Flexible Time-Windows for Advance Reservation in LambdaGrids

Neena R. Kaushik
Dept. of Computer Engineering
Santa Clara University
Santa Clara, CA 95053-0566, USA
nrkaushik@scu.edu

Silvia M. Figueira
Dept. of Computer Engineering
Santa Clara University
Santa Clara, CA 95053-0566, USA
sfigueira@scu.edu

Stephen A. Chiappari
Dept. of Applied Mathematics
Santa Clara University
Santa Clara, CA 95053-0560, USA
schiappari@scu.edu

ABSTRACT

Advance-reservation requests are an essential feature of LambdaGrids, where resources may need to be co-allocated at pre-determined times. In this paper, we discuss unconstrained advance reservations, which use flexible time-windows to lower blocking probability and, consequently, increase resource utilization. We claim and show using simulations that the minimum window size, which theoretically brings the blocking probability to 0, in a first-come-first-served advance reservation model without time-slots, equals the waiting time in a queue-based on-demand model. We also show, with simulations, the window sizes, which bring the blocking probability to its minimum, for an advance reservation model with time-slots.

Categories and Subject Descriptors

C.4 [**Computer Systems Organization**]: Performance of Systems – *Modeling techniques, Performance attributes, and availability.*

General Terms

Algorithms, Performance, Theory.

Keywords

Advance Reservation, Flexible Time-Windows, Scheduling, and LambdaGrids.

1. INTRODUCTION

A LambdaGrid [5] consists of a grid infrastructure, in which resources (e.g., computers, storage, and visualization engines) are interconnected by an optical network, and the optical channels are scheduled along with the other resources. Advance reservation of grid resources guarantees that they are available at pre-determined times. Advance reservation requests can be of two types: constrained and unconstrained, as mentioned in [1]. Unconstrained advance-reservation requests give the scheduler flexibility since the requests can be scheduled either at the start time or at any time within the specified window.

This paper shows that flexibility enables an advance reservation scheduling model to provide blocking probability close to zero, a behavior similar to a queue-based on-demand model, in which requests eventually are granted if they wait in the queue long enough.

SIGMetrics/Performance'06, June 26–30, 2006, Saint Malo, France.
ACM 1-59593-330-4/06/0006.

Hypothesis 1:

The value of the average window size, which will theoretically lower the blocking probability to zero, in the first-come-first-served advance-reservation scheduling domain, is the same as the mean waiting time in a queue-based on-demand system when $\lambda/\mu < 1$, i.e., $\rho < 1$, where λ is the average arrival rate, μ is the average service time, and ρ is the average traffic intensity.

2. ADVANCE RESERVATION MODELS

In this section, we show results for the M/M/1 system, in which the arrival and holding times follow an exponential distribution. The simulations were performed with ρ varying form 0.2 to 0.8, $\lambda = 0.2$, and the value of μ adjusted using λ/ρ. The average wait time per service interval is $\rho/(1-\rho)$ and was obtained in [3]. The window size is the average obtained from 50 different traces.

2.1 Model without time-slots

In this case, no time-slots were used, and the average window size per trace was calculated as the sum of the *window size flexibility*, i.e., the time needed for each request to be accepted, experienced by each user request, divided by the total number of requests.

2.2 Model with time-slots

Since using time-slots for managing the advance reservation allocation database is a practical alternative, we extended the model above to use time-slots. Each time-slot consisted of one hour. The start time of the request was aligned to the start of the next time-slot. The service time of each request was rounded to an integral number of time-slots. The average window size was calculated as the sum of the window flexibility experienced by each user request, divided by the total number of requests, where each request could be flexible to at most 1,000 hours. Note that, according to the simulations, for all the 50 traces and for different values of ρ, a window size of 1,000 led to a blocking probability of zero. Therefore, the maximum window size flexibility was chosen to be 1,000 hours.

From Figure 1, we observe that the window sizes for an advance reservation system without time-slots match the corresponding on-demand queue wait times. The window sizes obtained with time-slots are higher than the ones for the model without time-slots since time-slots introduce wastage, because each request is assigned an integral number of time-slots. Note that the unit used for the window size is hours, and the length of each time-slot is one hour.

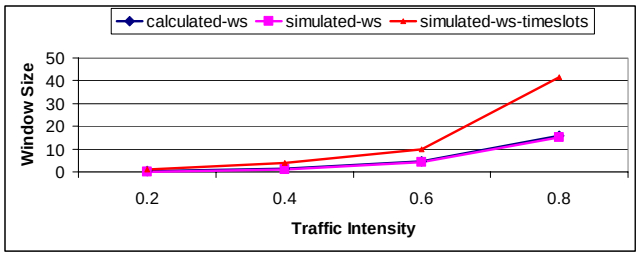

Figure 1: Window sizes from calculations, simulations without time-slots, and simulations with time-slots

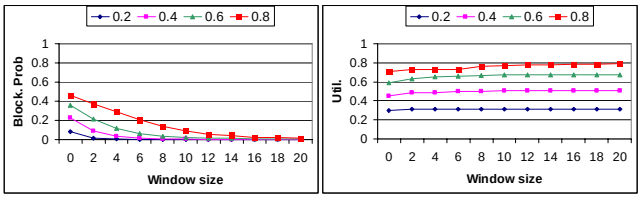

Figure 2: Window sizes (0-20)

Figure 2 shows, for a single trace, the effect of increasing the window size on both the blocking probability and the resource utilization. Note that, while the blocking probability decreases, the resource utilization increases.

3. SIMULATION RESULTS

We have simulated advance reservation scheduling using M/M/1, M/D/1, M/B/1, B/M/1, and B/B/1 types of arrival and service times, using a single server. These are standard notations [3], but note that we have replaced 'G' (which is used in the standard notation) by 'B' (bounded Pareto), since recent research on Internet traffic indicates heavy-tailed arrival and service times [4]. The probability density function for bounded Pareto distribution is defined as

$$f(x) = (\alpha k^{\alpha}/(1-(k/p)^{\alpha}))x^{(-\alpha-1)}, \; k \le x \le p, \qquad (1)$$

as mentioned in [2]. In our simulations, the parameters for the heavy tailed distribution for the service times were $\alpha = 1.7$, $k = 1$, and $p = 1{,}000$, with the mean being 2.401 for the M/B/1 case [2]. In the B/M/1 case, $\alpha = 0.9$ [4], $k = 1$, and $p = 1{,}000$ for the arrival times. In both cases, we used $p = 1{,}000$ since our allocations were for 60 weeks, i.e., 10,080 hours, and we wanted p to be, most of the time, much less than the length of the reservation window, so that a large number of requests would fall in the reservation window, to allow us to study the effect of the length of the flexible window on the blocking probability.

The following subsections present the results obtained for M/B/1, and B/M/1. We have omitted other results due to space constraints.

3.1 Exponential Inter-Arrival and Bounded Pareto Service Times

In this case, the value of ρ was varied from 0.2 to 0.8, where the arrival times followed an exponential distribution, and the service times followed a bounded Pareto distribution. The values of λ were 0.0833, 0.1666, 0.25, and 0.333 for the four cases presented in Figure 3, which shows the effect of increasing the window size. The service time values were adjusted as per values of λ and ρ using $\rho = \lambda/\mu$.

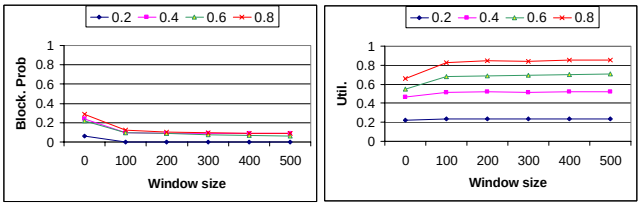

Figure 3: Window sizes (0-500)

3.2 Bounded Pareto Inter-Arrival and Exponential Service Times

In this case, the value of ρ was varied from 0.2 to 0.8, where the arrival times followed a bounded Pareto distribution, and the service times followed an exponential distribution.

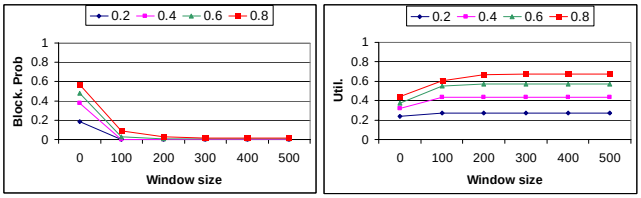

Figure 4: Window sizes (0-500)

The values of μ were 0.5572, 0.2785, 0.1857, and 0.1392 for the four cases shown in Figure 4, which also shows the effect of increasing the window size.

Note that, the results presented in Figures 3 and 4 show that increasing the window size leads to a decrease in the blocking probability (to its minimum) and to an increase in the resource utilization, corroborating the results presented in Figure 2.

4. CONCLUSION

In this paper, we have shown that, for an M/M/1 system, the window sizes in an advance reservation model without time-slots, match the corresponding average waiting times in an on-demand queue-based system. We have also shown that increasing window sizes, in an advance reservation model with time-slots, decreases the blocking probability and increases the resource utilization. Our results are important in contributing to improve user satisfaction while increasing resource utilization.

REFERENCES

[1] S. Figueira et al. DWDM-RAM: Enabling Grid Services with Dynamic Optical Network. In *IEEE CCGRID/GAN* 2004.

[2] M. Harchol-Balter, M. Crovella and Cristina Murta. On Choosing a Task Assignment Policy for a Distributed Server System. In *Journal of Parallel and Distributed Computing*, Vol. 59, no. 2, pp. 204-228, Nov. 1999.

[3] L. Kleinrock. Queuing Systems, Volume I: Theory. *Wiley-Interscience Publications*, John Wiley & Sons, 1975.

[4] V. Paxson and S. Floyd. Wide-area Traffic: The Failure of Poisson Modeling. In *IEEE/ACM Transactions on Networking*, 3(3):226-244, 1995.

[5] R. Wu and A. Chien. GTP: Group Transport Protocol for LambdaGrids. In *IEEE/ACM CCGrid 2004*