Grid Network Dimensioning by Modeling the Deadline Constrained Bulk Data Transfers

Kashif Munir
Institute of Computer Science
University of Innsbruck, Austria
E-mail: Kashif.Munir@uibk.ac.at

Pascale Vicat-Blanc Primet
LIP Laboratory, INRIA
University of Lyon, France
E-mail: Pascale.Primet@inria.fr

Michael Welzl
Institute of Computer Science
University of Innsbruck, Austria
E-mail: Michael.Welzl@uibk.ac.at

Abstract—**Grid applications need to move large amounts of data between distributed resources within deterministic time frames. In most cases it is possible to specify the volume and the deadline in advance. We formally define and analyze a mechanism of network reservations of bulk data transfer requests having opportunistic utilization of residual network capacity and analyze it using an $M/M/1/N-RPS$ queue. We compare the analytical results of our mechanism with the analytical results obtained from an $M/M/N/N$ queue, for the case in which there is no opportunistic sharing of residual capacity. We validate the analytical results through simulations in C++. The analytical model is also validated against *ns-2* simulations taking into account network level details (*IP* and *TCP* protocols), showing remarkably good coherence even under heavy loads. The model is orders of magnitude faster than simulation, which enables its application for capacity planning of Grid networks (also known as network dimensioning), and to enforce Connection Admission Control (*CAC*) under the practical hypothesis of a dominating bottleneck on the transfer route.**

Keywords: Grid, CAC, TCP, Modeling, PS (Processor Sharing), Bulk data transfers, Network dimensioning.

I. INTRODUCTION

Transporting large volumes of data in e-science has been discussed in [1], [2]. The Large Hadron Collider (*LHC*) facility at *CERN* [3] is expected to generate petabytes of experimental data every year, for each experiment. In addition to the large volume, as noted in [4], e-scientists routinely request schedulable high-bandwidth low-latency data transfers with known characteristics. A new generation of user controlled optical networks is being deployed to support e-science. In fact, end-2-end lightpath management has already been put into practice e.g., in the CA*net4 national research network [5].

Grid computing enables the virtualization of distributed computing and data resources such as processing, storage capacity and network bandwidth to provide a user with a unified view of the powerful computing system. It is therefore a major effort in Grid computing to hide some of the complexity from the programmers of Grid applications, which requires mechanisms to be in place for automatically distributing parts of applications — so-called schedulers, which work best if the underlying system exhibits a deterministic behavior. This can be attained by reserving resources such as CPUs and memory on machines (Advance Reservation); the underlying connection infrastructure being the Internet (or a specific part thereof), fully deterministic behavior can only be seen if such reservations include the network. These reservations have properties which make them somewhat different from the classical per-flow guarantees that have been demanded for multimedia services — the service may not be used immediately after its reservation and the flows are elastic.

The need of having *CAC* for elastic traffic has been demonstrated in [6], [12] using *PS* models. Without getting into complex details of packet level traffic characteristics, the *PS* models are mostly able to successfully capture the elastic properties of traffic generated by closed loop transport protocols like *TCP*.

The paper addresses the need of network dimensioning for bulk data transfers in Grid networks. Traditional dimensioning for telephone circuits uses the Erlang blocking model for multirate circuits [7]. These formulas assume constant rate connections suitable for the traditional circuits. The dimensioning problem presented in the paper is just a piece of the overall network design problem in a computer network. We focus on a single link and assume that all of the connections share the bottleneck link. Bandwidth dimensioning for elastic traffic in a single link's case is studied in [8] to satisfy a performance objective based on the mean probability of the per-connection bandwidth in high-speed data networks. The issue of dimensioning Internet access lines for elastic traffic is discussed in [9] using a $M/G/R-PS$ model characterizing *TCP* traffic at flow level. In our work, we take the deadline constraint of Grid bulk data transfers into account.

Our contribution is that we have used the $M/M/1/N-RPS$ queue model to predict the blocking probability of bulk data transfer requests and by applying the model we have provided a useful basis for network dimensioning in Grid. In Section II, we briefly describe the related work. In Section III, we formulate the problem. In Section IV, we present the analytical solution of the single link. The performance evaluation, where we analyze the blocking probabilities, validate the results through simulations and plan the network capacity is shown in Section V. We conclude in Section VI.

II. RELATED WORK

Since our work consists of modeling of elastic data transfers in Grid networks, we describe the related work by accordingly dividing it into the following sections.

A. Modeling of Elastic Traffic

In [10], performance modeling of elastic traffic is studied. The authors model the fair sharing bottleneck with an $M/G/1$ processor sharing queue to approximate the mean throughput achieved by *TCP*. The authors also demonstrate that flow-level statistical bandwidth sharing models can accurately predict the results of *ns* packet-level simulations. The difference with our approach is the deadline constraint which is taken into account to calculate the blocking probability of requests sharing a common bottleneck.

CAC for elastic traffic and fairness issues are discussed by J. Roberts and L. Massoulie in a number of publications, e.g., see [11], [12]. We extend the work by proposing a computational model to predict the deadline-constrained elastic data transfers in Grids along with providing network capacity planning by applying the model.

B. Grid Network Reservations

QoS guarantees are required by Grid applications [13], [14]. The problem of network resource reservation [15] has been proposed to be studied within the Grid scope targeting deadline requirements of bulk data transfer requests. An example for a Grid toolkit that supports such mechanisms is Globus with its *GARA* resource allocation component [16]. For elastic requests in [19], the method of [17] or [18] is used to adjust the bandwidth or duration to satisfy the requester. However for a fixed request in [19], the only way to avoid being rejected is to adjust the bandwidth of admitted malleable requests. The trouble with this scheme is the extra overhead in finding and adjusting the admitted reservations which may be modified. The multi-interval scheme, presented in [20], avoids this trouble. Rather than allocating a constant rate to each request, the *BDTS* multi-interval approach allocates a bandwidth profile, defined as a step function. The offline scheme proposed in [20], which can be adapted for online problem, ensures that a request will not be rejected if there is at least one feasible solution to accept it.

A time-slot based approach for scheduling the elastic and streaming requests is described in [21]. In this approach a flow sends at a fixed rate during a time slot and hence does not make use of unreserved bandwidth slots in that time slot. In this approach, the fixed rate can only be computed between a low rate and a high rate specified by a request, however the flows in the approach described in [22] can opportunistically utilize the available capacity.

In [26] periodic re-optimization is used to determine new bandwidth allocation in optical networks. However, it is not assumed that users make advance reservations with requested end times. As a result, [26] does not have the admission control step. In *WDM* optical networks, some authors [27], [28] study advance reservation and re-routing of lightpaths at the granularity of a wavelength, but they do not consider the deadline constraint.

III. PROBLEM FORMULATION

Let us define:

Definition 1: A *network* is represented by a connected graph $G(V, E)$, consisting of node set V and edge set E, with edge capacity $\mu(e) : E \rightarrow \mathbb{R}^+ - \{0\}$, where $\mathbb{R}^+$ is the set of nonnegative real numbers.

Definition 2: A *path* $\phi = (v_0, v_1, \ldots v_h)$ is a finite sequence of nodes, such that for $0 \leq i < h$, $(v_i, v_{i+1}) \in E$.

Definition 3: A *bulk-data transfer task* $r = (\nu_r, \omega_r, \Phi_r)$ is a triple, where ν_r is the *volume* of the dataset, $\omega_r = [\eta_r, \psi_r]$ is the *active window* (from *arrival time* η_r to *deadline* ψ_r), $|\omega_r| = \psi_r - \eta_r$ is the length of active window, and $\Phi_r = \{\phi_r^1, \phi_r^2, \ldots, \phi_r^{|\Phi_r|}\}$ is the set of paths connecting source s_r and destination d_r.

Definition 4: We evaluate the blocking probability *BP* as the ratio of rejected requests to offered requests at any given time t:

$$BP = \frac{1}{|R|} \left(|R| - \sum_{r \in R} x_r \right) .$$

The list of the symbols used in the problem formulation is summarized in Table I.

Table I
SUMMARY OF THE SYMBOLS USED IN PROBLEM FORMULATION

Symbol	Meaning
r	A reservation request
ν_r	Volume of r
η_r	Arrival time of r
ψ_r	Deadline of r
$\omega_r = [\eta_r, \psi_r]$	Active window from η_r to ψ_r
$|\omega_r| = \psi_r - \eta_r$	Length of the active window
s_r	Source of r
d_r	Destination of r
Φ_r	Path from s_r to d_r
MRR_r	Minimum Required Rate of r obtained by dividing ν_r with $|\omega_r|$
β_{Φ_r}	Capacity of bottleneck link of path Φ_r
R	Set of submitted requests r in time interval T
$Q(t)$	Set of accepted requests r in time interval T
σ_r	Scheduled start time of r
τ_r	Finishing time of r
$C_r(t)$	Residual capacity along Φ_r at time t
$\gamma_r(t)$	Effective rate of r at time t
x_r	$x_r = 1$ if r is accepted and $x_r = 0$ if it is not

Request r is accepted at time $\sigma_r = t$ and added to set $Q(t)$, only if path Φ_r can devote a fraction of its capacity β_{Φ_r}, that is, MRR_r, to request r from time σ_r to time $\psi_r = \sigma_r + \dfrac{\nu_r}{MRR_r}$.

It is straightforward to understand that a transfer is guaranteed to finish before its deadline if the transmitting rate never goes below its lower limit which is MRR_r. The capacity of the path poses a limit on the number of scheduled requests. The admission-control and request constraints are then stated formally as the following:

$$MRR_r * (\psi_r - \sigma_r) = \nu_r, \quad \forall r \in Q(t); \qquad (1)$$

$$\eta_r \leq \sigma_r, \quad \forall r \in Q(t); \qquad (2)$$

$$\sum_{r \in Q(t)} MRR_r \leq \beta_{\Phi_r}, \quad \forall t; \qquad (3)$$

Equation (1) gives the volume constraints, Equation (2) gives the starting time constraints, and Equation (3) gives the path

capacity constraints.

Definition 5: $C_r(t) = \beta_{\Phi_r} - \sum_{r \in Q(t)} MRR_r$. The residual capacity along the path can be on any physical link composing the path and may change in time when requests finish earlier than their deadline.

In the bandwidth management proposal in [22], the accepted requests opportunistically grab more bandwidth during execution by dividing C_r equally among the requests. The effective rate $\gamma_r(t)$ will then be in the interval of $[MRR_r, \beta_{\Phi_r}]$ for all t in $[\sigma_r, \psi_r]$. This means that an accepted request r can grab more bandwidth if available, thus the actual finishing time of a request r will be $\tau_r \leq \psi_r$. When request r is finished it is removed from $Q(t)$ the set of active accepted requests.

The resource-sharing and request constraints are then stated formally as follows:

$$\int_{\sigma_r}^{\tau_r} \gamma_r(t)dt = \nu_r, \quad \forall r \in Q(t); \tag{4}$$

$$\tau_r \leq \psi_r, \quad \forall r \in Q(t); \tag{5}$$

$$\gamma(t) : \omega_r \rightarrow \mathbb{R}^+ \tag{6}$$

Equation (4) is for opportunistic bandwidth-usage constraints. Equation (5) formulates the finishing-time constraints. Equation (6) gives the opportunistic bandwidth solution space.

MAX-REQ : Under the constraints in Equations (1), (2), and (3), one may maximize the ratio of the number of accepted requests to that of total requests within a given time interval T. The objective function, referred to as MAX-REQ, is:

$$\text{MAXIMIZE} \sum_{r \in R} x_r$$

RES-UTI: Under the same constraints, one may maximize the resource-utilization ratio, that is, the ratio of granted resources to total resources within T. The objective function, referred to as RES-UTI, is:

$$\text{MAXIMIZE} \frac{\sum_{r \in R} x_r * MRR_r}{\beta_\phi}$$

where numerator $\sum_{r \in R} x_r * MRR_r$ is the total bandwidth that has been assigned to requests.

RES-UTIop: The objective function, referred to as RES-UTIop, is:

$$\text{MAXIMIZE} \frac{\sum_{r \in R} x_r * \frac{\int_\eta^\psi \gamma_r(t)d(t)}{|\tau_r - \sigma_r|}}{\beta_\phi}$$

$$= \frac{\sum_{r \in R} x_r * \frac{\nu_r}{|\tau_r - \sigma_r|}}{\beta_\phi}$$

where numerator $\sum_{r \in R} x_r * \frac{\nu_r}{|\tau_r - \sigma_r|}$ is the total bandwidth opportunistically used by the accepted requests within time interval T.

IV. ANALYTICAL MODELING

The maximization objectives defined in Section III are met for a single path by the mechanism described in [22]. That work describes a *CAC* mechanism together with rationales of the path sharing mechanism, based on simplified assumption
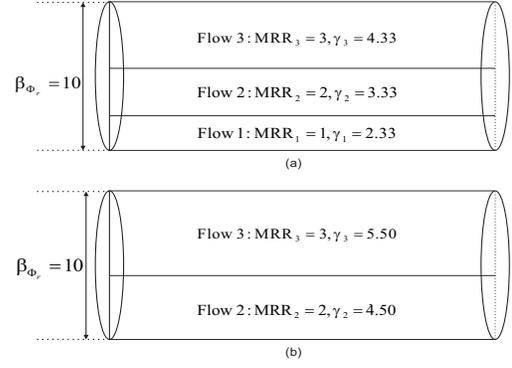

Figure 1. Rates allocation to the flows: (a) Before the termination of the first flow (b) After the termination of the first flow

on TCP-based transmissions. We are interested here in finding a suitable model for this mechanism that enables network planning and allows for the prediction of the performance of data transfers in Grid computing.

The mechanism guarantees an MRR_r of *x* bits per second to a flow so that it can meet its deadline. We assume that the available bottleneck link capacity is equally shared among the flows sharing the link. For the model, we consider an ideal case of the sharing of available capacity among existing flows sharing the link and do not consider the network conditions which have communication and computation losses and overheads. Practically, after admission, the flow starts with its MRR_r and gradually increases its rate using a congestion control scheme [29], [30], [31] in such a way that at any time its rate does not go below its MRR_r.

We briefly explain the working of the mechanism. Suppose we have a bottleneck with $\beta_{\Phi_r} = 10$ *Gbps*. Suppose there are three flows in the network. Let the MRR_r of the first, second and the third flow are 1, 2 and 3 *Gbps* respectively. There is a residual capacity of $10 - (1+2+3) = 4$ *Gbps* which is equally divided among the three flows. Which means each flow will increase its transfer rate by $4/3$ *Gbps*. The transfer rates of the three flows become 2.33, 3.33 and 4.33 *Gbps* respectively as shown in Figure 1. Suppose the first flow terminates, it means now the residual capacity is $10 - (2+3) = 5$ *Gbps* which is equally divided between the second and the third flow due to residual processor sharing and their transfer rates become 4.5 and 5.5 *Gbps* respectively as show in Figure 1.

The problem is to find the probability that a new reservation request is rejected or find a probability that a new reservation request finds N requests in the system, where N is the maximum number of flows in the network. The analytical model has been developed for the case of immediate reservations only, which is the most interesting as normally applications want to start transmission as soon as data is available. Results for mixed reservation requests — immediate and advance — were obtained by simulation and presented in [22], [23], [24], [25].

A. M/M/1/N-RPS Model

We model the path as a constant capacity transmission server with rate $\beta_{\Phi_r}(= C)$, which services customers (the requests r) in parallel with a guaranteed minimum rate. Since there is no practical evidence of service time distribution for bulk data transfers in Grids, we take Markovian services. It is a known fact that the formulas to calculate the blocking probabilities in an $M/G/1 - PS$ queuing system is insensitive to the nature of the services distribution [10]. With Markovian assumptions for arrivals and services, this system is an $M/M/1/N - RPS$ queue, whose meaning in the Kendall notation is explained in Table II. The summary of the new symbols introduced in this section is given in Table III.

Table II
$M/M/1/N - RPS$ QUEUE

Symbol	Meaning
M	Markovian arrivals
M	Markovian services
1	The number of servers is 1.
N	As there is no queue, the maximum number of flows in the system is N.
RPS	Processor sharing of the remaining capacity C_r.

Table III
SUMMARY OF THE NEW SYMBOLS USED IN MODELING

Symbol	Meaning
λ	Arrival rate
μ	Service rate
V	Mean ν_r
MRR	Mean MRR_r
$d(\psi_r - \sigma_r)$	Transfer duration of a flow
$C = \beta_{\Phi_r}$	Bottleneck link capacity
θ	Number of flows in a network at time t

The problem can be modeled using the birth-death process of an N state Continuous Time Markov Chain (CTMC) as shown in Figure 2. Each state from 0 to N represents the number of flows in the network. There is no queue in the system due to which a new reservation request is rejected if there are N flows in the network.

The birth coefficient can be written as:-

$$\lambda_k = \left\{ \begin{array}{ll} \lambda & 0 \le k < N \\ 0 & \text{otherwise} \end{array} \right\} \tag{7}$$

The death coefficient can be written as:-

$$\mu_k = \left\{ \begin{array}{ll} \mu & 1 \le k \le N \\ 0 & \text{otherwise} \end{array} \right\} \tag{8}$$

Due to processor sharing, the service rate is same. When there is a less number of flows in the network than the maximum number of flows, C_r is fairly shared and this decreases the service times of the existing flows. The blocking probability of a new reservation request, a well known result, is:-
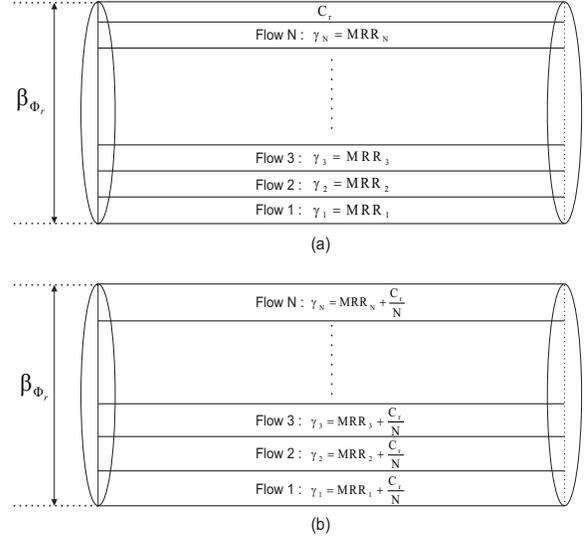

Figure 3. Rates allocation to N flows, $\theta = N$: (a) Without sharing of C_r (b) With equal sharing of C_r

$$\pi_N = \frac{\left(\frac{\lambda}{\mu}\right)^N}{\sum_{n=0}^{N}\left(\frac{\lambda}{\mu}\right)^n} \tag{9}$$

where,

$$\mu = \frac{C}{V} \tag{10}$$

Due to the full system utilization and due to processor sharing queue and fair sharing of C_r, the system service rate is constant.

Also we have,

$$MRR_r = \left\lfloor \frac{\nu_r}{d} \right\rfloor \tag{11}$$

$$N = \left\lfloor \frac{C}{MRR} \right\rfloor; \quad where \; MRR = mean \; MRR_r \tag{12}$$

From Equation (12), it is clear that when MRR is allocated to each of the N flows, we get C_r. Figure 3 shows the allocation of rates to N flows with and without equal sharing of C_r.

C_r at t is:

$$C_r = C - (MRR \times \theta), \quad 0 \le C_r \le C; \tag{13}$$

so the effective transfer rate of a flow at t becomes $\left(MRR_r + \frac{C_r}{\theta}\right)$.

The expected number of jobs E in the system is given below. Equation (14) is later used in Section V to calculate the aggregate *TCP* throughput.

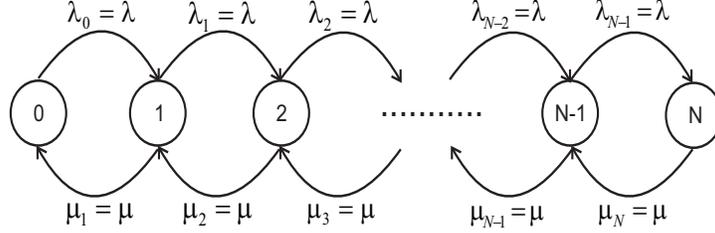$$E = \sum_{k=0}^{N} k\pi_k \tag{14}$$

Figure 2.   CTMC of the system

In the non-opportunistic case, C_r is not shared and each flow transfers at its MRR_r. To model this case we can take N as the maximum number of servers in the system. Each flow in this case can be served by one server only. Hence the system can serve at most N flows. The system in this case is not always fully utilized and C_r is wasted. A new reservation request is rejected if all N servers all busy. The blocking/rejection probability of a new reservation request for this case, also known as *Erlang B* formula, is obtained from the $M/M/N/N$ queue and is given below:

$$\pi'_N = \frac{\frac{\left(\frac{\lambda}{\mu'}\right)^N}{N!}}{\sum_{n=0}^{N} \frac{\left(\frac{\lambda}{\mu'}\right)^n}{n!}}, \quad where \ \mu' = \frac{1}{d}; \qquad (15)$$

In the next section, we compare the analytical results with the simulations results for both cases i.e., with and without sharing of C_r. We validate the analytical results and apply the model to plan the network dimensioning. We use Equation (9) and Equation (15) to calculate the analytical results with and without sharing of C_r respectively.

V. Performance Evaluation

The objectives of the performance evaluation are:

- Measuring the reduction in the blocking probability of requests due to the opportunistic sharing of residual capacity.
- Validating the analytical results through simulations with and without taking network level details into account.
- Network capacity planning to keep the blocking probability below a certain threshold.

We consider a single bottleneck link configuration. It should be noted here that our objective is to simulate the online reservation process and not the network conditions. In our earlier work [22], [23], [24], [25] we have simulated the mechanism keeping in view the communication and computation losses and overheads of network as well as the capacity wastage due to a congestion control protocol. The input parameters of the experiments are given in Table IV. We take $\lambda = 0.25$ for all cases.

A. Opportunistic vs. Non-opportunistic and Validation

Figure 4 shows that *BP* of requests in the case of processor sharing is significantly less than the case in which no processor sharing is used. The analytical results have been validated by
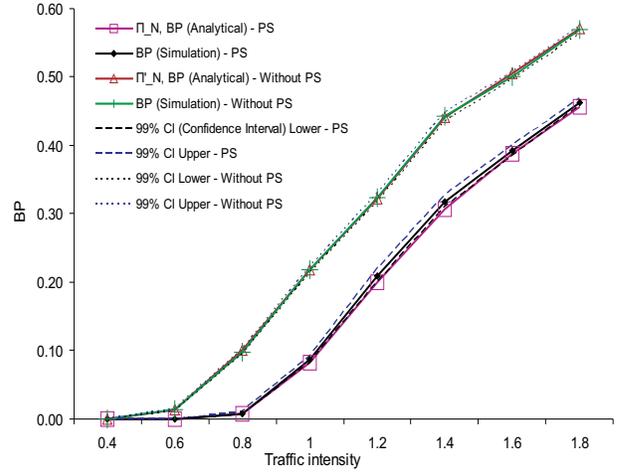

Figure 4.   Blocking probability (Analytical vs. Simulation) with and without processor sharing (PS)

the simulations as shown in Figure 4. The simulation code is written in *C++*. The simulations are run for the same traffic intensities. For each measurement, we run *10* simulations and take the average. Each simulation is run for *10,000* flows. The results in Figure 4 shows that the analytical results match the simulations.

B. Network Capacity Planning

In another experiment, we take the same traffic volumes per second as was used in the previous experiment and calculate the Minimum Required Capacity (*MRC*) of the link such that *BP* is less than a certain threshold. Algorithm 1 for the capacity planning is given below. The results of this experiment are shown in Figure 5.

C. Validation Through ns-2 Simulations Taking into Account Network Level Details

Finally, we validate the analytical model taking the network conditions into account. To estimate *TCP* throughput, we use Equation (14) to calculate the expected number of jobs *E* in the system which is used in a formula [32] to estimate aggregate throughput *x(E)* of *E* connections sharing a bottleneck link of fixed capacity *C* as given below.

Table IV
INPUTS OF THE EXPERIMENT

Measurement Nr. $-->$	1	2	3	4	5	6	7	8
V (in Mega Bits)	1600	2400	3200	4000	4800	5600	6400	7200
MRR (in Mbps)	32	48	64	80	96	112	128	144
Traffic intensity $\rho = \frac{\lambda}{\mu}$	0.4	0.6	0.8	1.0	1.2	1.4	1.6	1.8

Algorithm 1 Capacity planning

Procedure MinimumRequiredCapacity
Inputs: *V*, *C*, *MRR*, λ, Threshold of overall blocking probability:BP_{Thresh}
Output: Minimum Required Capacity:*MRC*

$C = 0$
$BP = 1$
while $BP \geq BP_{Thresh}$ **do**
    $C = C + 1$
    Generate the *CTMC(C)* of the system
    Calculate arrival λ and service μ rates
    Calculate *BP* using Equation (9)
end while
$MRC = C$

End Procedure

Figure 5.   Network capacity planning

Algorithm 2 Estimation of blocking probability of the mechanism combined with *TCP*

Procedure BP Estimation for TCP
Inputs: *V*, *C*, *MRR*, λ
Output: *BP*

$\epsilon = 1$
$C' = C$
$E = 1$
while $\epsilon > 0.01$ **do**
    $E' = E$
    Generate $CTMC(C')$ of the system
    Calculate E using Equation (14)
    Calculate BP using Equation (9)
    Compute $x(E)$ using Equation (16)
    $C' = C' \times x(E)$
    $\epsilon = ABS(E - E')/E$
end while

End Procedure

Figure 6.   Estimation of *BP* in an actual system
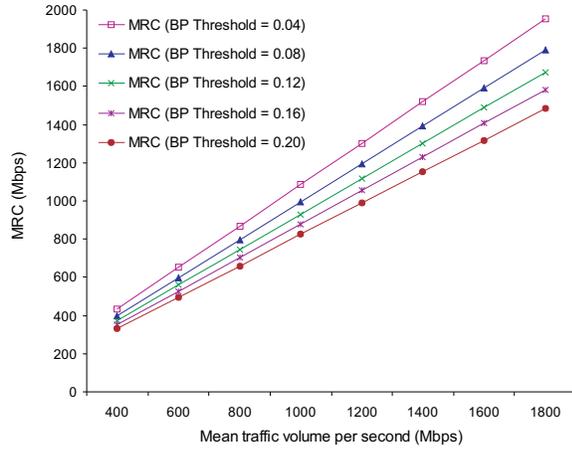
$$x(E) = C \left(1 - \frac{1}{1 + 3E} \right) \qquad (16)$$

Algorithm 2 estimates *BP* when the mechanism is combined with standard *TCP*.

To validate, we run *ns-2* simulations using the same traffic loads (intensities) by combining the mechanism with standard *TCP*. A single bottleneck link dumbbell network configuration is used for the simulations using *ns-2*. The bottleneck capacity is 1 *Gbps* and the bottleneck delay is set to 50ms. Drop Tail routers are used. The buffer size of the bottleneck link is set to *100* percent of the Bandwidth-Delay product. The packet size is set to *1500* bytes. The capacity of side links is *10 Gbps* and the delay of each side link is set to *2 ms*. For each measurement, we run *10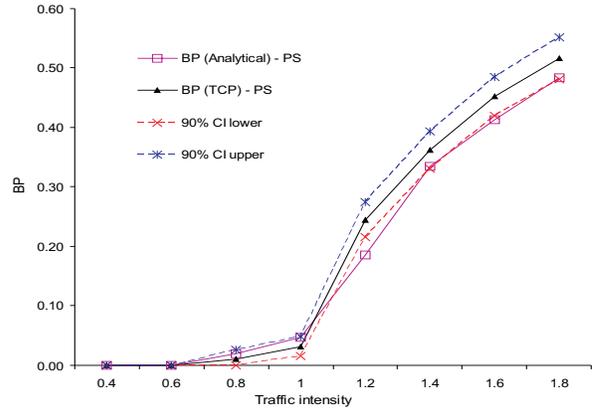* simulations and take the average of them. Each simulation is run for *100* flows. The *90* percent confidence interval curves are plotted along with the simulation curve. The blocking probabilities are shown in Figure 6.

The results show remarkably good coherence even under heavy loads. The analytical curve for *TCP* is not smooth due to rounding of *C'* and also due to using an approximation of the *TCP* throughput. The simulation curve shows that *BP* obtained from simulation is generally more than the analytically calculated *BP* for higher traffic intensities. When traffic intensity increases *TCP* performance deteriorates and consequently *BP* increases.

VI. Conclusion

In this paper, we have used the $M/M/1/N - RPS$ processor sharing queue with exponential arrivals and services. We have applied the model to calculate the blocking probability of requests, which yields quite accurate results as validated by the simulations. The results show that the opportunistic capacity utilization reduces the blocking probability of requests by a significant factor. The objectives given in Section III are met through opportunistic capacity sharing.

We have also shown that given the mean volume of requests per second, the model can be used to calculate the capacity of a link in order to have the blocking probability of requests below a certain threshold. The information is useful for a Grid network provider because over-dimensioning wastes precious bandwidth resources, while under-dimensioning generally leads to less satisfactory *QoS* perceived by Grid user.

The *ns-2* simulations were performed to check whether the model fits reality. We have seen that real rate-control methods like standard *TCP* combined with the *CAC* mechanism appear to provide a good approximation of the model, which shows that the model is representative of actual system behavior.

The results are encouraging to extend the modeling to a multi-service environment having classes of flows with heterogenous rate requirements. This point will be addressed in a further study.

References

[1] J. Bunn, and H. Newman, "Data-intensive grids for high-energy physics, in Grid Computing: Making the Global Infrastructure a Reality", F. Berman, G. Fox, T. Hey, Eds. John Wiley & Sons, Inc, 2003.

[2] H.B. Newman, M.H. Ellisman, and J.A. Orcutt, "Data-intensive e-science frontier research", Communications of the ACM, vol. 46, no. 11, Nov. 2003, pp. 68-77.

[3] CERN, http://www.cern.ch.

[4] T. DeFanti, C.D. Laat, J. Mambretti, K. Neggers, and B. Arnaud, "TransLight: A global-scale LambdaGrid for e-science", Communications of the ACM, vol. 46, no. 11, Nov. 2003, pp. 34-41.

[5] R. Boutaba, et al., "Grid-controlled lightpaths for high performance grid applications", Journal of Grid Computing, vol. 1, issue 4, 2003, pp. 387-394.

[6] S.B. Fredj, T. Bonald, A. Proutière, G. Régnié, and J.W. Roberts, "Statistical bandwidth sharing: a study of congestion at flow level", SIGCOMM, 2001.

[7] J.S. Kaufman, "Blocking in a shared resource environment", IEEE Trans. Commun., vol. 29, 1981, pp. 14741481.

[8] A.W. Berger, and Y. Kogan, "Dimensioning Bandwidth for Elastic Traffic in High-Speed Data Networks", IEEE/ACM Transactions on Networking, vol. 8, issue 5, Oct. 2000, pp. 643-654.

[9] Z. Fan, "Dimensioning Bandwidth for Elastic Traffic", Lecture Notes In Computer Science, vol. 2345, 2000, pp. 826-837.

[10] T. Bonald, and J. Roberts, "Performance modeling of elastic traffic in overload", Proceedings of the 2001 ACM SIGMETRICS international conference on measurement and modeling of computer systems, 2001, pp. 342-343.

[11] L. Massoulie, and J.W. Roberts, "Bandwidth Sharing and Admission Control for Elastic Traffic", http://www-sop.inria.fr/mistral/pub/massoulie.html, 1998.

[12] J.W. Roberts, "A survey on statistical bandwidth sharing", Computer Networks, vol. 45, issue 3, Oct. 2004, pp. 319-332.

[13] H. Zhang, K. Keahey, and B. Allcock, "Providing Data Transfer with QoS as Agreement-Based Service", International Conference on Services Computing (SCC 2004), Shanghai, China, Sep. 2004.

[14] I. Foster, A. Roy, and V. Sander, "A Quality of Service Architecture that Combines Resource Reservation and Application Adaptation", In 8th International Workshop on Quality of Service (IWQoS), 2000, pp. 181-188.

[15] I. Foster, M. Fidler, A. Roy, V. Sander, and L. Winkler, "End-to-end quality of service for high-end applications", Computer Communications, vol. 27, no. 14, 2004, pp. 1375-1388.

[16] I. Foster, C. Kesselman, C. Lee, R. Lindell, K. Nahrstedt, and A. Roy, "A Distributed Resource Management Architecture that Supports Advance Reservations and Co-Allocation", In 7th International Workshop on Quality of Service (IWQoS), London, UK, 1999, pp. 27-36.

[17] L. Burchard, H. Heiss, and D. Rose, "Performance issues of bandwidth reservations for grid computing", Proceedings of Computer Architecture and High Performance Computing, 2003, pp. 82-90.

[18] J. Xing, C. Wu, M. Tao, L. Wu, and H. Zhang, "Flexible Advance Reservation for Grid Computing", GCC 2004, pp. 241-248.

[19] L. Wu, J. Xing, C. Wu, and J. Cui, "An Adaptive Advance Reservation Mechanism for Grid Computing", PDCAT 2005, pp. 400-403.

[20] B.B. Chen, and P. Primet, "Supporting bulk data transfers of high-end applications with guaranteed completion time", IEEE ICC2007 International conference on computer communication, 2007.

[21] S. Naiksatam, and S. Figueira, "Elastic Reservations for Efficient Bandwidth Utilization in LambdaGrids", Elsevier's FGCS - The International Journal of Grid Computing: Theory, Methods and Applications, vol. 23, issue 1, 2007, pp. 1-22.

[22] K. Munir, S. Javed, M. Welzl, H. Ehsan, and T. Javed, "An End-to-End QoS Mechanism for Grid Bulk Data Transfer for Supporting Virtualization", IEEE/IFIP International Workshop on End-to-end Virtualization and Grid Management (EVGM 2007), San Jose, California, USA, Oct. 2007.

[23] K. Munir, S. Javed, and M. Welzl, "A Reliable and Realistic Approach of Advance Network Reservations with Guaranteed Completion Time for Bulk Data Transfers in Grids", ICST International Conference on Networks for Grid Applications (GridNets 2007), Lyon, France, Oct. 2007.

[24] K. Munir, M. Welzl, and S. Javed, "Scheduling Network Reservations with a Flexible and Reliable QoS Mechanism in Grids", EuroFGI Workshop on IP QoS and Traffic Control (EuroFGI-QoS2007), Lisbon, Portugal, Dec. 2007.

[25] K. Munir, S. Javed, M. Welzl, and M.M. Junaid, "Using an Event Based Priority Queue for Reliable and Opportunistic Scheduling of Bulk Data Transfers in Grid Networks", 11th IEEE International Multitopic Conference (INMIC 2007), Lahore, Pakistan, December 2007.

[26] R. Bhatia, M. Kodialam, and T.V. Lakshman, "Fast network re-optimization schemes for MPLS and optical networks", Computer Networks, vol. 50, no. 3, Feb. 2006, pp. 317-331.

[27] D. Banerjee, and B. Mukherjee, "Wavelength-routed optical networks: linear formulation, resource budgeting tradeoffs, and a reconfiguration study", IEEE/ACM Transactions on Networking, vol. 8, no. 5, Oct. 2000, pp. 598-607.

[28] E. Bouillet, J.F. Labourdette, R. Ramamurthy, and S. Chaudhuri, "Lightpath re-optimization in mesh optical networks", IEEE/ACM Transactions on Networking, vol. 13, no. 2, 2005, pp. 437-447.

[29] Y. Gu, and R. Grossman, "UDT UDP-based data transfer for high-speed wide area networks", Computer Networks, special issue on Hot topics in transport protocols for very fast and very long distance networks, 2007.

[30] D.X. Wei, C. Jin, S.H. Low, and S. Hegde, "FAST TCP: motivation, architecture, algorithms, performance", IEEE/ACM transactions on Networking 14 (6), 2007, pp. 1246-1259.

[31] I. Rhee, and L. Xu, "CUBIC: A New TCP-Friendly High-Speed TCP Variant", In Proceedings of PFLDnet, 2005.

[32] E. Altman, D. Barman, B. Tuffin, and M. Vojnovic, "Parallel TCP Sockets: Simple Model, Throughput and Validation", INFOCOM, Barcelona, Spain, April 2006.