Admission Control of CBR/VBR and ABR/UBR Call Arrival Streams: A Markov Decision Approach

Ernst Nordström

Department of Computer Systems, Uppsala University,
Box 325, S–751 05 Uppsala, Sweden
Tel: +46 18 18 31 11
Fax: +46 18 55 02 25
ernstn@DoCS.UU.SE

Søren Blaabjerg

Institute of Telecommunications, Technical University of Denmark
Building 371, DK–2800 Lyngby, Denmark

Gábor Fodor

Department of Telecommunications and Telematics, Technical University of Budapest,
Sztoczek 2, H–1111 Budapest, Hungary

Abstract

This paper evaluates a Markov decision approach to Connection Admission Control of guaranteed services and best effort services. Two different schemes that support integration of guaranteed services and best effort services are evaluated: a preemptive scheme and a partial blocking scheme. The Markov decision approach finds policies that are optimal in terms of long-term reward. The Markov decision policy performs intelligent blocking which implements bandwidth reservation for wide-band traffic. Numerical results with three traffic classes, a narrow-band and a wide-band guaranteed class and a narrow-band best effort class, show that the Markov decision method, applied to both the preemptive and the partial blocking scheme, yields higher long-term reward than the complete sharing method when the amount of wide-band guaranteed traffic is large. The results also show that the preemptive scheme and the partial blocking scheme are efficient for different types of traffic mixes.

1. Introduction

Connection Admission Control (CAC) in Asynchronous Transfer Mode (ATM) networks should support an efficient integration of the Variable Bit Rate (VBR), Constant Bit Rate (CBR), Available Bit Rate (ABR) and Unspecified Bit Rate (UBR) service classes. One of the main design issues is how to share the capacity between guaranteed services (CBR and VBR) and best effort services (ABR and UBR). The design must utilize that fact that best effort calls have the ability to reduce their bandwidth in case of congestion. Two methods that meet this constraint are the preemptive scheme and the partial blocking scheme.

In the preemptive scheme, best effort calls are preemptied when guaranteed service calls arrive to a busy link. In this paper, the best effort calls that are chosen for preemption are the ones with most recent arrival times. When calls depart from the link such that sufficient free capacity becomes available, a preemptied best effort call enters service again. The preemptive scheme was analyzed in [3] in the case when all calls enter a queue before service. It was found that the scheme is capable of improving the link utilization at the expense of fairness. The common FIFO policy was shown to maintain fairness at some expense of link utilization.

In the partial blocking scheme [1, 2], the best effort services adapt their bandwidth requirement to the available capacity such that the bandwidth - holding time product remains constant. Each best effort call can specify a minimal accepted service ratio, $r_{min} \in (0,1]$ (in addition to the bandwidth requirement b) which is used in the call negotiation process. A best effort call is accepted only if the available bandwidth b_a fulfills the criteria: $r_{min} b \le b_a \le b$. Thus, throughout the lifetime of a call, the instantaneous service rate $r(t)$, defined as $b_a(t)/b$, may fluctuate according to the current load and available capacity on the link. The partial blocking scheme was analyzed in [1, 2] were it was found that the scheme gives low blocking probability and efficient link utilization for best effort calls.

The purpose of this paper is twofold. First, we evaluate the efficiency of CAC based on Markov decision theory for the preemptive scheme and the partial blocking scheme. Second, we compare the performance of the preemptive scheme and the partial blocking scheme in terms of long-term reward and the average time an ABR call spends in the system.

Markov decision theory gives an efficient technique to find an optimal CAC policy in terms of long- term reward. The Markov decision policy maps states to admission decisions (actions), i.e. accept or reject. The Markov decision approach evaluates the long-term reward of each action in each state, and chooses the action which maximizes the reward. The evaluation is based on a Markov model of the decision task, which comprises the state transition probabilities and the expected reward delivered at each state transition. The decision task model is parameterized by the call arrival and departure rates, which are supposed to be measured on line. The CAC policy is adapted to actual traffic demand at regular intervals.

The Markov decision technique has been applied to the link access control problem [7] and the network routing problem [4] assuming that blocked calls are lost. The technique has also been applied to link allocation [8] and routing problems [5] in the context of blockable narrow-band and queueable wide-band call traffic.

Several alternative approaches to call-level CAC have been proposed in the literature, e.g. partial sharing (class limitation), trunk reservation and dynamic trunk reservation. The comparison presented in [6] indicates that for many cases, the trunk reservation and dynamic trunk reservation policies can provide fair, efficient solutions, close to the optimal Markov decision policy.

This paper is organized as follows. In the next section, the CAC problem is introduced. Section 3 presents a Markov decision model for the CAC task for the preemptive scheme and for the partial blocking scheme. Section 4 describes the policy iteration technique of Markov decision theory in which the value determination problem is handled by successive approximation. Section 5 presents the numerical results. Finally, section 6 concludes the paper.

2. The CAC problem

In the CAC problem, a link with capacity C [units/s] is offered calls from K traffic classes of CBR[1] and ABR calls. Calls belonging to class $j \in J = \{1, 2, ... K\}$ have the same bandwidth requirements and similar arrival and holding time dynamics. For ease of presentation, we consider K=3 traffic classes (two CBR classes and one ABR class) throughout the rest of this paper. The CBR classes are indexed by 1 and 2, and the ABR class is indexed by 3.

We assume that class-j calls with peak bandwidth requirement b_j arrive according to a Poisson process with average rate λ_j [s^{-1}], and that the CBR call holding time is exponentially distributed with average $1/\mu_j$ [s]. The ABR call holding time for the preemptive scheme and the partial blocking schemes is exponentially distributed with average $1/\mu_3$ in the case when the call experiences no preemption and no partial blocking, respectively. If the ABR calls are partially blocked, the call holding time can be calculated by techniques from Markov driven workload processes, see [2].

The task is to find a CAC policy π that maps *request states* $(j,x) \in J \times X$ to *admission actions* $a \in A$, $\pi: J \times X \rightarrow A$, such that the long-term reward is maximized. In the rest of this paper we assume that the long-term reward is proportional to the throughput at the call level. The set A contains the possible admission actions, {*ACCEPT*, *REJECT*}. The set X contains all feasible system states. For the preemptive scheme it is given by:

$$X_1 = \left\{(n_1, n_2, n_3, p) : \; p = 0, \; n_1, \; n_2, \; n_3 \geq 0, \; n_1 b_1 + n_2 b_2 + n_3 b_3 \leq C\right\} \; \bigcup$$
$$\left\{(n_1, n_2, n_3, p) : \; p \in \{1, 2, ..., p_{max}\}, \; n_1, \; n_2, \; n_3 \geq 0, \; n_1 b_1 + n_2 b_2 + n_3 b_3 = C\right\}, \tag{1}$$

where n_j is is the number of class-j calls accepted on the link, and p is the number of preemptied ABR calls, which can take on the values $p \in P = \{0,1, ..., p_{max}\}$. For later use, we also introduce the set of feasible link states for the preemptive scheme:

$$N = \left\{(n_1, n_2, n_3) : n_1, \; n_2, \; n_3 \geq 0, \; n_1 b_1 + n_2 b_2 + n_3 b_3 \leq C\right\}. \tag{2}$$

1. VBR calls can be modelled the same way adopting the notion of effective bandwidth.

For the partial blocking scheme, the set of feasible system states is given by:

$$X_2 = \left\{(n_1, n_2, n_3) : n_1,\ n_2,\ n_3 \geq 0,\ n_1 b_1 + n_2 b_2 + n_3 b_3 r_{min} \leq C\right\}, \tag{3}$$

where $r_{min} \in (0,1]$ is the minimal accepted service ratio for the ABR class.

3. A Markov Decision Model for CAC

This section presents a Markov decision model for CAC for the preemptive scheme and the partial blocking scheme. The Markov decision model specifies a Markov chain which is controlled by actions in each state. The actions result in state transitions and reward delivery to the system. The control objective is to find the actions that maximize the reward accumulated over time. In the current application, the Markov chain evolves in continuous time, and we therefore face a semi-Markov decision problem (SMDP).

The SMDP state x corresponds to the system state in the previous section, i.e. $x=(n_1,n_2,n_3,p) \in X_1$ for the preemptive scheme, and $x=(n_1,n_2,n_3) \in X_2$ for the partial blocking scheme. The SMDP action a is represented by a vector $a=(a_1,a_2,a_3)$, corresponding to admission decisions for presumptive call requests. The action space for both the preemptive and the partial blocking scheme becomes:

$$A = \{(a_1,a_2,a_3) : a_j \in \{0,1\}, j \in J\}. \tag{4}$$

were $a_j=0$ denotes call rejection and $a_j=1$ denotes call acceptance. The permissible action space in state x is a state-dependent subset of A. For the preemptive scheme, the permissible action space becomes:

$$A_1(x) = \{(a_1,a_2,a_3) \in A\text{: } a_j=0 \text{ if } n+\delta_j - min(b_j/b_3, n_3)\delta_3 \notin N \text{ or } p+b_j/b_3 \notin P, j \in \{1,2\},\ a_3=0 \text{ if } n+\delta_3 \notin N\} \tag{5}$$

where $n=(n_1,n_2,n_3)$ and δ_j denotes a vector with zeros except for a one at position j. Note that an arriving CBR call will be rejected if there is insufficient free capacity even after preemption of ABR calls, or if the maximum number of preemptied ABR calls will be exceeded.

For the partial blocking scheme, the permissible action space becomes:

$$A_2(x) = \{(a_1,a_2,a_3) \in A\text{: } a_j = 0 \text{ if } n+\delta_j \notin X_2, j \in J\}. \tag{6}$$

The Markov chain is characterized by state transition probabilities $p_{xy}(a)$ which expresses the probability that the next state is y, given that action a is taken in state x. For the preemptive scheme, the state transition probabilities become:

$$p_{xy}(a) = \begin{cases} \lambda_j a_j \tau(x,a), & \begin{array}{l} n_y = n_x + \delta_j \in N, \\ p_y = p_x = 0, \end{array} \quad j \in J, \\ \lambda_j a_j \tau(x,a), & \begin{array}{l} n_y = n_x + \delta_j - min(b_j/b_3, n_3)\delta_3 \in N,\ n_x + \delta_j \notin N, \\ p_y = p_x + min(b_j/b_3, n_3) \in P, \end{array} \quad j \in \{1,2\}, \\ n_{xj}\mu_j \tau(x,a), & \begin{array}{l} n_y = n_x - \delta_j + min(b_j/b_3, p_x)\delta_3 \in N, \\ p_y = max(p_x - b_j/b_3, 0) \in P, \end{array} \quad j \in J, \\ n_{xj}\mu_j \tau(x,a), & \begin{array}{l} n_y = n_x - \delta_j \in N, \\ p_y = p_x = 0, \end{array} \quad j \in J, \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

where the quantity $\tau(x,a)$ denotes the average sojourn time in state x: $\tau(x,a)=\{\sum_{j \in J}[n_{xj}\mu_j + \lambda_j a_j]\}^{-1}$.

The first term in the state transition probability expression above gives the state transition probability for a CBR or ABR call arrival to a link with some free capacity without any preemption of ABR calls. The second term gives the state transition probability for a CBR call arrival to a link with sufficient free capacity after preemption of ABR calls. The third term gives the state transition probability for CBR or ABR call departures when the preemption queue is non-empty. The fourth term gives the state transition probability for CBR or ABR call departures when the preemption queue is empty.

For the partial blocking scheme, the state transition probabilities become:

$$p_{xy}(a) = \begin{cases} \lambda_j a_j \tau(x,a), & n_y = n_x + \delta_j \in X_2, \quad j \in J \\ n_{xj}\mu_j \tau(x,a), & n_y = n_x - \delta_j \in X_2, \quad j \in \{1,2\} \\ n_{x3}\mu_3 r(x)\tau(x,a), & n_y = n_x - \delta_3 \in X_2, \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

where $r(x)$ denotes the instantaneous service rate at state x: $r(x)=(C-n_{x1}b_1-n_{x2}b_2)/(n_{x3}b_3)$. The average sojourn time in state x is given by $\tau(x,a)= \{n_{x1}\mu_1 + n_{x2}\mu_2+n_{x3}\mu_3 r(x) +\sum_{j\in J}\lambda_j a_j\}^{-1}$.

The expected accumulated reward in state x is given by $R(x,a)=q(x)\tau(x,a)$. For the preemptive scheme the reward accumulation rate is given by $q(x)=\Sigma_{j\in J}\, r_j n_{xj}\mu_j$. For the partial blocking scheme the reward accumulation rate is given by $q(x)=r_1 n_{x1}\mu_1+r_2 n_{x2}\mu_2 + r_3 n_{x3}\mu_3 r(x)$. The quantity r_j is the absolute reward of carrying a type-j call. In order to maximize the overall call level throughput, r_j should be equal to the product of class-j's bandwidth requirement and its mean holding time. For the preemptive scheme, $r_j=b_j/\mu_j$, and for the partial blocking scheme $r_1=b_1/\mu_1$, $r_2=b_2/\mu_2$ and $r_3=r(x)b_3/(\mu_3 r(x))$.

In order to solve the value determination step of the Markov decision task, the continuous-time SMDP model must first be transformed into a discrete-time MDP model [10]:

$$\tilde{R}(x,a) = \frac{\tau}{\tau(x,a)} R(x,a) \qquad x \in X \text{ and } a \in A(x),$$

$$\tilde{p}_{xy}(a) = \begin{cases} \dfrac{\tau}{\tau(x,a)}\, p_{xy}(a), & y \neq x,\ x\in X \text{ and } a\in A(x), \\ \dfrac{\tau}{\tau(x,a)}\, p_{xy}(a) + \left[1-\dfrac{\tau}{\tau(x,a)}\right], & y = x,\ x\in X \text{ and } a\in A(x), \end{cases} \tag{9}$$

where τ is the size of the discrete time step, chosen such that $0<\tau \le \min_{x,a} \tau(x,a)$. In the preemptive scheme, $\tau = \{\sum_{j\in J} [\, N_j\,\mu_j + \lambda_j]\}^{-1}$, where $N_j=C/b_j$ denotes the maximum number of class-j calls carried by the link. In the partial blocking scheme, τ can be computed as $\tau = \{N_1\mu_1 + N_2\mu_2 + N_3\mu_3 r_{min} + \sum_{j\in J}\lambda_j\}^{-1}$, where $N_3=C/(b_3 r_{min})$.

4. Adaptive Policy Iteration

This section describes a method for solving the CAC task, formulated as a semi-Markov decision problem. The method of choice is *policy iteration*, which is one of the computational techniques within Markov decision theory to determine an optimal policy. Another computational technique is *reinforcement learning*, which can be used in a model-free way to do Markov decision optimization [9].

A fundamental quantity of Markov decision theory is the *evaluation function.* The evaluation function is defined for each state in the state space and measures the accumulated reward received during an infinite time interval, starting in the given state. The evaluation function is used as a tool to find the optimal policy.

The policy iteration approach computes a series of improved policies in an iterative manner. The computation of an improved policy π_{k+1} from the current policy π_k involves three steps:

- task identification
- value determination
- policy improvement

The first step involves determining the Markov decision model, i.e. the state transition probabilities and the expected rewards. These quantities are parameterized by link call arrival rates λ_j and call departure rates μ_j, see section 3. The arrival/departure rates are obtained from measurements to make Markov decision model adaptive to actual traffic characteristics. The measurement period corresponds to the policy improvement period. The measurement period should be of sufficient duration for the system to attain statistical equilibrium.

The second step involves computing the evaluation function for the current policy. This is efficiently done by the method of successive approximations which relies on a basic equation of Markov decision theory. The equation states that the reward received within n decision epochs starting in given state x, should equal the expected immediate reward received after the first decision epoch, plus the expected accumulated reward within n–1 decision epochs starting from the neighbor states $\{y\}$:

$$V_n(x, \pi_k) = \tilde{R}(x, a) + \sum_{y \in X} \tilde{p}_{xy}(a) V_{n-1}(y, \pi_k) \qquad ; x \in X. \tag{10}$$

A full description of the method of successive approximations can be found in [10], see also [8]. The method can be proved to converge to the correct evaluation function in a finite number of steps, provided that the state and action space are finite [10].

The third step is the actual policy improvement. The new action in each state (the new policy π_{k+1}) is determined by searching for the action that maximizes the sum of the immediate reward and the expected evaluation of the neighbor states:

$$max_{a \in A(x)} \left\{ \tilde{R}(x, a) + \sum_{y \in X} \tilde{p}_{xy}(a) V(y, \pi_k) \right\} \qquad ; x \in X. \tag{11}$$

Where $V(y,\pi_k)$ denotes the evaluation function obtained from the value determination step. Since the search involves the evaluation function of the current policy π_k and not of the new policy π_{k+1} (which is unknown) we are not sure to find the optimal action. However, the method can be proved to converge to an optimal policy in a finite number of iterations in the case of finite state and action space [10].

During a policy improvement period, new calls are allocated according to a function obtained from the Markov decision computations: the admission gain function. The gain function measures the increase in long-term reward induced by the control action. The admission gain function is given by $V(x_{accept},\pi_k)–V(x_{reject},\pi_k)$, where x_{accept} is the state after call acceptance and x_{reject} is the state after call rejection. The call is rejected if the admission gain is negative.

The proposed method can be summarized as follows. Choose an initial CAC policy and an evaluation function. During a finite period, allocate calls according to the admission gain function associated with the chosen evaluation function. At the same time, measure traffic statistics (call arrival rates and call departure rates) in order to determine the Markov decision task for the current policy. Evaluate the applied policy in the context of the current Markov decision task, using the method of successive approximations, and improve the policy. Apply the new policy during the next period, measure the traffic statistics and repeat the policy evaluation and the policy improvement step and so forth.

5. Numerical Results

This section evaluates the performance of two CAC methods for the preemptive scheme and the partial blocking scheme: the Markov decision (MD) method and the complete sharing (CS) method. Performance measures of interest are the call level throughput and the average time an ABR call spends in the system (the call holding time). For the preemptive scheme, the average number of preemptied calls and the preemption probability are also evaluated.

The results are based on simulations for a single link with capacity C=24 [units/s], which is offered different mixes of CBR (class 1 and 2) and ABR (class 3) traffic. The bandwidth requirements are b_1=1, b_2=6 and b_3=1 [units/s] and the mean call holding times $1/\mu_1=1/\mu_2=1/\mu_3=1$ [s], assuming that the ABR calls experiences no preemptin and no partial blocking.

The arrival rates λ_1, λ_2, and λ_3 were varied so that:

$$\frac{b_1\lambda_1}{C\mu_1} + \frac{b_2\lambda_2}{C\mu_2} + \frac{b_3\lambda_3}{C\mu_3} = \rho \tag{12}$$

with the arbitrary constraint λ_3=λ_1. The figures show curves for two different load values (an underload with ϱ=0.8 and an overload with ϱ=1.2). A step size of 0.2 in the arrival rate ratio $(\lambda_1+\lambda_3)/\lambda_2$ has been used when plotting all the figures. Moreover, the curves presented in the figures are obtained after averaging over 30 simulation runs and 95% confidence intervals, computed assuming normal distributed values, are also shown for each curve.

Figure 1 shows the call level throughput for the preemptive scheme for different arrival rate ratios and load values. As can be seen in the figure, the throughput of the MD method is higher than for the CS method when arrival rate ratio is less than 2 (for load 0.8) or 2.5 (for load 1.2). This is due to the fact that the MD method implements so called “intelligent blocking” of narrow-band calls, either in individual link states or completely. By rejecting narrow-band call requests, typically when the free capacity equals the size of a wide-band call, bandwidth is reserved for the wide-band class, which increases the long-term reward. The policy computed in the simulations implemented complete intelligent blocking for the narrow-band CBR class when $(\lambda_1+\lambda_3)/\lambda_2 \leq 0.2$ and $(\lambda_1+\lambda_3)/\lambda_2 \leq 0.4$ at the load 0.8 and 1.2, respectively. For higher values on the arrival rate ratio, the policy blocked narrow-band CBR calls at individual states, e.g. in the non-preemptive link state (n_1,n_2,n_3)=(0, 2, 6) and the preemptive link state (0, 3, 6). Narrow-band ABR calls were never completely blocked, but blocked at individual states, e.g. in link states (5, 2, 1) and in (15, 0, 3). This may seem surprising since ABR calls will be preemptied when a CBR call arrives to a busy link. Simulations with the permissible action space modified such that ABR calls always are accepted showed no improvement over the permissible action space used in this paper, i.e. an arriving ABR call can be rejected although there is sufficient free capacity on the link.

When the narrow-band CBR class is completely blocked, we face a severe fairness problem. However, the complete blocking can be avoided by increasing the absolute reward r_1 of carrying a narrow-band CBR call. Of course, we can not expect the throughput to be as high as when the narrow-band CBR class is completely blocked since the blocking probability for the wide-band CBR class will increase. Nevertheless, changing the absolute reward parameters is a simple way to control the distribution of blocking probabilities among different call classes [4].

Figure 2 shows the average number of preemptied ABR calls for the MD method for the load 0.8 and 1.2. Note that the average number of preemptied calls is higher when amount of ABR traffic is high. The maximum number of preemptied calls (p_{max}) was set to 25 in the simulations, i.e. 10 times the maximum average number of preemptied calls.

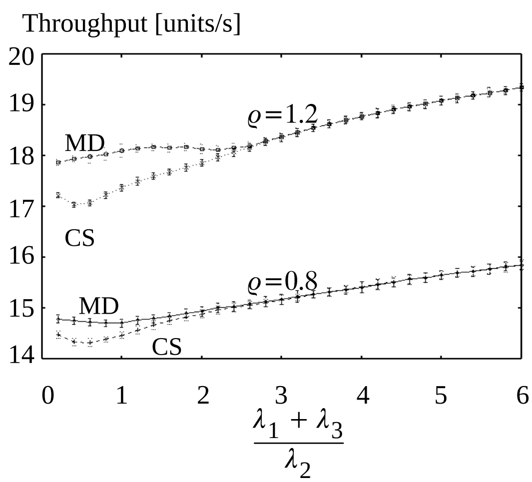

Figure 1: Call level throughput for different arrival rate ratios and load values for the preemptive scheme.

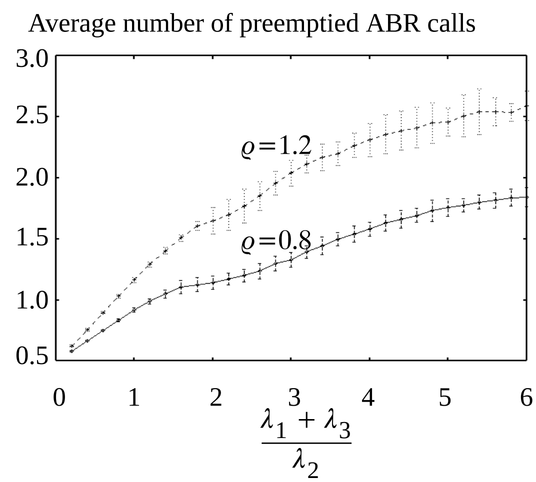

Figure 2: Average number of preemptied calls for different arrival rate ratios and load values for the preemptive/MD scheme.

Figure 3 and 4 shows the average time an ABR call spends in the system in the preemptive scheme with the MD method for different arrival rate ratios for a load of 0.8 and 1.2, respectively. Three different curves are shown in each figure. The lower curve shows the average system time for calls that are not preemptied. The middle curve shows the average system time taking all calls (preemptied and not preemptied) into account. The upper curve shows the average system time for calls that are preemptied. The average system time obviously increases if the calls are subject to preemption. Note that for an arrival rate ratio larger than 3, the average system time will be close to 1, since the probability of preemption is close to 0 (see figure 5 and 6). The lower curve is below 1 since short calls are more likely not to be preemptied.

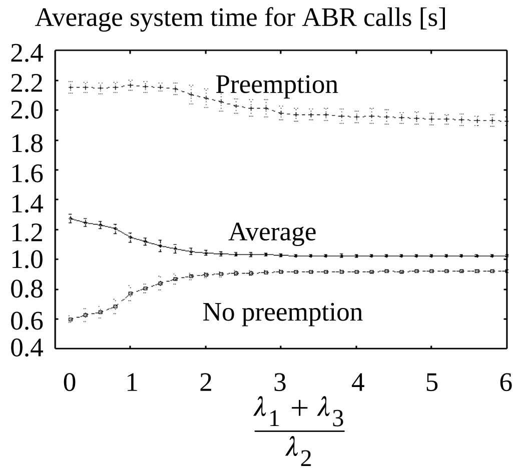

Figure 3: Average system time for ABR calls for different arrival rate ratios for load 0.8 for the preemptive/MD scheme.

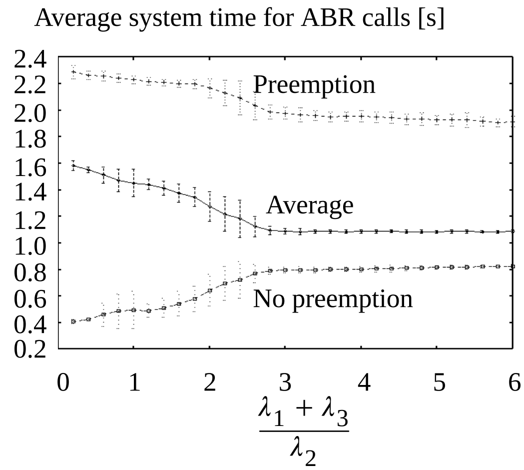

Figure 4: Average system time for ABR calls for different arrival rate ratios for load 1.2 for the preemptive/MD scheme.

Figure 5 and 6 shows that preemption probability for the preemptive scheme with the MD method for different arrival rate ratios for a load of 0.8 and 1.2, respectively. Two different curves are shown in each figure. The upper curve shows the probability of preemption occurring 1 or more times. The lower curve shows the probability of preemption occurring 2 or more times. Note that there is a high probability of preemption when the amount of ABR traffic is low compared to the amount of wide-band CBR traffic.

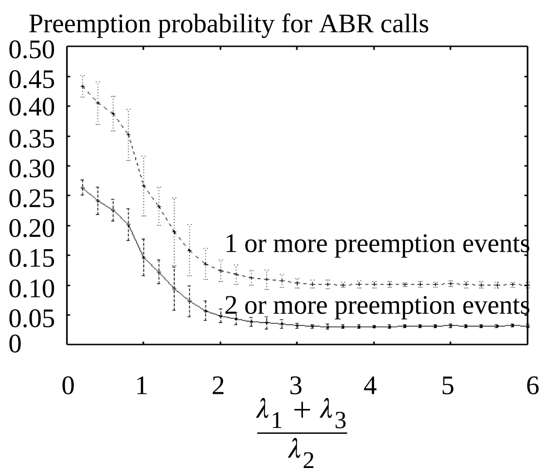

Figure 5: Preemption probability for ABR calls for different arrival rate ratios for load 0.8 for the preemptive/MD scheme.

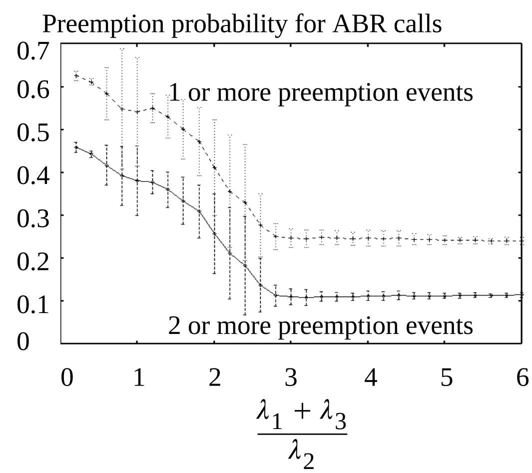

Figure 6: Preemption probability for ABR calls for different arrival rate ratios for load 1.2 for the preemptive/MD scheme.

Figure 7 and 8 shows the call level throughput for the partial blocking scheme for different arrival rate ratios and load values for r_{min}=0.5 and r_{min}=0.25, respectively. The computed MD policy blocks all narrow-band CBR and ABR calls when $(\lambda_1+\lambda_3)/\lambda_2 \leq 0.4$ and $(\lambda_1+\lambda_3)/\lambda_2 \leq 0.6$ at the load 0.8 and 1.2, respectively. For higher values on the arrival rate ratio, intelligent blocking of narrow-band CBR calls is sometimes performed, e.g. in the non-squeezing link state (6, 2, 0) and in the squeezing link state (0, 2, 12), assuming r_{min}=0.5. Intelligent blocking of ABR calls is performed in very few states, e.g. in link states (0, 2, 0) and (0, 3, 0). As the figures show, the intelligent blocking at individual states did not provide any significant throughput gain compared to the complete sharing method.

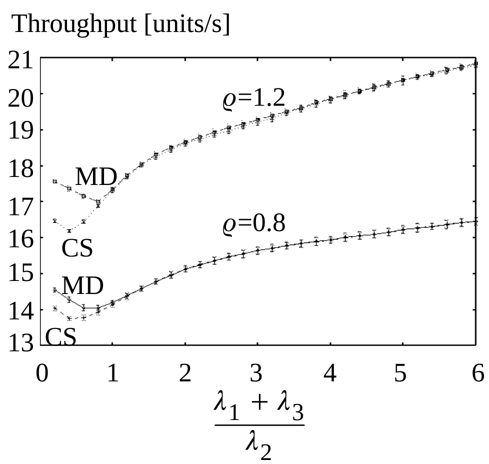

Figure 7: Call level throughput for different arrival rate ratios and load values for r_{min}=0.5 for the partial blocking scheme.

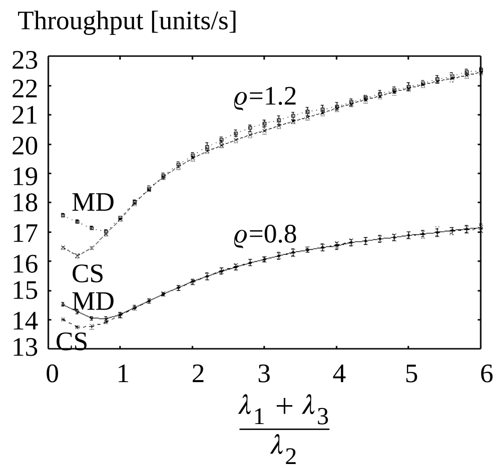

Figure 8: Call level throughput for different arrival rate ratios and load values for r_{min}=0.25 for the partial blocking scheme.

Figure 9 and 10 shows the average system time for ABR calls in the partial blocking scheme with the MD method for different arrival rate ratios and load values for r_{min}=0.5 and r_{min}=0.25, respectively. Note that at the lower load, the average system time for ABR calls is close to 1, meaning that most ABR calls do not experience any partial blocking. At the higher load, a significant fraction of the ABR calls will experience partial blocking, in particular when r_{min} =0.25, so that the average holding time will be larger than 1.

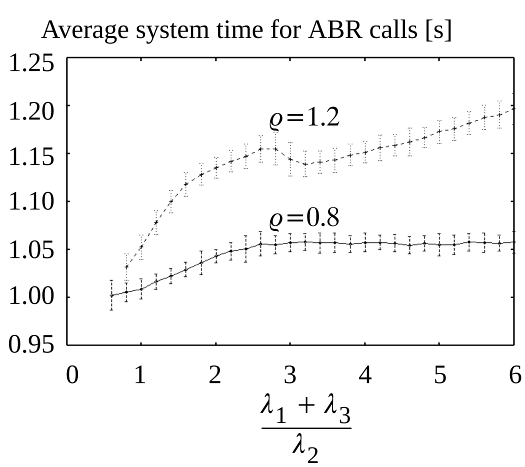

Figure 9: Average system time for ABR calls for different arrival rate ratios and load values for r_{min}=0.50 for the partial blocking scheme.

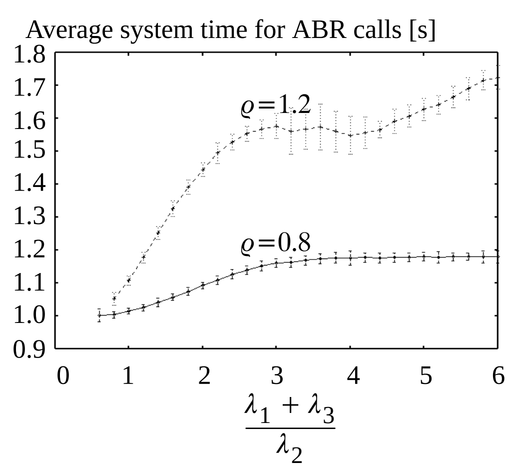

Figure 10: Average system time for ABR calls for different arrival rate ratios and load values for r_{min}=0.25 for the partial blocking scheme.

For comparison, figure 11 and 12 shows the call level throughput for the preemptive and the partial blocking scheme for CAC based on the MD method for load 0.8 and 1.2, respectively. The preemptive scheme is more efficient when the arrival rate ratio is less than 1.5, otherwise the partial blocking scheme is more efficient. The reason for the advantage of preemption at low values of the arrival rate ratio is that successful call squeezing for wide-band CBR calls requires at least 8 or 12 active ABR calls when r_{min}=0.25 and r_{min}=0.5, respectively. The reason for the advantage of partial blocking at high values of the arrival rate ratio is that the partial blocking scheme on average can accept more ABR calls than the preemptive scheme. Note that when the preemptive scheme or the partial blocking scheme gives higher throughput, the average system time for ABR calls will also be larger.

One possible extension of the the preemptive scheme could be to allow an arriving ABR call to enter the preemption queue directly when it finds the link busy. The extended preemptive scheme will reduce the blocking probability for ABR calls, but increase the blocking probability for CBR calls, which is not desirable. The reason is that if we allow the ABR calls to enter the preemption queue directly when the link is busy there is a risk that the preemption queue will fill up so that arriving CBR calls must be rejected. Moreover, in order to limit the holding time for ABR calls, the maximal size of the preemption queue should not be too large.

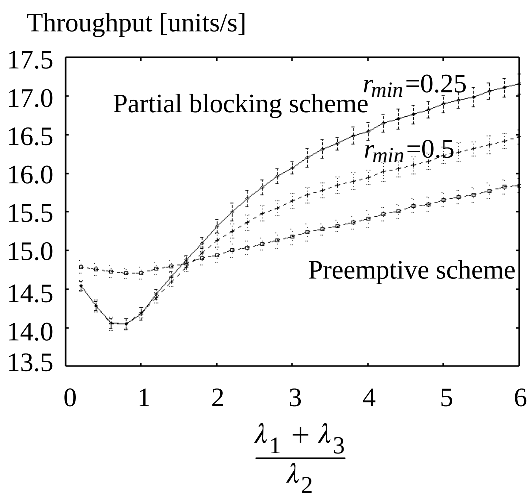

Figure 11: Throughput comparison between the preemptive/MD and the partial blocking/MD scheme for different arrival rate ratios for load 0.8.

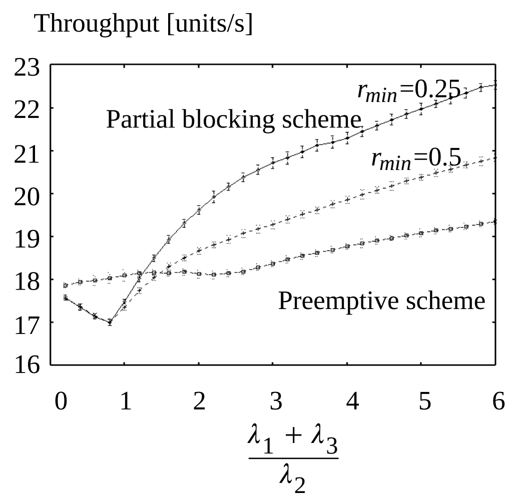

Figure 12: Throughput comparison between the preemptive/MD and the partial blocking/MD scheme for different arrival rate ratios for load 1.2.

The results presented in the figures were obtained after 5 adaptation epochs with the adaptive policy iteration method. The adaptation period contained 1000 simulated call events. The discrete time step τ used by the value determination algorithm were set to 1/60. The performance values in the figures are based on measurements of 300 000 call events after policy convergence.

6. Conclusion

This paper has evaluated the efficiency of Connection Admission Control (CAC) based on Markov decision theory for two schemes that supports integration of guaranteed services and best effort services: a preemptive scheme and a partial blocking scheme. The Markov decision technique can be used to compute CAC policies that are optimal in terms of long-term reward. The optimality is achieved by intelligent blocking of narrow-band calls, either completely, or at link states where typically the free capacity equals the size of a wide-band call.

The presented numerical results showed that the Markov decision method yields higher long-term reward than the complete sharing method when the fraction of narrow-band (CBR and ABR) traffic is low. The advantage of the Markov decision method is larger for the preemptive scheme than for the partial blocking scheme. A deeper analysis of the properties of the optimal Markov decision policy for the preemptive scheme and the partial blocking scheme is part of future work.

The numerical results also showed that the preemptive scheme and the partial blocking scheme have complementary regions of high efficiency. Preemption is more efficient when the amount of narrow-band traffic is low, partial blocking is more efficient when the amount of narrow-band traffic is high. A combined preemptive-partially blocking scheme therefore seems worth while to investigate, both analytically and numerically, including the evaluation of Markov decision based CAC. A general observation is that high throughput can only be achieved if the ABR calls are delayed (either preemptied and/or partially blocked).

Acknowledgements

Ernst Nordström was financially supported by NUTEK, the Swedish National Board for Industrial and Technical Development. Ernst Nordström would like to thank Jakob Carlström for stimulating discussions. Gábor Fodor was financially supported by Ericsson and Technical University of Budapest. Gábor Fodor acknowledges the encouragement from Miklós Boda, Ericsson and Tamás Henk, High Speed Network Laboratory, Technical University of Budapest.

References

[1] S. Blaabjerg and G. Fodor, "A Generalization of the Multirate Circuit Switched Loss Model to Model ABR Services in ATM Networks", in *Proc. of the IEEE International Conference on Communications*, Singapore, Nov. 1996.

[2] S. Blaabjerg, G. Fodor and A. Andersen, "A Partially Blocking-Queueing System with CBR/VBR and ABR/UBR Arrival Streams", in *Proc. of 5th International Conference on Telecommunications Systems*, Nashville, USA, Mar. 1997.

[3] B. Kraimeche and M. Schwartz, "Bandwidth Allocation Strategies in Wide-Band Integrated Networks", *IEEE Journal on Selected areas in Commun.*, vol. SAC-4, no. 6, pp. 869-878, Sep. 1986.

[4] Z. Dziong and L. Mason, "Call Admission and Routing in Multi-Service Loss Networks", *IEEE Trans. on Commun.*, vol. 42, no. 2, pp. 2011-2022, Feb. 1994.

[5] Z. Dziong, K. Liao and L. Mason, "Flow Control Models for Multi-Service Networks with Delayed Call Set Up", in *proc. of INFOCOM'90,* pp. 39-46, San Francisco, USA, Jun. 1990.

[6] Z. Dziong and L. Mason, "Fair-Effficent Call Admission Control Policies For Broadband Networks - A Game Theoretic Framework", *IEEE/ACM Trans. on Networking.*, vol. 4, no. 1, pp. 123-136, Feb. 1996.

[7] K. Ross and D. Tsang, "Optimal Circuit Access Policies in an ISDN Environment: A Markov Decision Approach", *IEEE Trans. on Commun.*, vol. 37, no. 9, pp. 934-939, Sep. 1989.

[8] E. Nordström, "Near-Optimal Link Allocation of Blockable Narrow-Band and Queueable Wide-Band Call Traffic in ATM Networks", in *Proc. of 15th International Congress on Telecommunications, ITC'15*, Washington D.C., USA, Jun. 1997.

[9] E. Nordström and J. Carlström, "A Reinforcement Learning Scheme for Adaptive Link Allocation in ATM Networks", in *Proc. Int. Workshop on Appl. of Neural Networks to Telecommun., IWANNT'95*, pp. 88-95, Stockholm, Sweden, May 1995.

[10] H. Tijms, *Stochastic Modeling and Analysis - a Computational Approach,* Wiley, 1986.