Available at
www.ComputerScienceWeb.com
POWERED BY SCIENCE @ DIRECT®

Performance Evaluation 52 (2003) 59–84

PERFORMANCE EVALUATION
An International Journal

www.elsevier.com/locate/peva

Pricing in next generation networks: a queuing model to guarantee QoS

Mohamed Yacoubi[a], Maria Emelianenko[b], Natarajan Gautam[c,*]

[a] *Deloitte Consulting, 3, Place Ville-Marie, Bureau 300, Montréal, Que., Canada H3B 5K1*
[b] *Department of Mathematics, The Pennsylvania State University, University Park, PA 16802, USA*
[c] *Department of Industrial Engineering, The Pennsylvania State University, 310 Leonhard Building, University Park, PA 16802, USA*

Received 5 January 1999; received in revised form 21 August 2002

Abstract

We consider the scenario where users access Next Generation Networks via Network Access Providers (NAP). We assume that users belong to N different classes and the bandwidth received by each class is determined by a User-Share Differentiation (USD) scheme. According to USD, each class is guaranteed a minimum bandwidth and all users accepted into the NAP are allocated the minimum bandwidth corresponding to their class and any remaining bandwidth is shared according to the ratio of the minimum bandwidths of each class. We develop a queuing-based model and solve an optimization problem to determine the minimum bandwidth (defined in USD) for each of the N classes that maximizes the revenue of the NAP, subject to satisfying a request blocking performance guarantee.

Keywords: Bandwidth allocation; Multi-class traffic; Queuing model; Quality of service; Access providers

1. Introduction

In this paper, we consider Next Generation Networks, where users, with the aid of Network Providers, access the network for information transfer (non-real-time data files like images, graphics, animation and texts, web browsing, and real-time multimedia applications like telephony, chat sessions, videoconferencing, telemedicine, live web television etc.). The current Internet offers only best-effort service, i.e. on a first come first serve basis. However, real-time multimedia applications require stringent Quality of Service (QoS) guarantees from the network, meaning hard bounds on bandwidth, end-to-end delay and jitter, packet loss probability etc. For a network to provide performance guarantees, it has to reserve resources and exercise call admission control. Due to the increased demand of real-time multimedia applications

[*] Corresponding author. Tel.: +1-814-865-1239; fax: +1-814-863-4745.
E-mail addresses: myacoubi@dc.com (M. Yacoubi), emeliane@math.psu.edu (M. Emelianenko), ngautam@psu.edu (N. Gautam).

PII: S0166-5316(02)00180-3

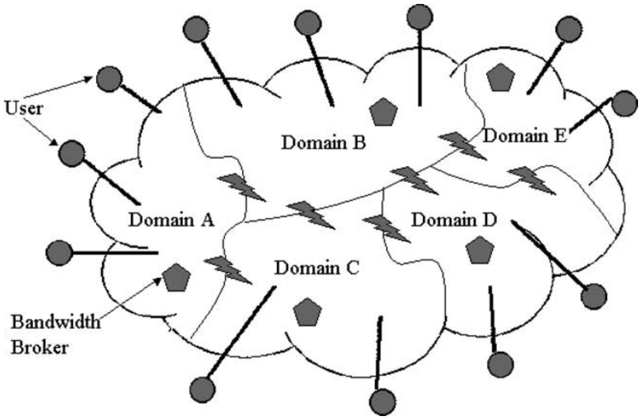

Fig. 1. Representing the Internet using domains.

in the last few years, lot of work has been done by the industry as well as the research community on providing end-to-end network guarantees. Providing end-to-end QoS guarantee can be broadly divided into three areas (see Fig. 1), call admission and resource reservation between the client and the network domain manager, intra-domain and inter-domain.

A domain (or AS, Autonomous System) is a part of the Internet under a single organization or Internet Service Provider (ISP). Currently the Differentiated Services (DS) architecture (RFC 2475, 2638) is the most popular framework for providing QoS. As per this model, every domain has a centralized network manager, known as the Bandwidth Broker (BB) [19], which is aware of the domain topology and status, using the underlying routing protocols. First the client (maybe a single user, corporate network or aggregated sub-networks) negotiates its QoS requirement, known as Service Level Agreement (SLA) negotiation, with the BB in its domain. Then the source domain BB negotiates resource allocation with the intermediate and destination domain BBs. We note here that currently these are the areas of active research in the networking community, and that there is no existing IETF standard for any of these mechanisms.

In this paper, we concentrate on resource negotiation and call admission between the client and the source domain BB. We assume that the negotiated bandwidth can be provided through the remainder of the network. To provide this guaranteed QoS, the Network Provider charges the users a price.

Consider users accessing information through the Internet. The users connect to the Internet via ISPs (see Fig. 2). Note that several independent users subscribe to an ISP and typically dial up using a telephone line. However, with the increasing bandwidth requirements, users are shifting to faster connections to the ISP such as cable modems. Users and the ISP agree on a contract, where the ISP guarantees the users a minimum level of network resources and performance, which we refer to as the QoS parameters. Typically, end-to-end delay, delay jitter, packet loss probability, bandwidth, and call-blocking probability, are the QoS parameters of interest.

Under the above-mentioned contract stipulations, the ISP must guarantee the negotiated QoS for the connections established (we will use the words connections and calls interchangeably to refer to successful access to the network by the users). In this paper, we concentrate on two of the above QoS parameters namely, required bandwidth and probability of blocking a connection.

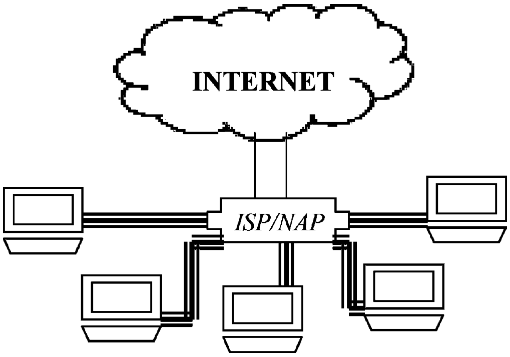

Fig. 2. The Internet, ISP and users.

The users accessing the network via the ISP are guaranteed a minimum bandwidth of the total available link capacity (determined via the SLAs negotiated by the BBs) for their connections. The network chooses an optimal sharing scheme for the different users of the total bandwidth (available link capacity) to fulfill connection requirements. In addition, the probability of rejection of connection requests (due to lack of resources) is kept below a negotiated level.

The scheme used by the Network Access Provider (NAP) to allocate bandwidths to the connections is based on a User-Share Differentiation (USD) mechanism and is explained in detail in Section 4.1. This scheme has been proposed by Wang [13] in the working Internet draft "*User-Share Differentiation* (*USD*) *Scalable bandwidth allocation for differentiated services*". When requests for connections are accepted by the NAP, the NAP first allocates a negotiated minimum bandwidth. Additionally, any remaining link capacity to the network is distributed to all the connections in a proportion stipulated by the contract.

The major difficulty faced by the NAP in guaranteeing the negotiated QoS criteria is congestion. One way to regulate an overwhelming number of connections is to introduce a pricing scheme as a control mechanism. Establishing a pricing scheme that charges the users for demanding hard QoS guarantees from the network would reduce network resource allocation for non-critical applications, thereby facilitating provisioning of guaranteed service to users performing critical applications during congestion times.

Pricing mainly depends on fixed connection-based charges or subscriptions and usage charges to cover the resource operating cost (based on the duration, the volume and the distance of a connection). The ISP can also enforce extra charges for congestion periods and for the QoS. These factors are explained in more details in more specific works, for example in Walrand and Varaiya [12]. We now discuss some of the literature dealing with pricing issues.

Kelly [3] describes a charging and accounting mechanism based on an effective bandwidth concept. Edell et al. [2] present a system for billing users for their TCP traffic. MacKie-Mason and Varian [6] discuss issues on pricing in the Internet. Parris et al. [8] present a framework for pricing services and study the effect of pricing on user behavior and network performance. Cocchi et al. [1] study the role of pricing policies in multiple service class networks. Shenker et al. [9] state that the research agenda on pricing in computer networks should shift away from the optimality paradigm and focus on certain structural/architectural issues.

We note here that quantifying the amount of bandwidth required by a connection is a well-known problem in the Internet community. It is easier for a user to specify the observed call-blocking probability (which is application-independent) rather than translating the application requirements into an exact bandwidth requirement model (such as peak and average bandwidth rates, or other application-specific statistical models). Most of the existing proposed pricing schemes do not address this problem and assume a bandwidth requirement model specified by the user. One of the main aims of this paper is to determine the minimum bandwidth allocation that maximizes the revenue (as defined in our pricing mechanism) for the NAP subject to satisfy the constraint of the user's required call-blocking probability. The pricing scheme that we consider is discussed in detail in the next section, and takes into account not only the amount of bandwidth allocated but also the time of the day. These two goals are important in an environment where many NAPs share the market, and have to respond to users behavior and demand.

Queuing models have been extensively used in the design and control of communication networks (for example, see Kleinrock [4] and Walrand [11]). We will describe the system (NAP and users) using a queuing model with service (from NAP) according to a processor sharing mechanism, and where requests for connections represent customers arriving at the system, with no waiting line. As soon as requests are accepted by the system, the service begins. Processor Sharing models have been studied by de Veciana and Kesidis [10], Parekh and Gallager [7], etc.

In Section 2, we explain the pricing mechanism adopted in this paper. In Section 3, we describe a queuing model for the single-class connection requests arriving at the NAP. We solve an optimization problem to determine the minimum bandwidth allocation that maximizes the NAP's revenue subject to the blocking probability QoS constraint. Similar analysis is performed for the case when we consider multiple classes of users in Section 4, and a connection admission control (CAC) policy is discussed. We conclude by summarizing the results in Section 5.

2. Pricing scheme

2.1. *The purpose of a pricing scheme*

Certain goals are to be taken into consideration when designing a pricing scheme. Essentially, the NAP must recover the operating costs incurred for setting and maintaining a connection. These costs depend on the type of information being transmitted, and the duration of the connection. For example, a videoconference will use more resources (say bandwidth) than an email. Other than recovering costs, the NAP can indirectly perform congestion control by charging the users for time-of-the-day access, especially regulating the arrival rate of the connections during peak periods. Otherwise, an excessive congestion will result in the inability to provide service to critical applications for which users will be willing to pay a price.

Therefore, the components of an efficient pricing scheme must deal with the main features offered by applications in Next Generation Networks with multiple classes of traffic and QoS required by each class. The operating costs can be determined by the type of traffic transmitted (data, voice, video) and the QoS guaranteed for such transfer (limited delay, small cell loss probability, delay jitter, reserved bandwidth and blocking probability). As far as QoS is concerned, the concepts of reserved bandwidth and blocking probability are the key elements of the pricing scheme considered in the following sections.

In this paper, we develop the following pricing scheme. The users are charged a cost C_b per unit of minimum bandwidth the network allocates. The users are also charged a cost C_t per unit of time for the amount of time they spend accessing the network. Note that C_t could be different for different classes of traffic. Besides, C_b and C_t can possibly be varied according to the time of the day to serve as a congestion control mechanism.

2.2. Objective

The goal of this study is to determine the minimum bandwidth allocated for each class of traffic so that the revenue per unit of time earned by the NAP is maximized.

For a given class of traffic, let λ be the rate at which connection calls arrive at the server. And let b_m be the minimum bandwidth assigned to each connection. Let p_k be the long-run probability for the server to have k ongoing connections simultaneously of the given class. Let p_con be the long-run probability of not rejecting (i.e. providing connection to) an incoming user.

The long-run average revenue per unit time (AvR) for the given class of traffic is then expressed in terms of b_m as

$$\mathrm{AvR} = \sum_k C_\mathrm{t} k p_k + C_\mathrm{b} \lambda b_\mathrm{m} p_\mathrm{con}. \tag{1}$$

Note that p_k and p_con are functions of b_m.

The total long-run average revenue is obtained by summing over Eq. (1) for all classes of traffic. This will be the objective function of an optimization problem discussed in the following sections. A constraint in the optimization problem is to keep the rejection probability user requests below a certain negotiated value ε. For example, $\varepsilon = 0.01$ implies that users of a given class see no more than one in a 100 of their requests on an average rejected by the network at different times.

In this paper, we assume that the NAP enforces a given set of charges (interchangeably called costs) as suggested by the pricing model studied above. We will not discuss how to set those charges in this paper. We will rather be interested in setting an optimal resource allocation policy (namely, the minimum bandwidth) to maximize the NAP's revenue while satisfying the users demands for QoS.

3. Single-class calls

In order to elucidate the analysis, we first present a simple model where we assume that there is only one class of traffic. Later we extend the results to a more general N-class setting.

3.1. Model

Consider users who connect to an NAP to access the network. This server (NAP) has a total link capacity (or total bandwidth of the link between the NAP and the Internet) B (in megabits per second) available for all the users (see Fig. 2). User calls or requests arrive at the server according to a Poisson Process with parameter λ (represented as PP(λ)). Therefore, on an average, there are λ incoming requests at the server per unit time.

In this section, we assume that all requests are considered identical, hence we refer to them as requests belonging to a "single class". Each single-class call results in a transfer of a random amount of information whose size is assumed to be exponentially distributed with mean $1/\alpha$. All the calls that are on, simultaneously share the bandwidth equally. Therefore if there are k simultaneous connections, each connection receives a bandwidth of B/k (this is the rate at which information is transferred to the user). This is frequently denoted as Processor Sharing (see [7,10]). Hence when there are k users simultaneously accessing the NAP, the holding time for each connection is exponentially distributed with mean $(1/\alpha)/(1/\beta)$ $(=k/B\alpha)$. We denote

$$\mu_k = \frac{B\alpha}{k}. \tag{2}$$

According to the contract stipulations, each accepted request is guaranteed a minimum bandwidth b_{m} for the information flow. Note that a maximum of $S = \lfloor B/b_{\mathrm{m}} \rfloor$ connections can be handled simultaneously. Therefore, if there are S requests being processed, there is no more available capacity to accept another request. Hence any incoming request is rejected.

Based on the parameters λ and b_{m} and the costs C_{t} and C_{b} defined in Section 2.1, the long-run average revenue per unit time for the NAP is indeed Eq. (1) itself since we are considering a single class of calls. The objective is to calculate an optimal b_{m} that maximizes the revenue (AvR) in Eq. (1), subject to the blocking probability QoS constraint to keep the blocking probability lower than a certain level. We solve an optimization problem to calculate the optimal bandwidth in Section 3.4. We begin by calculating p_k defined in Eq. (1).

3.2. Analysis

The above model can be thought of as a multiserver queuing system with state-dependent service and no waiting where customers (or requests) arrive into the system according to PP(λ), a Poisson process with rate λ. There can be a maximum of S customers in the queuing system simultaneously. An arriving customer who finds S other customers in the system leaves immediately. Service begins as soon as a customer is accepted into the system.

Let $X(t)$, the state of the system, be the number of ongoing requests at time t. Since the maximum number of connections is $\lfloor B/b_{\mathrm{m}} \rfloor$, we have $0 \leq X(t) \leq S$ for all t. Each customer departs after an $\exp(\mu_k)$ amount of time when there are k customers in the system, where μ_k is defined in (2). Thus the time for the first of the k customers to depart the system is distributed as $\exp(k\mu_k)$ (the minimum of k exponential random variables, see [5]), which can be written as $\exp(k\mu_k) = \exp(B\alpha)$. Henceforth μ will denote $B\alpha$.

Therefore $\{X(t), t \geq 0\}$ is a Continuous Time Markov Chain (CTMC) with state space: $\{0, 1, \dots, S\}$, infinitesimal generator matrix $Q = [q_{ij}]$ such that

$$\begin{aligned} &q_{i,j} = \lambda \quad \text{if } j = i+1, \qquad q_{i,j} = B \cdot \alpha \quad \text{if } j = i-1, \\ &q_{i,j} = 0 \quad \text{otherwise}, \qquad q_{i,i} = -\sum_{j \neq i} q_{i,j}. \end{aligned} \tag{3}$$

The transition diagram is represented in Fig. 3.

Note that the $\{X(t), t \geq 0\}$ process is analogous to the queue length process in an M/M/1/S queuing system. From [5], solving the balance equations related to the above rate diagram provides us with the long-run probability for the system to be in state S and the long-run expected queue length, as expressed in the following equations:

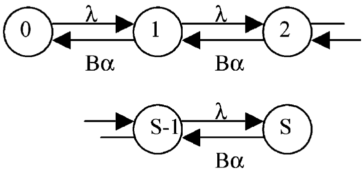

Fig. 3. Transition diagram of the single-class process.

$$p_S = \frac{(1-\rho)\rho^S}{1-\rho^{S+1}}, \tag{4}$$

$$L(S) = \sum_{k=0}^{S} k p_k = \frac{\rho}{1-\rho}\left[\frac{1-\rho^S}{1-\rho^{S+1}} - S\frac{(1-\rho)\rho^S}{1-\rho^{S+1}}\right], \tag{5}$$

where

$$\rho = \frac{\lambda}{\mu}, \tag{6}$$

and $L(S)$ is the long-run queue length in terms of S.

We can then write p_{con} as

$$p_{\text{con}} = 1 - P\{\text{being in state } S\} = 1 - p_S.$$

Substituting $L(S)$ for $\sum kp_k$ and p_{con} in Eq. (1), we get

$$\text{AvR} = C_{\text{t}} L(S) + C_{\text{b}} \lambda \frac{B}{S}(1 - p_S). \tag{7}$$

An incoming call will see S customers in the system with probability p_S (due to PASTA [5]) and hence be rejected. The probability at which customers are rejected is p_S. Then the constraint to satisfy is

$$p_S \le \varepsilon, \tag{8}$$

where ε (in terms of number of calls blocked per day) is the negotiated QoS call-blocking probability.

3.3. Optimization problem

The goal is to maximize AvR as expressed in Eq. (7) with respect to S, the decision variable, while satisfying the constraint (8). Therefore we can formulate the above optimization problem:

$$\max\left\{\text{AvR} = C_{\text{t}} L(S) + C_{\text{b}} \lambda \frac{B}{S}(1 - p_S)\right\}, \qquad \text{subject to } p_S \le \varepsilon, \quad S \ge 0, \text{ integer}. \tag{9}$$

Once an optimal value S^* is obtained, the equivalent optimal minimum bandwidth allocation b_{m}^* is directly derived from $b_{\text{m}}^* = B/S^*$.

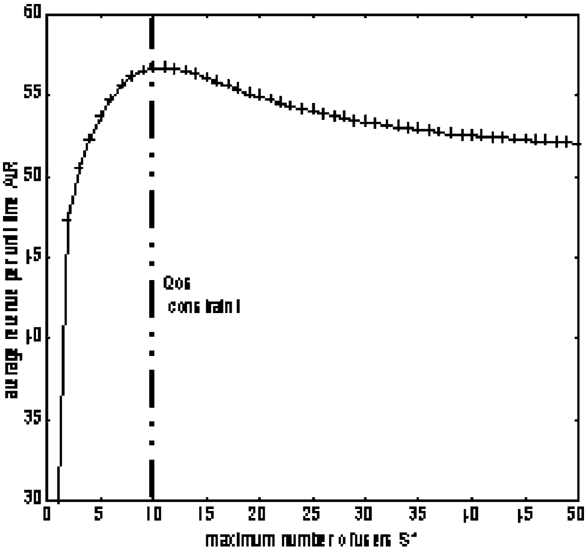

Fig. 4. Long-run average remove per unit time versus maximum number of users.

3.4. Results

Since no closed form algebraic expression of the optimal S^* exists, we plot the objective function and the constraints in (9) and solve it numerically. We use the following numerical values in Fig. 4: $\lambda = 2$ customers/second, $\mu = 3$ customers/s, $B = 10$ mbps, $C_t = 25$ cents/s, $C_b = 5$ cents/mbps, $\varepsilon = 10^{-2}$.

Fig. 4 characterizes the revenue function (AvR) for the above set of input values. At first, as S tends to zero, the part of the revenue attributed to the bandwidth gets high (due to the B/S term) but is offset by the high blocking probability yielding $1 - p_S \approx 0$. As S increases, the time spent on the network takes the greatest share and becomes more significant: the requests take more time to be processed because of a smaller bandwidth. For large values of S, the revenue approaches a constant asymptotically. This constant corresponds to C_t times the long-run average queue length of a queuing system with an infinite number of servers: $C_t\lambda/(\mu - \lambda)$ (=50 in the computational example).

The vertical line represents the value of S satisfying $p_S = \varepsilon$. Therefore, the feasible region (satisfying (9)) is only to the right of the vertical line. The maximum revenue is the largest AvR in this feasible region. In this case, the optimal value turns out to be AvR $= 56.69$ cents/s, for $S^* = 11$ users, and $b_m^* = 0.909$ mbps.

Note that the numbers used in this example may not seem realistic when compared to real scale problems. We used them purely for illustration purposes.

3.5. Equilibrium

Notice that the arrival rate λ into an NAP is dependent on the user behavior. We conjecture that λ is very sensitive to changes in the minimum bandwidth provided as users switch NAPs when the QoS (minimum bandwidth) is not satisfactory.

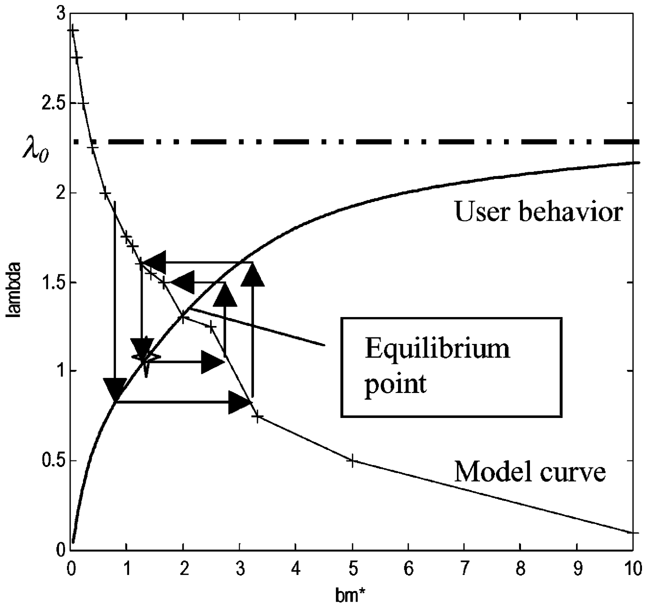

Fig. 5. Illustration of an equilibrium point between the arrival rate and the offered bandwidth.

Therefore, it is of great interest to study the interactions between the parameters of our model, such as the arrival rate, and how they may drive it to an eventual equilibrium. In the above model, those parameters were chosen a priori and others constitute an output: b_{m}^*.

The purpose of this section is to describe how a further interaction between these parameters can affect the equilibrium of our model. Specifically, we studied the effect of modifying λ on b_{m}. On one hand, the model suggests that the lower the arrival rate λ, the higher the minimum bandwidth b_{m}. This is intuitive: high arrival rates yield a high blocking probability, unless the number of admitted requests is high, or equivalently the minimum bandwidth is low. On the contrary, low arrival rates lead to low revenue unless the NAP provides a high minimum bandwidth. In fact, this is true of any QoS parameter as the higher the arrival rate the worse the QoS provisioned. The curve describing the model (Fig. 5) is obtained by determining the optimal values b_{m}^* for different values of λ.

On the other hand, users have an option to choose their NAP. From a user's perspective, the higher the minimum bandwidth, the more attractive the NAP. So a model for users behavior is the higher the minimum bandwidth, the higher the arrival rate. Thus the arrival rate is an increasing function of the minimum bandwidth. However, we conjecture that a typical user behavior suggests that the function be concave with declining margins, and asymptotically reaching a limit λ_0 (the users' arrival rate cannot keep on increasing indefinitely). We assume that such dependence can be modeled using the following function:

$$\lambda = \lambda_0(1 - \mathrm{e}^{-b_{\mathrm{m}}^*}). \qquad (10)$$

We briefly discuss two different cases of convergence.

Fig. 5 represents the above scenario. The graph uses the same values of B, μ, C_{t}, C_{b}, and ε, as used in Section 3.4.

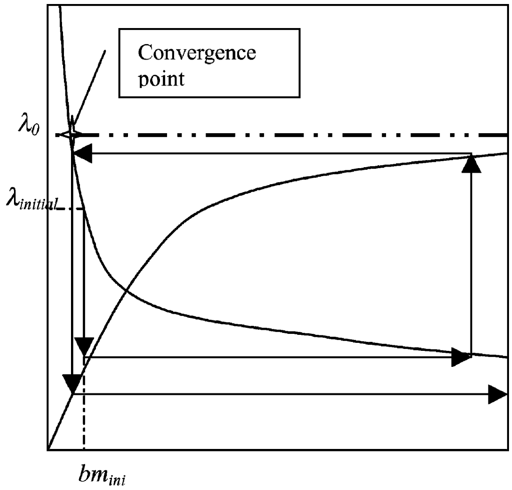

Fig. 6. Extreme case for illustration of an unstable convergence point.

For example, by first choosing an arrival rate of 1.9, the optimal minimum bandwidth turns out to be about 0.75 (megabits per second). This yields a new user behavior corresponding to an arrival rate of 0.8, that in turn leads to $b_{\mathrm{m}}^* = 3.25$, and so on. An equilibrium point is the intersection of the model curve and the user behavior representation.

In theory, a second case can occur when this iterative process converges to another point, as represented in Fig. 6 where the model curve does not decrease rapidly towards zero. In this case, the convergence point is the intersection between the model curve and the line $\lambda = \lambda_0$. Slightly moving the minimum bandwidth from the convergence point forces a great disturbance in the system, since it requires the NAP to set a large value of b_{m} in order to bring the arrival rate close to λ_0. Obviously, this convergence point is not stable, unlike the first one (see Fig. 5).

In most practical cases, the model curve will decrease towards zero, because a large minimum bandwidth can never welcome enough users due to the system's limited capacity. However, it is useful to have a plot of the user behavior, so that the provider makes a judicious choice of the initial minimum bandwidth.

We now present a mathematical analysis to determine if the system would be stable and converge or not. Consider variables x and y (surrogates for minimum bandwidth and arrival rates). Let $f(\cdot)$ be a decreasing function, a surrogate for the model curve, that maps y to x values. Similarly, let $g(\cdot)$ be an increasing function, a surrogate for user behavior, that maps x to y values. Therefore we have $x = f(y)$ and $y = g(x)$. Given an initial value y_0, by iterating over $i = 0, 1, 2, \dots, x_i = f(y_i)$ and $y_{i+1} = g(x_i)$, we get the series of coordinate points (x_i, y_i). Next we derive the conditions under which a stable equilibrium point (x_n, y_n) exists as $n \to \infty$. It follows directly from the definition of the process that

$$|y_{i+1} - y_i| = |g(x_{i+1}) - g(x_i)| = |gf^{-1}(y_i) - gf^{-1}(y_{i-1})|.$$

By the Lagrange theorem the right-hand side of this equality is equal to $|(gf^{-1})'(\xi)||y_i - y_{i-1}|$, where $\xi \in (y_{i-1}, y_i)$, or (y_i, y_{i-1}) if $y_i \leq y_{i-1}$.

So the convergence exists provided that $|(gf^{-1})'(y)| < 1$ on all such segments. By continuity and concavity of both functions, one can find some neighborhood U_0 of the point $y_0 = f(x_0)$ where this condition holds for every interior point. Hence if the condition $|(gf^{-1})'(y)| < 1$ is satisfied for an arbitrary point in U_0, the iterative process is guaranteed to have convergence within the domain U_0. It is easy to show that in this case the limit is unique and is equal to $\lim_{n\to\infty} y_n = y_0 = f^{-1}(x_0) = g(x_0)$. Indeed, by continuity of function g, we have $g(x_0) = g(\lim_{n\to\infty} x_n) = \lim_{n\to\infty} g(x_n) = \lim_{n\to\infty} y_n = y_0$. From the picture it is clear that in case of divergence we will eventually reach the threshold value λ_0, corresponding to the unstable equilibrium mentioned above, which makes our convergence analysis complete. We do not present this analysis for the multi-class case with the understanding that the reader can extrapolate the results in multi-dimensions by letting x and y be vectors.

3.6. *General distribution for file sizes*

In Section 3.1, we considered single calls that result in a transfer of a random amount of information whose size is assumed to be exponentially distributed with mean $1/\alpha$. However, the exponential assumption can be relaxed. In fact only the mean file size is required for the analysis, as we will see in this section. Therefore, distributions with infinite variance (such as Pareto) can also be used. The reason we presented the exponential distribution analysis is that in the multi-class case, the problem is tractable only under exponential distributions for which we extend the single-class analysis.

Assume that the file size is distributed generally (with CDF $G(\,)$) with mean $1/\alpha$. Using the result for M/G/C/C queues with state-dependent service in Smith and Jain [14], we have the probability that there are i $(0 \le i \le C)$ customers in the system, $p(i)$, as

$$p(i) = \frac{(\lambda/\mu)^i}{i!r(i)r(i-1)\cdots r(1)} p(0),$$

where $r(i)$ is the ratio of the service rate of a single customer when there are i customers to the service rate when there is one customer. The expression $p(0)$ can be computed by solving $\sum_{i=0}^{C} p(i) = 1$.

Our processor sharing queue can also be modeled as an M/G/S/S queue with state-dependent service such that $r(j) = 1/j$ for $j = 1, 2, \ldots, S$. Therefore, the long-run probability for the system to be in state S and the long-run expected queue length, are the exact same expression in Eqs. (4) and (5). This leads us to believe that the exponential distribution results derived for the single-class case hold good for any general distribution with mean $1/\alpha$ thereby making the analysis more powerful and generalized. Unfortunately this analysis cannot be easily extended to the multi-class case. At this time, we only conjecture that for multi-class traffic, the results for general file sizes would be identical to those assuming exponential file sizes. This will be addressed in future work.

4. Multi-class calls

The single-class model in Section 3 can be extended to multi-class calls that take into account the multi-class DS architecture proposed for Next Generation Networks. We are considering the case where different levels of QoS are required, which could happen in two different ways: either the type of information conveyed requires a higher amount of bandwidth to meet the regular needs of QoS, or the users themselves ask for a “privileged” (higher priority) service that they are ready to pay for to fulfill their needs in case of delay sensitive or urgent information transfers.

4.1. Model

Consider that there are N different classes of calls. Class n connections ($n = 1, \dots, N$) arrive according to a Poisson process PP(λ_n) with a mean of λ_n calls per unit time. The size of information transfers is exponentially distributed with mean $1/\lambda_n$ for class n connections.

The bandwidth allocation to an arriving call is based on the following. Class n connections are allocated a minimum bandwidth $b_{\mathrm{m}_n}, n = 1, \dots, N$. Without loss of generality, we assume $b_{\mathrm{m}_1} < b_{\mathrm{m}_2} < \cdots < b_{\mathrm{m}_N}$. When a call of type m requests a connection of type m, given that there are already k_n calls of type n ($n = 1, \dots, N$) in progress, the service provider accepts the request if there is enough bandwidth for the new call. This is equivalent to satisfying the condition

$$\sum_{n=1}^{N} k_n b_{\mathrm{m}_n} + b_{\mathrm{m}_m} \le B. \tag{11}$$

Let b_n be the instantaneous bandwidth assigned to each connection of type n. Note that the instantaneous bandwidth changes whenever a new call arrives or an existing call terminates, but it always is greater than the minimum allocated bandwidth b_{m_n}. Keeping class-1 as a reference, let β_n be the ratio $b_{\mathrm{m}_n}:b_{\mathrm{m}_1}$. Each class-$n$ call is allocated the minimum bandwidth b_{m_n}, and then the rest of the total available bandwidth is shared among all the ongoing calls in the same proportion as the minimum bandwidths, i.e. β_n. This is equivalent to having

$$\frac{b_n}{b_1} = \frac{b_{\mathrm{m}_n}}{b_{\mathrm{m}_1}} = \beta_n \quad \forall n = 1, \dots, N \tag{12}$$

at every instant.

This is also referred to as the USD as proposed in [13]. In order to obtain insights into the model and also to slowly build the model from simple to complex we first start with a two-class model ($N = 2$), then later generalize for any N.

4.2. Two-class model: analysis

We consider two classes of traffic (i.e. $N = 2$) to explain the analysis.

Let λ_1 and λ_2 be the mean arrival rates of the connections of types 1 and 2, respectively. Their information transfer size is distributed exponentially with mean $1/\alpha_1$ and $1/\alpha_2$, respectively. Let k_1 and k_2 be the number of ongoing connections. Also, b_{m_1} and b_{m_2} are the minimum bandwidths allocated to each class. We denote $\beta_2 = b_{\mathrm{m}_2}/b_{\mathrm{m}_1}$as β.

When a new call arrives or an existing call departs, resulting in k_1 and k_2 calls of class-1 and class-2, respectively, the instantaneous bandwidths b_1 and b_2 are chosen such that

$$k_1 b_1 + k_2 b_2 = B \quad \text{and} \quad \frac{b_2}{b_1} = \beta. \tag{13}$$

Therefore,

$$b_n = B \frac{b_{\mathrm{m}_n}}{k_1 b_{\mathrm{m}_1} + k_2 b_{\mathrm{m}_2}} \quad \text{for } n = 1, 2. \tag{14}$$

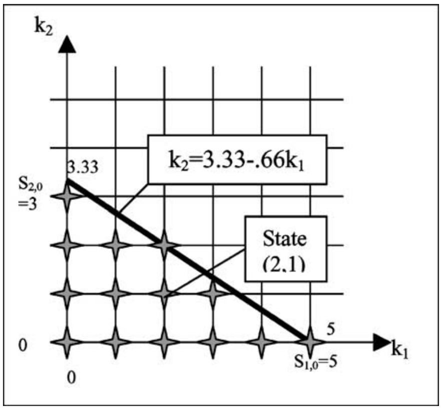

Fig. 7. Graphical representation of the possible states of the two-class calls model.

Table 1
Example for computing S_{2,k_1}

k_1	0	1	2	3
S_{2,k_1}	2	1	1	0

Let μ_n, k_1, k_2 be the instantaneous rate at which a type n call is completed when there are k_1 (respectively k_2) ongoing calls of type 1 (respectively type 2) in the network. Therefore, similar to Eq. (2), we obtain

$$\mu_{n,k_1,k_2} = \alpha_n b_n = \frac{\alpha_n b_{\mathrm{m}_n} B}{k_1 b_{\mathrm{m}_1} + k_2 b_{\mathrm{m}_2}}. \tag{15}$$

Let $X_1(t)$ (respectively $X_2(t)$) be the number of ongoing connections of type 1 (respectively of type 2) at time t. We model the stochastic process $\{(X_1(t), X_2(t)), t \leq 0\}$ as a CTMC. The number of possible states depends on the values of B, b_{m_1}, and b_{m_2}. For a given set B, b_{m_1}, and b_{m_2}, the maximum number of users of type 2 (1) given that there are k_1 type 1 (2) users is

$$S_{2,k_1} = \left\lfloor \frac{B - k_1 b_{\mathrm{m}_1}}{b_{\mathrm{m}_2}} \right\rfloor, \tag{16}$$

$$S_{1,k_2} = \left\lfloor \frac{B - k_2 b_{\mathrm{m}_2}}{b_{\mathrm{m}_1}} \right\rfloor. \tag{17}$$

For $B = 10$, $b_{\mathrm{m}_1} = 2$, $b_{\mathrm{m}_2} = 3$, we have $S_{1,k_2} = \lfloor 5 - 1.5k_2 \rfloor$ and $S_{2,k_1} = \lfloor 3.33 - 0.66k_1 \rfloor$. The states of $\{(X_1(t), X_2(t)), t \leq 0\}$ are represented graphically by the set of points in Fig. 7. To draw the transition diagram (Fig. 9), the states are represented as in Fig. 8.

Consider the simple example represented in Tables 1 and 2 and Fig. 9 based on the following numerical values: $B = 5$, $b_{\mathrm{m}_1} = 1.5$, $b_{\mathrm{m}_2} = 2$.

Note that in Fig. 9, the states with stripes are the ones where at least one class of calls cannot be accepted, and hence blocking occurs. The vertical stripes show the states where class-2 calls are blocked (extreme right of the rate diagram). Horizontal stripes show class-1 blocking (extreme bottom states). We call them class-n "blocking states" ($n = 1, 2$). We see that in some states (state (0,2), (2,1), and (3,0))

Table 2
Example for computing S_{1,k_2}

k_2	0	1	2
S_{1,k_2}	3	2	0

both classes of calls are blocked. Note that class-1 blocking states are a subset of class-2 blocking states: it is obvious that every class-2 blocking state is also a class-1 blocking state since $b_{\mathrm{m}_2} \geq b_{\mathrm{m}_1}$.

As explained in Section 3.2, a state change will occur when either a connection (of type n) arrives into the system (at rate λ_n) or when such a connection terminates (at rate $k_n\mu_{n,k_1,k_2}$). The transition rates are summarized generically in Fig. 10.

Let p_{ij} be the long-run probability of being in state (i, j) of the CTMC. Let Q be the infinitesimal generator matrix with transition rates from state (i, j) to state (k, l) represented in Fig. 10.

From [5], $[p_{ij}]$-vector is the unique solution to the balance equations:

$$pQ = 0, \tag{18}$$

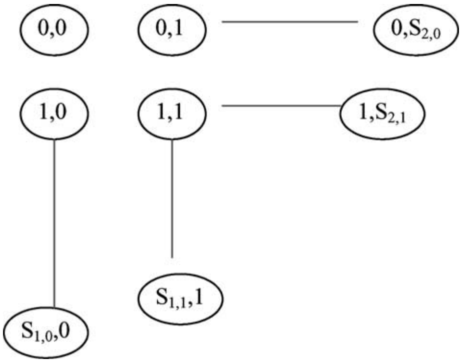

Fig. 8. The possible states for the two-class calls model.

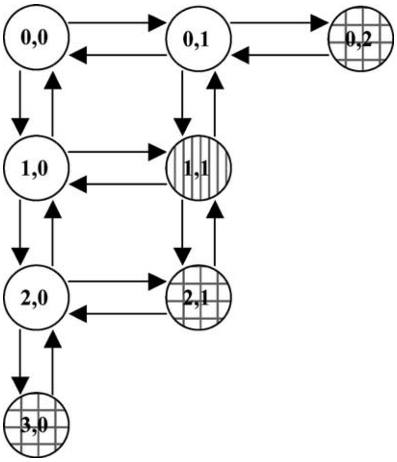

Fig. 9. Transition diagram of the two-class CTMC example.

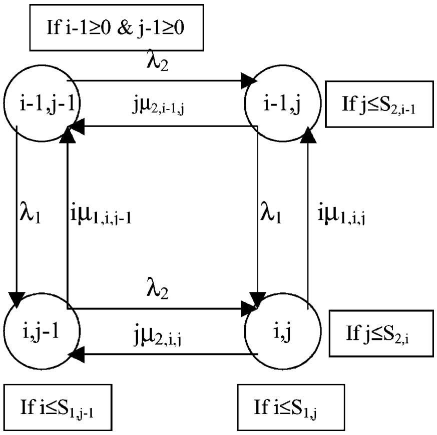

Fig. 10. Summary of the transition rates for the two-class calls model.

and

$$\sum pij = 1. \tag{19}$$

Since no closed form algebraic expression exists for p_{ij}, we solve Eqs. (18) and (19) numerically given the values of B, b_{m_1} and b_{m_2}. For the optimization problem stated in Eqs. (20)–(25), the decision variable is $S_{1,0}$ (which is the maximum number of users of type 1 given that there are no type-2 users $= B/b_{m_1}$). We assume that β is given and is part of a strategic decision made by the ISP.

Once the elements of the $[p_{ij}]$-vector obtained, it is possible to compute AvR using the following objective function:

$$\text{AvR} = \sum_{(i,j)} p_{ij}(iC_{t_1} + jC_{t_2}) + C_{\text{b}}(\lambda_1 p_{\text{con}_1} b_{\text{m}_1} + \lambda_2 p_{\text{con}_2} b_{\text{m}_2})$$

or

$$\text{AvR} = \sum_{(i,j)} p_{ij}(iC_{\text{t}_1} + jC_{\text{t}_2}) + C_{\text{b}} b_{\text{m}_1}(\lambda_1 p_{\text{con}_1} + \lambda_2 p_{\text{con}_2}\beta). \tag{20}$$

We are seeking to maximize the above objective function with respect to $S_{1,0}$ (Q and $p_{i,j}$ depend on $S_{1,0}$). We also aim at keeping the call-blocking probability lower than ε_n for class-n calls. Therefore, the optimization problem is subject to the following constraints:

$$\sum_{j=0}^{S_{2,0}} p_{S_{1,j},j} \le \varepsilon_1, \tag{21}$$

$$\sum_{i=0}^{S_{1,0}} p_{i,S_{2,i}} \le \varepsilon_2, \tag{22}$$

$$p_{\text{con}_2} = 1 - \sum_{i=0}^{S_{1,0}} p_{i,S_{2,i}}, \tag{23}$$

$$p_{\mathrm{con}_1} = 1 - \sum_{j=0}^{S_{2,0}} p_{S_{1,j},j}, \tag{24}$$

$$S_{1,0} = \left\lfloor \frac{B}{b_{\mathrm{m}_1}} \right\rfloor, \qquad b_{\mathrm{m}_2} = \beta b_{\mathrm{m}_1},\ S_{1,0} > 0,\ \text{integer}. \tag{25}$$

The expression in (20) sums up the long-run probabilities of being in the class-1 blocking states. Hence the long-run probability of providing connection to an incoming type-1 user is computed as derived in Eq. (23). Similarly, the expressions in (21) and (23) relate to the class-2 blocking states and the long-run probability of providing connection to an incoming type-2 user. We solve the above optimization problem with respect to $S_{1,0}$. Once $S_{1,0}$ is obtained, the corresponding values of b_{m_1} and b_{m_2} are derived using the equations in (24).

4.3. Results for $N = 2$ classes

For a given value of β, we seek the maximum of AvR versus the values of $S_{1,0}$. The choice of β is dictated by the application, the market characteristics and user behavior. For $\beta = 1$, it is equivalent to the single-class problem in Section 3. Although we assume that the choice of β is defined by exterior conditions (usually a strategic value decided by the ISP planners), we briefly discuss the effect of the choice of β on the revenue. We illustrate the cases of $\beta = 2$ and 3 in Figs. 11 and 12, respectively. The figures also show the blocking probability constraints (20) and (21). For Figs. 11 and 12, the numerical values used are: $B = 0.5$ mbps, $\beta = 2$ and 3, $\lambda_1 = \lambda_2 = 0.25$ customers/s, $\alpha_1 = \alpha_2 = 10/3\,\mathrm{mbps}^{-1}$, $C_{\mathrm{t1}} = C_{\mathrm{t2}} = 12$ cents/s, $C_{\mathrm{b}} = 10$ cents/mbps, $\varepsilon_1 = \varepsilon_2 = 10^{-2}$.

The vertical line on the left (respectively on the right) in Figs. 11 and 12 represent the value of $S_{1,0}$, where the blocking probability for connections of type 1 (respectively type 2) is equal to ε_1 (respectively

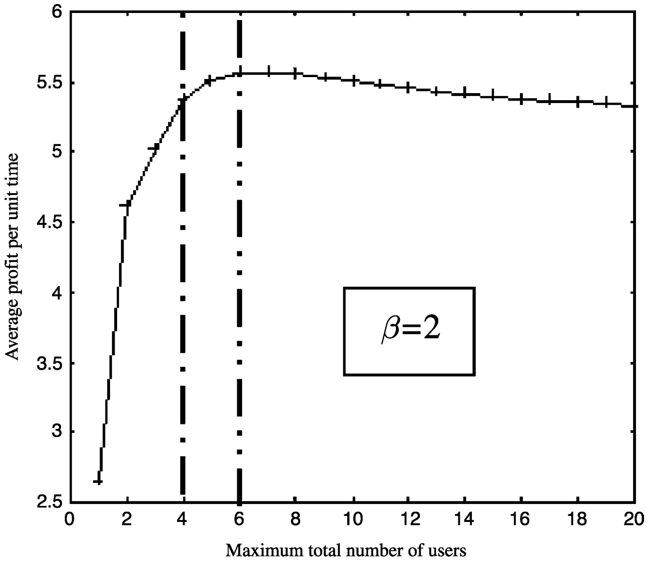

Fig. 11. Long-run average revenue per unit time versus $S_{1,0}$ (β=2).

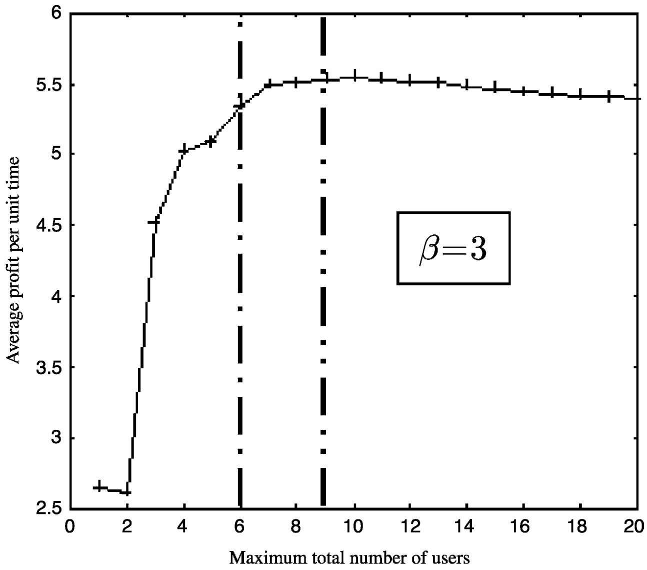

Fig. 12. Long-run average revenue per unit time versus $S_{1,0}$ ($\beta = 3$).

ε_2). The feasible region is to the right of the right-hand side line. Note that for usual cases, for the same values of ε_1 and ε_2, we will always have the line corresponding to the type 2 on the right of that of type 1. And that is because the class-1 blocking states are a subset of class-2 blocking states, as noticed in Section 4.2, hence the sum in inequality (20) is larger than that in inequality (21).

Therefore, the optimum point is the largest AvR in the feasible region. The optimal value for $\beta = 2$ is AvR $= 5.57$ cents/s when $S_{1,0} = 7$ users; for $\beta = 3$, the optimum is AvR $= 5.55$ cents/s at $S_{1,0} = 10$

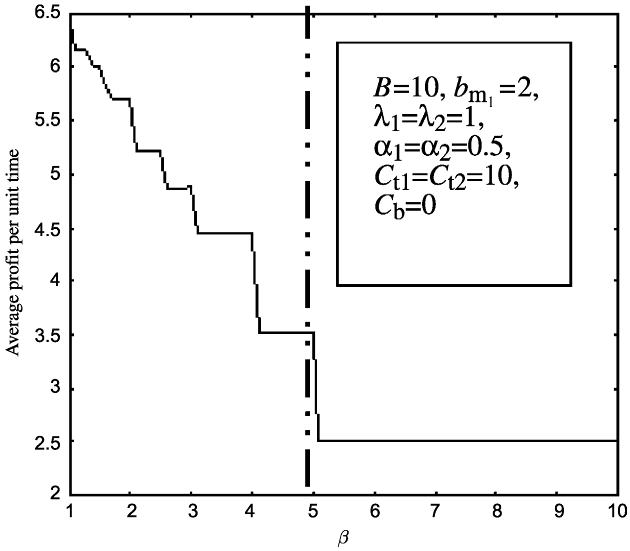

Fig. 13. Long-run average revenue versus β only with the cost C_t (C_b is set to zero).

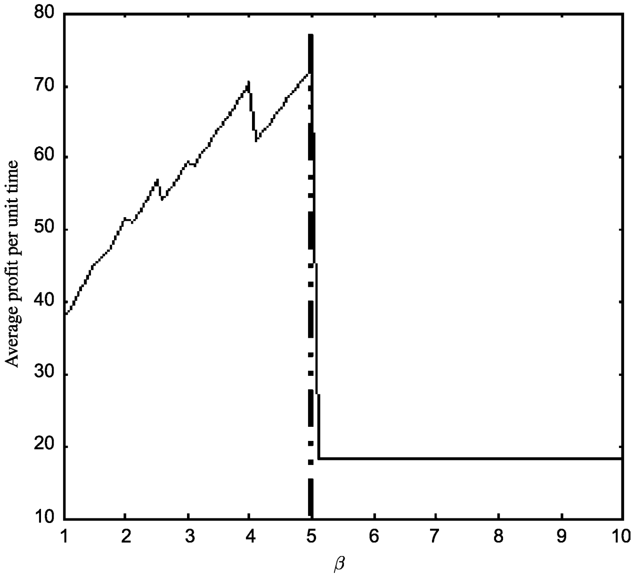

Fig. 14. AvR versus β with both C_t and C_b are considered.

users. Since the problem consists of a single variable, the solution methodology to the optimization problem is just complete enumeration.

In Fig. 13, we plot AvR versus β, with B and b_{m_1} remaining constant (and we only represent the type 2, i.e. the more constraining user QoS constraint).

We see from Eq. (20) that the term of the revenue due to C_b increases linearly with β. And we see in Fig. 13 that the variation with β of the term comprising C_t is decreasingly piecewise constant.

The revenue consists in two terms of opposite trends, and the combination of both terms could yield an undesirable result such as a strongly decreasing function. However, for an appropriate combination of the two charges C_t and C_b, the curve adopts the behavior shown in Fig. 14. The most favorable cost combination and the choice of the best value of β can be determined through the study of price optimality for a given set of system parameters, and is beyond the scope of this paper.

Figs. 3, 11–14 are highly dependent on the numerical values of the parameters (λ, α, C_t, C_b) involved in the computations. For every change in these parameters, the shapes of the curves obtained can be different. The values of these parameters may depend on the applications and the user behavior: users with the same behavior (λ and α), but requesting different classes of transfer, or users with totally different behaviors (large mean size transfer, but low arrival rate for example) and willing to pay much more for their special requests ($C_{t_2} \gg C_{t_1}$). Hence, in Tables 3 and 4, we summarize a qualitative description of how the changes in the above parameters affect the optimal revenue.

4.4. *N*-class model: analysis and results

The study of the two-class calls is very useful to help set the frame of the problem, since it is easier to visualize two-dimensional CTMCs than N-dimensional ($N > 2$) CTMCs. It also helps understand the general behavior of the revenue function, the QoS constraint and the optimum bandwidth.

The two-class model is a very simplistic case. Our model can then be generalized to N classes of calls with some computational effort. Consider N applications (for example FTP, SMTP, Telnet, etc.) that require N different amounts of minimum bandwidths and different QoS constraints. Let $X_n(t)$ be the number of ongoing connections of type n (for $n = 1, 2, \dots, N$) at time t. We model the stochastic process $\{(X_1(t), X_2(t), \dots, X_N(t)), t \geq 0\}$ as a CTMC. Since the arrival or departure of a class n call changes the state of the CTMC as an increase or decrease of "one" in the nth dimension, the CTMC is an N-dimensional birth and death process. To compute the stationary probabilities $p = [p_{i_1,i_2,\dots,i_N}]$, we need to solve the balance equation (18). This is computationally intensive. Recent papers by Servi et al. [15,16] contain an extremely fast algorithm to solve the balance equations for N-dimensional birth and death processes. However, that algorithm requires a special structure not found in our birth and death process. Therefore, we need to make some clever modification to our birth and death process. Before explaining the novel technique and details of the algorithm, the parameters used are first explained.

The instantaneous bandwidth and transmission rate for class n traffic when there are k_i class i traffic for $i = 1, 2, \dots, N$ are

$$b_n = B\frac{b_{\mathrm{m}_n}}{k_1 b_{\mathrm{m}_1} + k_2 b_{\mathrm{m}_2} + \cdots + k_N b_{\mathrm{m}_N}} \quad \text{for } n = 1, 2, \dots, N,$$

$$\mu_{n,k_1,k_2,\dots,k_N} = \alpha_n b_n = \frac{\alpha_n b_{\mathrm{m}_n} B}{k_1 b_{\mathrm{m}_1} + k_2 b_{\mathrm{m}_2} + \cdots + k_N b_{\mathrm{m}_N}}.$$

The objective function is

$$\mathrm{AvR} = \sum_{(i_1,i_2,\dots,i_N)} p_{i_1,i_2,\dots,i_N}(i_1 C_{\mathrm{t}_1} + i_2 C_{\mathrm{t}_2} + \cdots + i_N C_{\mathrm{t}_N}) + C_{\mathrm{b}} b_{\mathrm{m}_1}(\lambda_1 p_{\mathrm{con}_1} + \lambda_2 p_{\mathrm{con}_2}\beta_2 + \cdots + \lambda_N p_{\mathrm{con}_N}\beta_N).$$

In order to explain the algorithm to solve the balance equation, we first start with the rate matrix Q. In case of a birth–death process, Q can be represented in the following recursively tri-diagonal

Table 3
Effect of parameter changes on optimal revenue

Effect of/on	Maximum revenue	Blocking probability	Minimum bandwidth, B_{m}
Inc. α	Lower	Lower	Higher
Inc. λ	Higher	Higher	Lower
$C_{\mathrm{b}} \ll C_{\mathrm{t}}$		Same	Lower

Table 4
Steady-state probabilities p_{ij} compared to solutions $\tilde{p}_{ij}$ of the perturbed system for $S_{1,0} = 2$

State	Exact probability, p_{ij}	Approximated solution, $\tilde{p}_{ij}$
(0,0)	0.755946	0.755942
(1,0)	0.113505	0.113508
(2,0)	0.017043	0.017049
(0,1)	0.113505	0.113502

form:

$$\begin{bmatrix} v_0^0 & v_0^+ & 0 & \cdots & 0 \\ v_1^- & v_1^0 & v_0^+ & \cdots & 0 \\ & \ddots & \ddots & \ddots & \\ 0 & \cdots & v_{m-1}^- & v_{m-1}^0 & v_{m-1}^+ \\ 0 & \cdots & 0 & v_m^- & v_m^0 \end{bmatrix},$$

where $v_{\underline{j}}^{\underline{s}}$ are the $n \times n$ block tri-diagonal infinitesimal generator matrices whose (i, k)th component is the probability flow from $(j_1, \dots, j_N, i)$ to $(j_1 + s_1, \dots, j_N + s_N, k)$. For skip-free birth–death processes with such matrices, a method was found (Servi et al. [15,16]), which allows to reduce the number of computations from $\mathrm{O}(n^6)$ to $\mathrm{O}(n^3)$. It essentially uses the recursively block tri-diagonal structure of the matrix and exploits recursive procedure to reduce the number of multiplications and hence memory usage.

The difficulty in using the algorithm in Servi et al. [16] is that it requires the N-dimensional grid for the birth and death process to be cuboidal. In this paper, since all states $(k_1, k_2, \dots, k_N)$ must satisfy the constraint $\sum_{n=1}^{N} k_n b_{\mathrm{m}_n} \le B$ the resulting N-dimensional grid is not cuboidal. In order to use the algorithm in Servi et al. [15,16], we convert our grid into a cuboidal one by inserting the appropriate states. Define state space S such that all states $(k_1, k_2, \dots, k_N)$ that satisfy the constraint $\sum_{n=1}^{N} k_n b_{\mathrm{m}_n} \le B$, belong to S. This matrix for Servi et al. [15,16] algorithm can be obtained from the original rate matrix by adding "fictitious" states into the lattice, i.e. replacing the state space by the larger set S_0

$$S \subset S_0 = \{(k_1, \dots, k_N) : 0 \le k_i \le S_{i,k_1,\dots,k_N}, i = 1, \dots, N\},$$

where S is the state space in the original setting. The dimension of so constructed matrix Q is $\prod_{i=1}^{N} (S_{i,k_1,k_2,\dots,k_N} + 1)$. Now that the fictitious states have been defined the next question is: what are the rates, i.e. the birth and death parameter? In fact, they would be zero and infinity in the appropriate places. But in order to make the matrix invertible, we assign birth and death parameter as follows (δ—sufficiently small):

1. $v_{\underline{j}}^{\underline{s}}(i, j) = \delta$ when $(\underline{j}, i) \in S$, $(\underline{j} + \underline{s}, k) \notin S$.
2. $v_{\underline{j}}^{\underline{s}}(i, j) = A$ when $(\underline{j}, i) \notin S$, $(\underline{j} + \underline{s}, k) \in S$.
3. $v_{\underline{j}}^{\underline{s}}(i, j) = A$ when $(\underline{j}, i) \notin S$, $(\underline{j} + \underline{s}, k) \notin S$.

This perturbation introduces a small flow rates δ into virtual states compensated by high outgoing rates A to ensure low long-run probability of being in fictitious states. We demonstrate this approach on the two-dimensional example considered in Section 4.2. Fig. 15 is a modification of Fig. 9 for perturbed situation.

Following Ross [18], for the fictitious states we always pick the outgoing rate A to be of the order $1/\delta$ to satisfy the time reversibility of the birth–death process, i.e. for the flows from state i to state j, we need to satisfy: $p_i q_{ij} = p_j q_{ji}$, wherein for the transition from "real" to "fictitious" state we have $q_{ij} = \delta$, $q_{ji} = A$. To preserve the order of magnitude, A should be picked as $1/\delta$ for the numerical model we consider.

It has to be noted that the discrepancy of solution produced by such perturbation behaved as $\mathrm{O}(\delta^2)$ in all numerical examples that we considered. Table 1 shows solutions to both regular and perturbed systems for the example considered above (see Fig. 11), where the exact solutions were obtained by

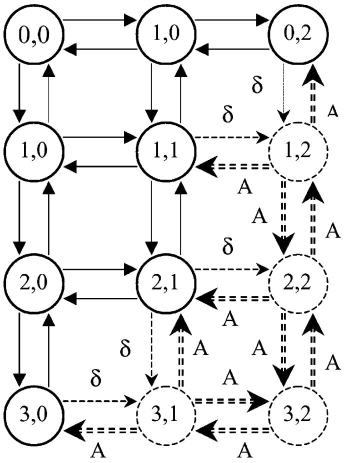

Fig. 15. State diagram for a two-class CTMC example with perturbed matrix, where δ is the perturbation parameter, $A = 1/\delta$.

solving the two-dimensional birth and death process in Section 4.2: $\delta = 0.25 \times 10^{-3}$, $B = 0.5$ mbps, $\lambda_1, \lambda_2 = 0.25$ customers/s, $\alpha_1, \alpha_2 = 10/3\,\text{mbps}^{-1}$, $\beta_2 = 2$.

In all considered examples discrepancy became even smaller with increase of $S_{\max}(1)$, so it was dominated by its values at the early stages of the algorithm, which leads us to believe that the estimate $|p_{ij} - \tilde{p}_{ij}| \le C\delta^2$ would have to be true for the whole convergence process. This agrees with the fact that partial pivoting elimination method used in our algorithm is stable to small perturbations of initial data, as shown in [17].

Numerical considerations showed, that deviation behaves similarly for large multi-class systems. However, one can expect this estimate to grow when δ is picked inadequately large, and at the same time reducing perturbation by several orders can lead to increase in computational complexity and the problem becomes less stable. Hence one should seek a compromise between these two conditions when fixing the order of perturbation.

We can now resort to a modification of the method discussed in Servi et al. [15,16] for solving the non-singular system. Although we follow the same idea as discussed in Servi et al. [16], we choose a slightly different realization, more suitable for our purposes. Similar results could be obtained by directly following the algorithm [16] for a perturbed system.

We now present some results of this algorithm obtained for perturbation parameter $\delta = 10^{-3}\lambda_{\min}$, where $\lambda_{\min} = \min_{1\le n\le N}\lambda_n$. Complete list of numerical data used in these examples as well as other graphs are provided in Table 5 and Figs. 18 and 19 in Appendix A. Fig. 16 represents the graph of the revenue function in case of a three-class model.

Taking into consideration QoS requirements, shown in Fig. 16, the solution to optimization problem is found to be 4.6917 cents/s for $S_{\max}(1) = 9$ users.

Fig. 17 shows the results of calculations for the six-class model. It has to be mentioned that higher dimensional models become more exposed to the "granularity", or "round-off" effect, than smaller ones. By assuming $S_{\max}(1)$ to be an integer-valued function, we keep rejecting customers of higher priority classes in case there is only enough space for a "fraction" of such customer. When this fraction becomes large enough, a decrease in revenue can occur. Accordingly, the graph can acquire sudden drops and does

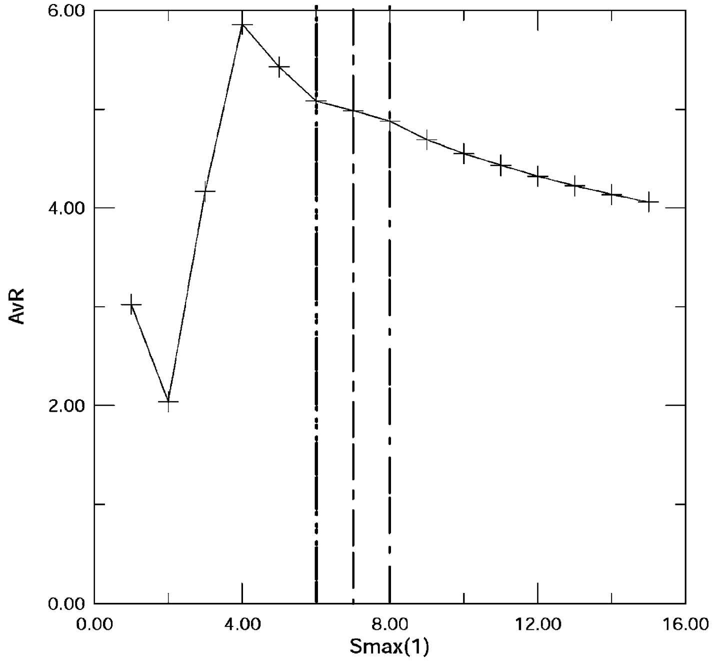

Fig. 16. Average revenue per unit time versus $S_{\max}(1)$ for a three-class problem $B = 1$ mbps; $\beta_{1,2} = 3, 4$; $C_b = 10$ cents/mbps.

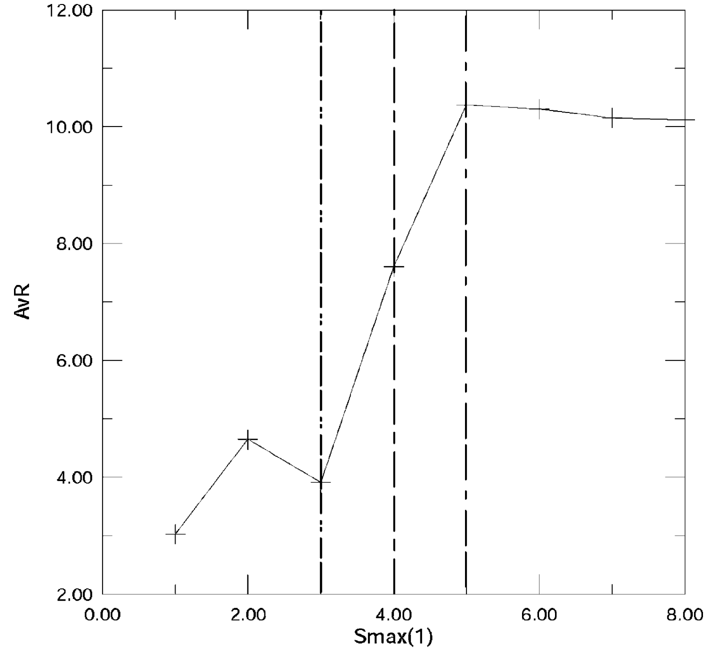

Fig. 17. Average revenue per unit time versus $S_{\max}(1)$ for a six-class problem $B = 1$ mbps; $\beta = 2$; 3.8; 4; 4.9; 5; $C_b = 10$ cents/ mbps.

not in general preserve concavity. This effect can be most readily observed in Fig. 18 in Appendix A, where priorities were chosen to be sufficiently far apart from one another. Maximal average revenue for the system considered in Fig. 17 is approximately equal to 10.302 cents/s and corresponds to the maximum of six users of class-1.

4.5. Connection admission control (CAC)

When a request for connection arrives, the NAP, based on the number of ongoing connections of each class, can decide to accept or reject the arriving request. This kind of CAC can be implemented to increase the revenue of the NAP. Notice that the connection admission policy enforced so far was to accept any arriving request that could be accommodated.

However, when requests arrive, it could be more advantageous to block a certain class of call to anticipate the arrival of a more expensive class of calls, and hence reserve room for the users who pay more. For example, a simple CAC policy would be not to accept a type-1 user if there is only “enough room” for one other type-2 user. Thus not loosing the opportunity to have a client who pays well, such policies can possibly increase the revenue. We implemented several CAC policies, progressively reserving the bandwidth for one, or two, or more than two type-2 users. This procedure surprisingly turned out to yield less average revenue than the admission policy used in previous sections.

However, a better control of the blocking probability is provided by this policy inasmuch as the vertical QoS lines were shifted to the left, allowing a larger feasible region. Hence such policies could be enforced during high peak usage periods to avoid congestion, and meet more easily the needs of the high priority users.

5. Conclusions and future work

In this paper, we modeled connections to Next Generation Networks using a bandwidth sharing mechanism. We defined an appropriate pricing scheme to charge the users for their use of the network, depending on the class of traffic and required QoS (minimum bandwidth). We formulated an optimization problem to determine the optimal resource allocation in terms of minimum bandwidth, subject to a call-blocking probability QoS constraint. We show that for both the single-class calls and the multi-class calls, an optimal value of the reserved minimum bandwidth can be found that maximizes the NAP’s revenue, and guarantees a blocking probability (QoS) lower than a certain negotiated level. We solved the problem for a general N-class system by modeling the system as an N-dimensional birth and death process. We showed that in order to plug in the algorithm in Servi et al. [15,16], we need to make a novel modification to the CTMC by suitably inserting dummy nodes. Although we only considered exponential distributions of file sizes, we conjectured (proved in case of $N = 1$) that the results for general distributions of file sizes would not be any different. We also provided a mathematical framework for describing the interaction between user behavior (in terms of arrival rates and minimum bandwidth) and pricing structures.

Future work will deal with different QoS requirements (delay, packet loss). Other pricing schemes will be considered, along with their eventual equilibrium and the effects of CAC policies. We will also focus on optimization problems to choose optimal pricing structures. We will explore more general distributions for service times for the N-class case.

Table A.1
Numerical values for different N-class models ($C_b = 10$ cents/mbps in all examples)

Number of classes, N (cents/s)	Parameters	Rejection probabilities	Maximal AvR (cents/s)	Optimal $S_{max}(1)$ (users)
3 (Fig. 16)	$\beta = 3; 4$, $\lambda = 0.25; 0.23; 0.2$ customers/s, $\alpha = 3.3; 3.1; 3$ mbps^{-1}, $C_t = 10; 10; 12$ cents/s	$\varepsilon_1 = 0.01$, $\varepsilon_2 = 0.02$, $\varepsilon_3 = 0.03$	4.6917	9
4 (Fig. A.1)	$\beta = 2; 5; 7$, $\lambda = 0.25; 0.23; 0.2; 0.15$, customers/s, $\alpha = 3.33; 3; 3.2; 3.5$ mbps^{-1}, $C_t = 12; 15; 18; 23$ cents/s	$\varepsilon_1 = 0.06$, $\varepsilon_2 = 0.1$, $\varepsilon_3 = 0.12$, $\varepsilon_4 = 0.18$	6.14879	10
5 (Fig. A.2)	$\beta = 2; 3.8; 4; 4.9$, $\lambda = 0.25; 0.23; 0.2; 0.2; 0.18$ customers/s, $\alpha = 3.33; 3.3; 3.2; 3; 2.8$ mbps^{-1}, $C_t = 10; 12; 13; 15; 16$ cents/s	$\varepsilon_1 = 0.065$, $\varepsilon_2 = 0.09$, $\varepsilon_3 = 0.1$, $\varepsilon_4 = 0.12$, $\varepsilon_5 = 0.19$	6.37743	8
6 (Fig. 17)	$\beta = 2; 3.8; 4; 4.9; 5$, $\lambda = 0.25; 0.2; 0.2; 0.2; 0.18; 0.16$ customers/s, $\alpha = 3.33; 3.3; 3.2; 3; 2.8; 2.5$ mbps^{-1}, $C_t = 10; 12; 13; 15; 16; 17$ cents/s	$\varepsilon_1 = 0.01$, $\varepsilon_2 = 0.03$, $\varepsilon_3 = 0.1$, $\varepsilon_4 = 0.12$, $\varepsilon_5 = 0.14$, $\varepsilon_6 = 0.2$	10.3023	6

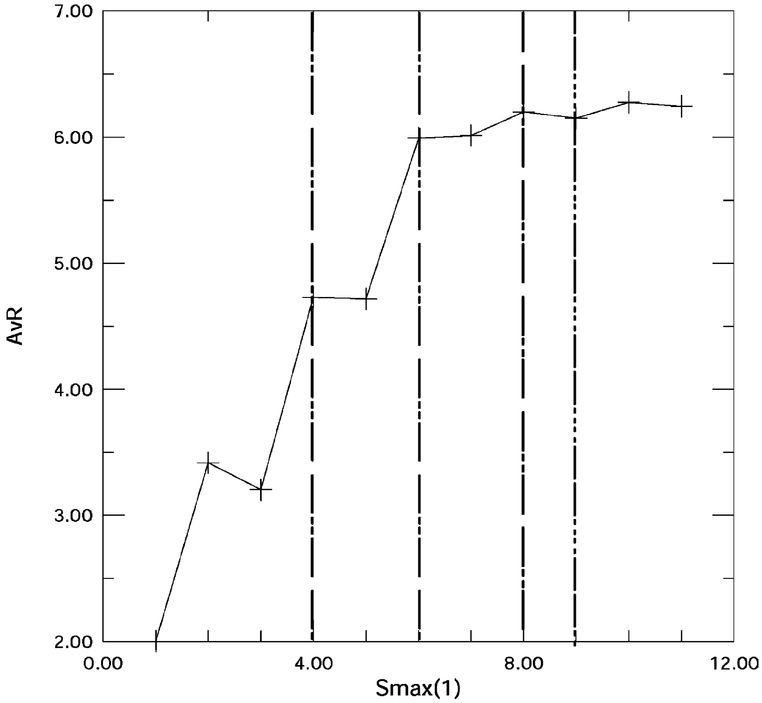

Fig. A.1. Average revenue per unit time for a four-class model.

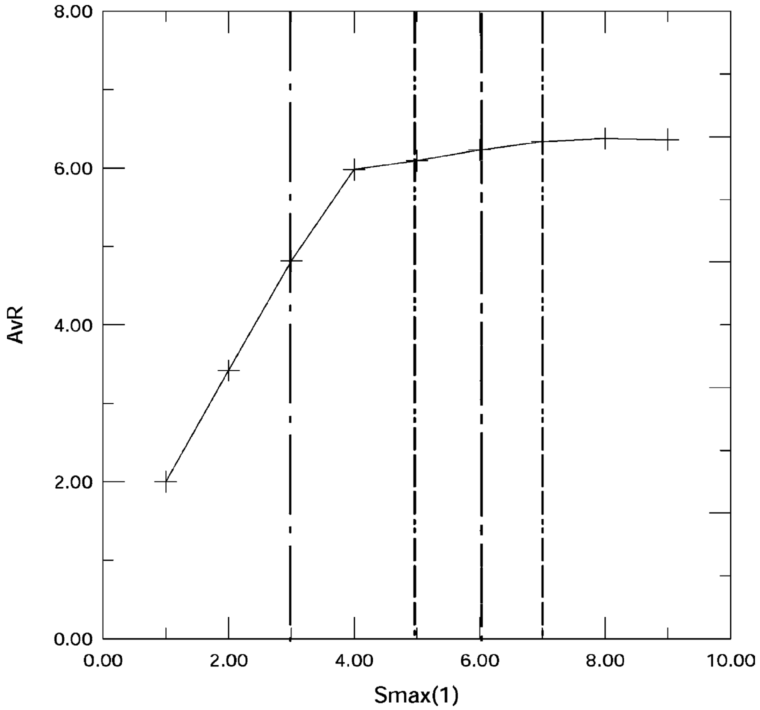

Fig. A.2. Average revenue per unit time for a five-class model.

Acknowledgements

The authors thank all the anonymous reviewers for their suggestions that led to considerable improvements in the content and presentation of this paper. The authors also express gratitude to Donna Ghosh for carefully going through the paper and offering valuable criticisms.

Appendix A

Numerical values for different N-class models are given in Table A.1. Average revenue per unit time for a four-class model and five-class model are given in Figs. A.1 and A.2, respectively.

References

[1] R. Cocchi, D. Estin, S. Shenker, L. Zhang, Pricing in computer networks: motivation, formulation and example, ACM/IEEE Trans. Netw. 1 (1993) 614–627.
[2] R.J. Edell, N. McKeown, P.P. Varaiya, Billing users and pricing for TCP, IEEE J. Sel. Area Commun. 13 (7) (1005) 1–14.
[3] F.P. Kelly, Charging and Accounting for Bursty Connections, Internet Economics, MIT Press, Cambridge, MA, 1996.
[4] L. Kleinrock, Queueing Systems, Wiley, New York, 1975.
[5] V.G. Kulkarni, Modeling and Analysis of Stochastic Systems, Texts in Statistical Science Series, Chapman & Hall, London, 1995.

[6] J.K. MacKie-Mason, H. Varian, Pricing the Internet, Public Access to the Internet, MIT Press, Cambridge, MA, 1995, pp. 269–314.

[7] A.K. Parekh, R.G. Gallager, A generalized processor sharing approach to flow control in integrated services networks: the single node case, IEEE/ACM Trans. Netw. 1 (3) (1993) 344–357.

[8] C. Parris, S. Keshav, D. Ferrari, A framework for the study of pricing in integrated networks, Technical Report TR-92-016, International Computer Science Institute, Berkeley, CA, p. 1002.

[9] S. Shenker, D. Clark, D. Estrin, S. Herzog, Pricing in Computer Networks: Reshaping the Research Agenda, Newblock, 1996.

[10] G. de Veciana, G. Kesidis, Bandwidth allocation for multiple qualities of service using generalized processor sharing, IEEE Trans. Info. Th. 42 (1) (1996) 268–271.

[11] J. Walrand, An Introduction to Queueing Networks, Prentice-Hall, Englewood Cliffs, NJ, 1988.

[12] J. Walrand, P. P. Varaiya, High-Performance Communication Networks, Morgan Kaufman, Los Altos, CA, 1996.

[13] Z. Wang, USD: User-Share Differentiation. http://alternic.net/drafts/drafts-w-x/draft-wang-diff-serv-usd-00.html, 2000.

[14] R. Jain, J.M. Smith, Modeling vehicular traffic flow using M/G/C/C state dependent queueing models, Transport. Sci. 31 (1997) 324–336.

[15] L.D. Servi, T. Gerhardt, S. Humair, Fast, accurate solution to large birth–death problems, Stoch. Model, submitted for publication.

[16] L.D. Servi, Algorithmic solutions to recursively tridiagonal linear equations with applications to multi-dimensional birth–death processes, INFORMS J. Comput., submitted for publication.

[17] G.H. Golub, C.F. Van Loan, Matrix Computations, Johns Hopkins University Press, Baltimore, MD, 1989.

[18] S.M. Ross, Introduction to Probability Models, Academic Press, New York, 2000.

[19] B. Teitelbaum, P. Chimento, QBone Bandwidth Broker Architecture. http://qbone.internet2.edu/bb/bboutline2.html.

M. Yacoubi is an Information Technology consultant with Deloitte Consulting. He specializes in Systems Implementations in the area of Customer Relationship Management for Telecommunications, Manufacturing, and Financial Services industries. He holds an Engineering Diploma from the Ecole Centrale de Lyon (France) and received his M.S. in Industrial Engineering and Operations Research from the Pennsylvania State University.

M. Emelianenko is a doctoral student in the Department of Mathematics at the Pennsylvania State University. She received her B.S. and M.S. degrees in Computer Science and Mathematics from the Moscow State University, Russia and M.A. degree in Mathematics from the Pennsylvania State University. She is a student member of American Mathematical Society. Her research interests include numerical analysis and modeling in the areas of computer systems and telecommunications.

N. Gautam is an Assistant Professor in the Harold and Inge Marcus Department of Industrial and Manufacturing Engineering at the Pennsylvania State University. He received his B.Tech. degree in Mechanical Engineering from the Indian Institute of Technology, Madras, and his M.S. and Ph.D. in Operations Research from the University of North Carolina at Chapel Hill. He is a member of IEEE, INFORMS and MAA, and a senior member of IIE. He is an Associate Editor for the INFORMS Journal on Computing and the Newsletter Editor as well as Website Editor for the INFORMS Applied Probability Society. His research interests are in the areas of modeling, analysis and performance evaluation of computer, telecommunication and information systems.