

Perhaps most extensively considered is the subject of advance reservations co-existing with non-BA calls, also referred to as immediate (or instantaneous)-request (IR) calls. [11] [12] In BA calls, users specify their call-initiation times and call durations, while IR calls are open-ended. Both [11] and [12] consider techniques such as preemption of IR calls if required to honor a previous BA reservation. Given our eScience application motivation, we are interested in designing schedulers for networks in which *all* calls are of the BA type, simply because the dynamic circuit services being planned for ESnet4 and Internet2 will only accept BA calls, i.e., calls with prespecified call durations.

In other areas relating to optimization of advance reservation systems, [13] offers an algorithm to improve network utilization by delaying the admission decision in order to gather a number of requests and compute a better resource allocation. Their batch processing approach allows for better optimization, but suffers from a cost of speed and efficiency. Given the eScience community's interest in multi-domain circuit/VC scheduling, we design our system for simplicity, wherein arriving calls are either immediately assigned an interval or rejected.

A number of papers, such as [14] and [15], recognize the value of flexible bandwidth reservations for applications such as file transfers, where instead of requesting bandwidth for a particular duration, users specify the file size and a deadline. The scheduler can then maximize the number of admitted calls by breaking up the file into pieces and accommodating them wherever possible. While these problems are interesting, building practical schedulers able to achieve higher levels of utilization is secondary to a more near-term implementation of a simpler BA scheme, which is the focus of our paper.

In summary, most of the papers cited above seem to tackle more complex problems than ours. While these problem formulations are more interesting, their results have less practical value since often it is quite difficult to implement all the underlying details assumed in their formulations. We instead focus on a simpler problem, drawing upon the current direction of eScience targeted networks, in which all calls are of the BA-type, and understand the bandwidth-sharing aspect to a deeper level.

Papers that we found more directly relevant to our work are as follows. In [16], the impact of advance reservations on the computational complexity of path selection was examined. We are currently focused on understanding the bandwidth sharing aspect of book-ahead calls on just a single link, but this work will be important when we consider multi-link reservations. A model and a closed-form solution for a single-server book-ahead system with deterministic call holding times and specified call-initiation time are presented in [17]. It thus differs from our BA mechanism, in which users willingly accept any allocated interval. In other words, our work is similar to [17], but for a different BA system.

III. ANALYTICAL MODEL

We first describe the specific BA mechanism being modeled. We model the system with a non-homogeneous Continuous-Time Markov Chain (CTMC) in which we identify an embedded Discrete-Time Markov Chain (DTMC). We solve this model to obtain measures, such as call congestion, mean scheduling delay, and link utilization.

A. The *BA-First* mechanism

Our *BA-First* mechanism for sharing the capacity of a single link works as follows. The link capacity C is discretized and expressed as m channels where the bandwidth of each channel is C/m . Time is also discretized into equal-length intervals, of duration τ , as illustrated in Fig. 1. The reservation window, as defined earlier, consists of the set of future time intervals, for which the scheduler will accept reservation requests. The size of the reservation window is expressed as a number of time intervals, and denoted K . The scheduler maintains a record of the number of reserved channels for each time interval in the reservation window. At each time interval boundary, the reservation window slides to the right by one time interval.

Each call requests one channel for a duration of one time interval. Calls arriving in the current interval are assigned channels in the first-available reservation interval; in other words, even if the link is not fully occupied in the current interval, a newly arriving call will not be assigned a channel in that interval. Instead it will have to wait and be assigned to the earliest future interval (among the K intervals in the reservation window) in which a channel is available. A call will be blocked (rejected) if all channels have been reserved in every time interval of the entire K -interval reservation window.

B. *Non-homogeneous continuous-time Markov chain model*

We assume calls arrive according to a Poisson process with rate λ . The time for reservation processing by the scheduler is considered negligible in the model, which allows us to have the system accept or reject calls even if they arrive just before the start of a time-interval boundary.

To help us conceptualize the system, we use the word "bin" to represent an entity that holds reservations for each of the K reservation intervals as shown in Fig. 1. Each bin holds reservations for its corresponding reservation interval. A bin

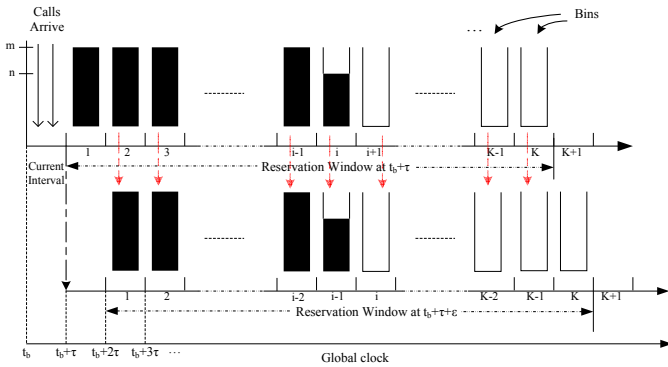


Fig. 1: An illustration of a *BA-First* scheduling system

is empty (i.e., the occupancy of a bin is 0) if no channel has as yet been reserved for its corresponding reservation interval. A bin is full (i.e., the bin occupancy is m) if all m channels are reserved for its corresponding reservation interval. Further, because a *BA-First* scheduler always tries to reserve the earliest available interval, if the i^{th} bin is not full, all bins after it must be empty. Therefore, the system state can be expressed as a 2-tuple (i, n) , where i is the index of the first bin that is not full, and n is the number of reserved channels in the i^{th} bin, as illustrated in Fig. 1. For all $1 \leq i < K$, the second element in the state vector, $n \leq (m - 1)$. For example, if $K = 4$ and $m = 2$, and the first two bins are full, the state of the system is $(3, 0)$ because bin 3 is the first “non-full” bin, rather than $(2, 2)$. But when $i = K$, we need an additional state (K, m) to denote that all bins are full. The state space S is then defined as

$$S \triangleq \{s = (K, m) | s = (i, n) : 1 \leq i \leq K \ \& \ 0 \leq n \leq m - 1\}.$$

The size of the state space is $N = mK + 1$.

The system meets the Markov property requirement that the future state depends only on the current state, not on the past history. The state of the system changes when one of two events occurs: (1) a call arrives, or (2) a time-interval boundary is encountered. Since a call can arrive at any instant, the Markov chain is a CTMC. As calls arrive within the current time interval, the scheduler immediately either accepts the call by allocating it a channel in the earliest available reservation interval, or rejects the call. This causes the system state (i, n) to change upon each call arrival. Fig. 2A shows the state transitions for call arrivals in an example system in which $K = 4$ and $m = 2$.

Fig. 2B shows the state transitions at time-interval boundaries for our example system in which $K = 4$ and $m = 2$. The system state will necessarily change at these time-interval boundaries because the reservation window slides one time interval to the right causing all calls with reservations for channels in bin 1 to instantly turn “live,” reservations from bin 2 to be moved to bin 1, reservations from bin 3 to be moved to bin 2, and so forth, up to and including the last bin K , as illustrated in Fig. 1. This explains why in Fig. 2B if the state system is $(1, 1)$ just before a time-interval boundary is

encountered, it changes to $(1, 0)$ at the time-interval boundary. The call with the single reservation in bin 1 goes live. Since bin 2 was necessarily empty (because bin 1 was not full, recall $m = 2$), in the new interval bin 1 remains the first non-full bin, and its occupancy is 0, i.e., the new state is $(1, 0)$. We arbitrarily pick one of two possible options to model when exactly the instantaneous change of state occurs at these time-interval boundaries. If the time-interval boundaries occur at $t_b, t_b + \tau, t_b + 2\tau, \dots$, as shown in Fig. 1, we select $(t_b + h\tau) + \epsilon$, where $h = 0, 1, 2, \dots$ and $\epsilon \rightarrow 0$, as the instant at which state changes occur. This convention statement is necessary for later formulations.

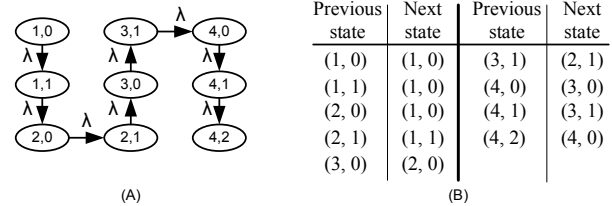


Fig. 2: Example transition diagram for $m = 2$ and $K = 4$.

This CTMC is non-homogeneous because the system behavior at the time-interval boundaries is different from its behavior at other times. However, if we only look at the system at the time-interval boundaries (i.e., at time instants $t_b, t_b + \tau, t_b + 2\tau, \dots$), there is an embedded time-homogeneous DTMC.

C. Embedded discrete-time Markov chain model

We derive the transition probability matrix for the embedded DTMC in this subsection. The discrete time instants are the time-interval boundaries, $t_b, t_b + \tau, t_b + 2\tau, \dots$. As stated in the previous section, we adopted the convention that state transitions occur at some small time, ϵ , past each time-interval boundary. State transitions for call arrivals in the just-past time interval, e.g., $(t_b + (h - 1)\tau, t_b + h\tau]$, where h is an integer, which would have occurred in the CTMC, are effectively aggregated into the one-step state transition in DTMC.

The transition probability in the DTMC from state (i, n) to state (j, q) , denoted by $p^{(i,n),(j,q)}$, is

$$p^{(i,n),(j,q)} = \begin{cases} 1 - F_A(m(K - 1) + m - 1) & \text{if } i = 1 \ \& \ (j, q) = (K, m), \\ G_A(m(j - i) + q) & \text{if } i = 1 \ \& \ (j, q) \neq (K, m), \\ 1 - F_A(mK + m - n - 1) & \text{if } i \neq 1 \ \& \ (j, q) = (K, m), \\ G_A(m(j - i + 1) + q - n) & \text{if } i \neq 1 \ \& \ (j, q) \neq (K, m), \end{cases} \quad (1)$$

where $F_A(a)$ is the Cumulative Distribution Function (CDF) of A , a Poisson random variable with parameter $\lambda\tau$, representing the number of call arrivals within a time interval, and $G_A(a)$ is defined as

$$G_A(a) = \begin{cases} P_A(a) & \text{if } a \geq 0, \\ 0 & \text{if } a < 0, \end{cases} \quad (2)$$

where $P_A(a)$ is the Probability Mass Function (PMF) of A .

Consider the first row of the Right-Hand Side (RHS) of (1). When $i = 1$, it means the first bin is not full. The number of reservations held in this bin, n , is irrelevant because when the next time instant in the DTMC (i.e., a time-interval boundary) arrives, these reservations turn into active calls. On the other hand, the number of calls that arrive in the just-past time interval is important. If this number is mK or higher, it means all the bins will become full and the system will transition to the (K, m) state. This is captured in the first row of (1), while the second row captures the case when the number of call arrivals is fewer than mK . As an example consider our $m = 2, K = 4$ system. If the source state is $(i = 1, n = 1)$ and the destination state of the transition is $(j = 4, q = 2)$, i.e., $(j = K, q = m)$, by applying the first row of the RHS of (1) we obtain the transition probability $p_{(1,1),(4,2)}$ as $(1 - F_A(7))$, which is the probability that more than 7 calls arrived during the current interval.

In the third and fourth rows of the RHS of (1), $i \neq 1$. Unlike when $i = 1$, n was irrelevant, here, when $i \neq 1$, n becomes important. For example, the first tuple in the new state j will equal $i - 1$, if the number of call arrivals in the just-past time interval is fewer than $m - n$. Again, based on whether the number of call arrivals is sufficient to completely reserve all the available channels across the entire reservation window or not, we get the third and fourth cases, respectively.

D. Performance metrics

To characterize the performance of the *BA-First* mechanism, we use the following three metrics: P_B , the call congestion, which is defined as the ratio of calls blocked to the total number of calls arrivals in a long observation interval [18], W , the mean scheduling delay for admitted calls, and U , the long-run utilization of the system. To calculate these three metrics, we first calculate the steady-state probabilities, denoted by vector π , of this discrete-time Markov chain using well-known techniques [18].

1) *Call congestion*: In any time interval, $(t_b + (h - 1)\tau, t_b + h\tau]$, where h is an integer, the average number of call arrivals is $\lambda\tau$. The number of calls blocked in this interval depends upon the state of the system at time $(t_b + (h - 1)\tau)$. We therefore use a vector $(\mathbf{b}_{t_b} \triangleq b_{(t_b, i, n)}, (i, n) \in S)$, where each element $b_{(t_b, i, n)}$ denotes the blocked-call ratio in this interval if the system is in state (i, n) at the start of the interval. We define the accepted-call ratio as $Q_{(t_b, i, n)}$. Since the DTMC is time-homogeneous, for simplicity, we drop the t_b parameter, and compute $b_{(i, n)}$ as follows:

$$\begin{aligned} b_{(i, n)} &= 1 - Q_{(i, n)} \\ &= 1 - \frac{\sum_{j=0}^{d_{(i, n)}} j P_A(j) + d_{(i, n)}(1 - F_A(d_{(i, n)}))}{\lambda}, \end{aligned} \quad (3)$$

where $P_A(i)$ and $F_A(i)$ are the PMF and CDF of the previously defined random variable A , respectively, and $d_{(i, n)}$ is the total number of channels that are available for reservation in that interval, which can be calculated as

$$d_{(i, n)} = \begin{cases} mK & \text{if } i = 1, \\ m(K - i + 1) + m - n & \text{if } i > 1. \end{cases} \quad (4)$$

The long-run call congestion, P_B , can then be computed as

$$P_B = \pi \mathbf{b}^T, \quad (5)$$

where the components of the vector \mathbf{b} are the blocked-call ratios computed in (3).

2) *Scheduling delay*: The scheduling delay of a call consists of a fractional part, which is the delay within the arrival interval, and an integral part, which is the number of intervals before the scheduled service interval in the reservation window. Consider an arbitrary interval $(t_b + (h - 1)\tau, t_b + h\tau]$, where h is an integer. Let $w_{(i, n)}$ be the average scheduling delay conditioned on the embedded DTMC being in state (i, n) at time $t_b + (h - 1)\tau$. The average is calculated over the calls admitted during the interval of interest. Denote $r_{(i, n)}$ and $z_{(i, n)}$ as averages of the fractional parts and integral parts of scheduling delays, respectively. We have $w_{(i, n)} = r_{(i, n)} + z_{(i, n)}$. Before computing $w_{(i, n)}$, we derive $a_{(i, n)}$, the conditional expected number of calls admitted in the interval, which can be computed as

$$a_{(i, n)} = \sum_{j=1}^{d_{(i, n)}-1} j P_A(j) + d_{(i, n)}(1 - F_A(d_{(i, n)} - 1)), \quad (6)$$

where $d_{(i, n)}$ is given in (4).

Let us focus on $r_{(i, n)}$ first. Recall that the arrival process is Poisson. Assume there are exactly j arrivals in the interval under consideration. It is a well-known result that, given that the number of total arrivals is j in the interval, the expected time between the arrival instant of the l^{th} arrival and the end of the interval, i.e., $(t_b + h\tau)$, is $\tau(1 - \frac{l}{j+1})$ [19]. In our problem, the number of calls admitted within the interval is at most $d_{(i, n)}$. If $j \leq d_{(i, n)}$, the average fractional part of scheduling delays for j calls is $\tau(1 - (\sum_{i=1}^j \frac{i}{j+1})/j) = \frac{\tau}{2}$. If $j > d_{(i, n)}$, the average is computed over just the first $d_{(i, n)}$ calls, which results in $\tau(1 - \frac{d_{(i, n)}+1}{2(j+1)})$. Therefore, $r_{(i, n)}$ can be computed as

$$\begin{aligned} r_{(i, n)} &= \sum_{j=1}^{d_{(i, n)}} \frac{\tau}{2} P_A(j) + \sum_{j=d_{(i, n)}+1}^{\infty} P_A(j) \tau(1 - \frac{d_{(i, n)}+1}{2(j+1)}) \\ &= \tau - \frac{\tau}{2} F_A(d_{(i, n)}) - \frac{\tau(d_{(i, n)}+1)}{2\lambda} (1 - F_A(d_{(i, n)}+1)). \end{aligned} \quad (7)$$

Denote $v(i, n, j)$ as the sum of the integral parts of scheduling delays conditioned on the DTMC being in state (i, n) at time $t_b + (h - 1)\tau$ for j calls admitted in the interval. Then, $z_{(i, n)}$ can be computed as

$$\begin{aligned} z_{(i, n)} &= \frac{1}{a_{(i, n)}} \left(\sum_{j=1}^{d_{(i, n)}-1} P_A(j) v(i, n, j) \right. \\ &\quad \left. + (1 - F_A(d_{(i, n)} - 1)) v(i, n, d_{(i, n)}) \right), \end{aligned} \quad (8)$$

where $a_{(i, n)}$ is obtained in (6), and $v(i, n, j)$ can be computed

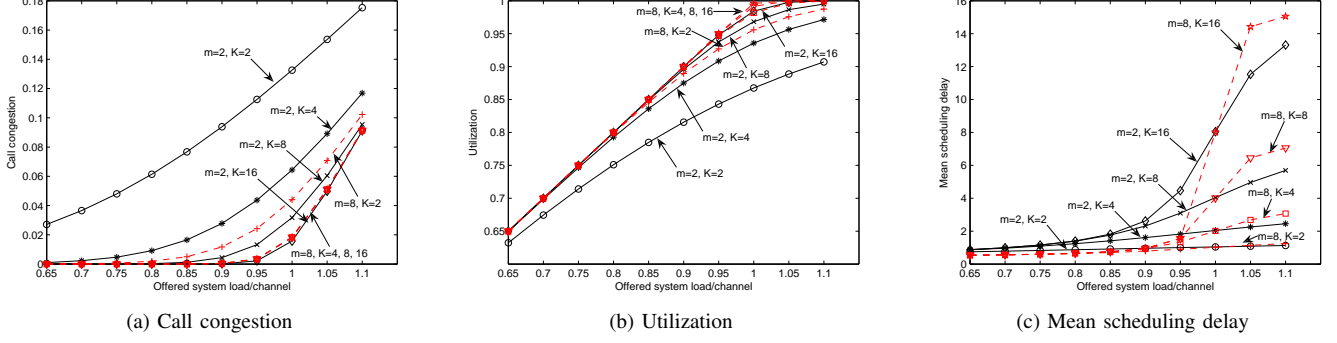


Fig. 3: The effect of reservation window size on the system performance.

by

$$v(i, n, j) = \begin{cases} \tau \sum_{l=1}^j \lfloor \frac{l-1}{m} \rfloor & \text{if } i = 1, \\ \tau \sum_{l=1}^j ((i-2) + \lfloor \frac{l+n-1}{m} \rfloor) & \text{if } i > 1. \end{cases} \quad (9)$$

The long-run mean scheduling delay W is

$$W = \frac{\sum_{(i,n) \in S} \pi(i,n) a(i,n) w(i,n)}{\sum_{(i,n) \in S} \pi(i,n) a(i,n)}. \quad (10)$$

3) *Utilization*: Link utilization in a time interval $(t_b + (h-1)\tau, t_b + h\tau]$, where h is an integer, depends upon the state of the system at time $t_b + (h-1)\tau$, i.e., just before the state transition. It solely depends on the occupancy of the first bin in state (i, n) because the occupancy of the first bin at time instant $t_b + (h-1)\tau$ decides the link occupancy during the time interval $(t_b + (h-1)\tau, t_b + h\tau]$. Therefore, if $i = 1$, which means the number of occupied channels in the first bin is less than m , the utilization of the system in this interval is $\frac{n}{m}$. Otherwise, the first bin is full, and the system utilization is 1. Again, we define the vector $(\mathbf{u} \triangleq u_{(i,n)}, (i,n) \in S)$, where each element $u_{(i,n)}$ is the utilization if the system is in state (i, n) in any interval. Therefore $u_{(i,n)}$ can be calculated as

$$u_{(i,n)} = \begin{cases} \frac{n}{m} & \text{if } i = 1, \\ 1 & \text{if } i > 1. \end{cases} \quad (11)$$

The long-run link utilization U can then be calculated by

$$U = \boldsymbol{\pi} \mathbf{u}^T. \quad (12)$$

IV. NUMERICAL RESULTS

We show numerical results obtained from the analytical model of the *BA-First* system, and compare to simulation results of an $M/D/M/p$ queueing system in this section.

A. The effect of K

In this subsection we study the effect of the reservation window size, K , on system performance for two values of m , 2 and 8. Fig. 3 plots three metrics, call congestion, link utilization, and mean scheduling delay. For a given value of m , which is determined by the per-VC bandwidth granularity used for sharing the link, our model allows a designer to select an appropriate value for the reservation window size.

For example, if we choose a system load/channel of 1 (since it is a blocking system, this load can be greater than 1), and want to run the system at a call congestion level of 4%, then if $m = 2$, we need a reservation window of 8 time intervals, but if $m = 8$, this number is only 4 time intervals. Including mean scheduling delay in our consideration, we see that if $m = 8$, the call congestion and utilization plots for $K = 4$, $K = 8$, and $K = 16$ are almost overlapping, while the mean scheduling delay increases significantly as K increases. Thus $K = 4$ is the right choice if $m = 8$. In summary, these results show that increasing reservation window size beyond a certain level is actually detrimental to system performance. We should choose the smallest K value to achieve a desired call congestion level at the expected system load so that the mean scheduling delay remains low (especially at high loads).

B. System design

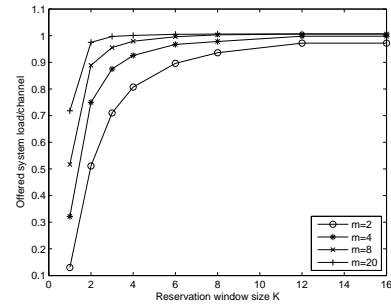


Fig. 4: Offered system load vs. reservation window size.

Consider a system designer who wants to know the payoff, with respect to the multiplexing factor on a shared link, by increasing the reservation window size. Our model can be used to help answer this question. Fig. 4 plots the offered system load/channel against K for a fixed call congestion, 1%, for different values of m . From this figure, we observe that for a shared link with $m = 4$, by increasing K from 2 to 4, the system load/channel can be increased from 75% to 93%. This is quite significant in that it allows for a 23.5% increase in the number of endpoints multiplexed on to the link.

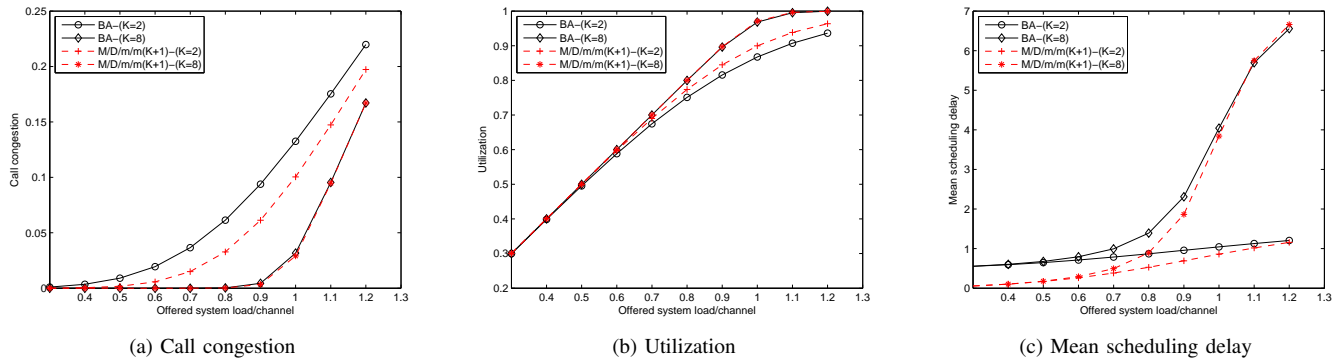


Fig. 5: The comparison of *BA-First* mode and $M/D/m/p$ queueing model.

C. *BA-First* vs. $M/D/m/p$

Recognizing the “queueing” nature of the *BA-First* mechanism with its K reservation window intervals in which the system can hold calls, we decided to compare it with an appropriate queueing system. Given our assumptions of Poisson arrivals and deterministic service times, the comparable queueing system is the $M/D/m/p$ system. A *BA-First* system with parameters m and K is comparable to a $M/D/m/m(K+1)$ queueing system to have an equal number of buffer locations. As there is no existing analytical solution for the $M/D/m/p$ system (to our knowledge), we implemented a simulation model.

Fig. 5 shows that the call congestion and link utilization metrics are almost the same for both systems. The mean scheduling delay of the *BA-First* mode is roughly half a time interval higher than that of the $M/D/m/m(K+1)$ queueing system when the system load is low. This is due to the fractional part of the scheduling delay in the *BA-First* system, in which calls cannot be served immediately in the arrival intervals. At high loads, the mean scheduling delay is almost the same in the two systems. In summary, our model, designed for the *BA-First* scheme, can be used as an approximate solution for an $M/D/m/p$ queueing system at moderate-to-high loads.

V. CONCLUSIONS

While it is difficult to create an exact analytical model for book-ahead (BA) bandwidth schedulers in general, we were able to model a particular BA system using a non-homogeneous CTMC. We call our BA mechanism *BA-First*, because it returns the earliest time interval in which a channel is available. Observing the system at the discrete time-interval boundaries, we extracted an embedded DTMC, and solved it for the steady-state probabilities. Using these, we obtained solutions for significant metrics, such as call congestion, link utilization, and mean scheduling delay. We demonstrated the use of the model as a design tool with numerical results. For example, with the model, a designer can determine that a 23.5% increase in system load can be accommodated, while maintaining a 1% call congestion rate when K , the size of

the reservation window, is increased from 2 to 4 for a 2-channel link. Additionally, we showed that this model is a good approximation for an $M/D/m/p$ queueing system at moderate-to-high loads.

ACKNOWLEDGMENT

This work was carried out under the sponsorship of NSF ITR-0312376, NSF ANI-0335190, and DOE DE-FG02-04ER25640 grants.

REFERENCES

- [1] ATLAS. [Online]. Available: <http://atlas.ch/>
- [2] LambdaVision. [Online]. Available: <http://www.evl.uic.edu/cavern/lambdavisio/>
- [3] GEANT2. [Online]. Available: <http://www.geant2.net/>
- [4] Japan Gigabit Network. [Online]. Available: <http://www.jgn.nict.go.jp/>
- [5] ESnet4. [Online]. Available: <http://www.es.net/ESnet4/>
- [6] Internet2 Network. [Online]. Available: <http://www.internet2.edu/network/>
- [7] UltraScience Net. [Online]. Available: <http://www.csm.ornl.gov/ultranet/>
- [8] On-demand secure circuits and advance reservation system (OSCARS). [Online]. Available: <http://www.es.net/oscars/index.html>
- [9] Dynamic resource allocation via gmpls optical networks (DRAGON). [Online]. Available: <http://dragon.east.isi.edu/>
- [10] T. L. N. Rao and C. Guok, “Integrated control plane and scheduler developments,” Meeting of Joint Engineering Taskforce, June 2006.
- [11] A. G. Greenberg, R. Srikant, and W. Whitt, “Resource sharing for book-ahead and instantaneous-request calls,” *IEEE/ACM Trans. Netw.*, vol. 7, no. 1, pp. 10–22, 1999.
- [12] O. Schelén and S. Pink, “Resource sharing in advance reservation agents,” *Journal of High Speed Networks*, vol. 7, 1998.
- [13] C. Bouras and K. Stamos, “An adaptive admission control algorithm for bandwidth brokers,” in *3rd IEEE International Symposium on Network Computing and Applications (NCA04)*, Cambridge, USA, Aug 2004.
- [14] L. Yuan, C. Tham, and A. L. Ananda, “A probing approach for effective distributed resource reservation,” in *Second International Workshop on Quality of Service in Multiservice IP Networks*, Milano, Italy, Feb. 2003.
- [15] L.-O. Burchard, “On the performance of computer networks with advance reservation mechanisms,” in *Proc. of the 11th International Conference on Networks*, Sydney, Australia, 2003.
- [16] R. Guérin and A. Orda, “Networks with advance reservations: The routing perspective,” in *Proc. of IEEE INFOCOM*, Tel Aviv, Israel, 2000.
- [17] J. T. Virtamo, “A model of reservation systems,” *IEEE Transactions on Communications*, vol. 40, pp. 109–118, 1992.
- [18] M. Schwartz, *Telecommunication networks: protocols, modeling and analysis*. Boston, MA: Addison-Wesley, 1986.
- [19] R. G. Gallager, *Discrete Stochastic Processes*. Dutch: Kluwer Academic Publishers, 1996.