Novel Techniques for the Design and Control of Generalized Processor Sharing Schedulers for Multiple QoS Classes

Krishnan Kumaran Geoffrey E. Margrave Debasis Mitra Keith R. Stanley
Lucent Technologies, Bell Labs Innovations
600 Mountain Avenue, Murray Hill, NJ 07974
{gmargrave,kumaran,mitra,kstanley}@lucent.com

Abstract— **Generalized Processor Sharing (GPS) is a scheduling discipline which provides minimum service guarantees as well as fair resource sharing. The performance of GPS is governed by the scheduling *weights* associated with individual connections. We address the design of the GPS weights, along with the related Connection Admission Control (CAC). Our main goal is to achieve statistical multiplexing gains in the presence of multiple traffic and Quality of Service (QoS) classes of connections that share a common trunk. We present several novel techniques to compute and adapt the weights. We also characterize the capacity region of the system, and propose a natural CAC procedure. Numerical results on 2-class and 3-class examples demonstrate the effectiveness of our methods.**

I. INTRODUCTION

The explosive growth of bandwidth in high speed communication networks is supporting the creation of a large variety of applications with diverse characteristics and service requirements. Despite the abundance of resources, resource management in the presence of such diversity can nevertheless constitute a challenge if contractual guarantees matched to the needs of the services are to be provided. In this respect, large scale per-VC (per-Virtual-Circuit) queueing in switches designs is a significant development since it allows tight control over resource allocation and usage for each connection. This paper addresses this resource management issue in the design of Generalized Processor Sharing (GPS) schedulers. GPS promises substantial sharing as well as isolation and QoS guarantees [CRU91], [PGA93], [PGA94], [KES96], [KES97], [SVA98a]. These properties of GPS follow from the proper design of *weights* assigned to individual connections. Also, GPS with output regulation has been proposed in [EMI99] to accommodate *best-effort* traffic by enforcing a *peak-rate* on guaranteed service connections, which provides no better service to the latter than just what is guaranteed.

Weight assignments in GPS are strongly driven by the traffic characteristics and QoS requirements. We restrict ourselves in this paper to the worst-case *Dual-Leaky-Bucket-Regulated* connections [EMW95], [LZT97], [RRR98], [RRR99], [EMI99], in which connections are classified by their leaky-bucket parameters. QoS is specified by the probability of violating the delay bound, where the probability and delay bound are QoS parameters associated with each connection class. Multiplexing gains are extracted by assuming that the regulated sources are independent and non-colluding.

We propose design methods for selecting weights in the GPS scheduler for an arbitrary number of QoS classes. This extends prior work on 2-class GPS [EMI99]. We characterize the capacity region of the multi-class system and, furthermore, provide a real-time algorithm for controlling the weights to maximize capacity. Of particular importance is the fact that we develop two approaches which are based on the Central Limit [FEL66] and Large Deviations [SWE95] regimes, whereas prior work in this area has mostly focussed on the latter, with few exceptions [CSH97]. These regimes give rise naturally to the Gaussian and the Chernoff asymptotic approximations. The Central Limit approach is applicable when the violation probabilities are relatively high (typically 10^{-6} and higher), whereas the latter is more suited to small probabilities (typically lower than 10^{-6}). While it is conventional for ATM systems to be designed for low violation probabilities, the Internet has shown that many applications may be adequately served with less stringent requirements. We also present a novel technique based on a reservation mechanism for the slow adaptation of weights, which may also be implemented by switching among a small set of pre-computed weights. This contrasts with the technique in [EMI99] for two classes in which switching between at most two weights is adequate. The technique that we present enforces fairness among connections with varying resource requirements, an important requirement of CAC.

Section II describes the system and states the problem, followed by section III which gives crude deterministic upper and lower bounds, followed by an exact solution for the Central Limit approximation. The approach using the Chernoff (section IV) approximation is then formulated and analyzed. Our CAC algorithm, based on a reservation mechanism combined with weight changes on a slow time scale, is described in section V, which is followed by section VI on numerical results, conclusions (section VII) and the Appendices.

II. PRELIMINARIES

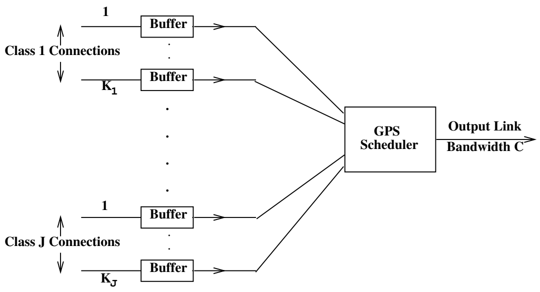

Fig. 1. Multi-class GPS Scheduler at a node.

The system investigated in this paper is shown in figure 1. Each connection is regulated by a dual leaky bucket. The typical dual leaky bucket regulator has parameters (r, P, B_T), where r is the token rate, P is the constraint on the peak rate and B_T is the token buffer size. The parameters of the regulator for each class of connections are characteristic of the class. The QoS specifications for each connection of class j, $1 \leq j \leq J$, are the delay bound D_j, and the loss probability L_j, which bounds the probability of violation of the delay bound. Thus each class is specified by the five tuple (r, P, B_T, D, L).

In [EMW95] and [EMI99] it has been shown that in the extremal regulated connection process the bandwidth demand to satisfy QoS is an on-off process taking values e_0 and 0 for time fractions a and $(1-a)$, respectively. Here e_0 is the *effective bandwidth for lossless multiplexing* and given by

$$e_0 = \max\left(r, \frac{P}{1+\frac{D}{B_T}(P-r)}\right) \tag{1}$$

The activity indicator a is given by $a = r/e_0$.

The weight given to connections of class j in the GPS scheduler is denoted by ϕ_j, $1 \leq j \leq J$. The bandwidth allotted to such a connection at any time instant by th scheduler is

$$\frac{\phi_j C}{\sum_l \phi_l \sum_{i=1}^{K_l} \xi_i^l} \tag{2}$$

where ξ_i^j is the binary activity indicator variable for connection i in class j.

In considering statistical QoS, the extremal regulated connection processes have independent random phases, as a consequence of which the activity indicators ξ_i^j are independent, binomial random variables,

$$Pr(\xi_i^j = 1) = 1 - Pr(\xi_i^j = 0) = a_j. \tag{3}$$

Now consider a tagged connection which is active, and assume without loss of generality that it is connection 1 of class k. This tagged source will satisfy its QoS requirement if it receives at least bandwidth e_0^k with probability not less than $(1-L_k)$, i.e., from (2),

$$Pr\left[\frac{\phi_k C}{\phi_k + \phi_k \sum_{i=2}^{K_k} \xi_i^k + \sum_{j\neq k} \phi_j \sum_{i=1}^{K_j} \xi_i^j} \geq e_k^0\right] \geq 1-L_k.$$

Equivalently,

$$\begin{aligned} L_k &\geq Pr\left[\phi_k + \phi_k \sum_{i=2}^{K_k} \xi_i^k + \sum_{j\neq k} \phi_j \sum_{i=1}^{K_j} \xi_i^j \geq \phi_k C/e_0^k\right] \\ &> Pr\left[\sum_{j=1}^{J} \phi_j \sum_{i=1}^{K_j} \xi_i^j \geq \phi_k C/e_0^k\right]. \end{aligned} \tag{4}$$

Hence statistical QoS is satisfied for all extremal regulated connection processes if, for all k $(1 \leq k \leq J)$,

$$-\ln Pr\left[\sum_{j=1}^{J} \phi_j \sum_{i=1}^{K_j} \xi_i^j \geq \phi_k C/e_0^k\right] \geq -\ln L_k. \tag{5}$$

Equation (5) gives the conditions to be satisfied for the set of connections given by $\mathbf{K} = \{K_j\}$ to meet their QoS requirements when the scheduler weights are $\{\phi_j\}$. Note from (2) that scheduling is not affected by any constant of proportionality applied to all the weights.

Of fundamental importance is the realizable set, or region, $\mathcal{R}$ which is defined as the set of connection population vectors $\mathbf{K}$ for which there exist weights $\{\phi_j\}$ such that (5) is satisfied. In investigating $\mathcal{R}$ in the f ollowing sections, we will find it convenient to parametrize $\mathbf{K}$ thus.

$$\mathbf{K} = K(\eta_1, \eta_2, ..., \eta_J), \quad \sum_j \eta_j = 1. \tag{6}$$

The vector η denotes the traffic mix, and, for given η, the goal is to find the largest value of K that is realizable.

III. CALCULATION OF THE REALIZABLE REGION: GAUSSIAN APPROXIMATION

Below we explore approximations and bounds for characterizing the realizable region.

A. Mean Value Approximation

A simple and conservative bound for the realizable region is obtained by assuming all connections are active all the time:

$$\sum_j K_j e_0^j \leq C; \quad \phi_j = \frac{e_0^j}{\sum_{k=1}^J e_0^k} \quad (1 \leq j \leq J). \quad (7)$$

At the other extreme, a rough optimistic estimate of the realizable region is obtained by using mean values of the activity indicators,

$$\mathcal{E}[\sum_{i=1}^{K_j} \xi_i^j] = K_j a_j.$$

Using mean values for the condition invoked in the RHS of (4) gives

$$\sum_{j=1}^J \phi_j K_j a_j \leq \phi_k C / e_0^k \quad (1 \leq j \leq J). \quad (8)$$

It is straightforward to show that the corresponding realizable region is a simplex:

$$\sum_{j=1}^J K_j a_j e_0^j \leq C \quad (9)$$

with the weights $\phi_j = e_0^j$. This is a crude and optimistic bound on the realizable region.

B. The Central Limit Approximation

Invoking the Central Limit Theorem for large $\{K_j\}$,

$$\sum_{i=1}^{K_j} \xi_i^j \to K_j a_j + \sqrt{K_j a_j (1 - a_j)} \; U_j \quad (10)$$

where $U_j = \mathcal{N}(0,1)$ is a zero mean, unit variance Gaussian random variable. From (10) and 5), for $k = 1, .., J$

$$-\ln Pr \left[\sum_j \phi_j \sqrt{K_j a_j (1 - a_j)} \; U > \frac{\phi_k C}{e_0^k} - \sum_j \phi_j K_j a_j \right]$$

$$\geq - \ln L_k. \quad (11)$$

where $U = \mathcal{N}(0,1)$ and satisfies the following equality in distributions:

$$\sum_j \phi_j \sqrt{K_j a_j (1 - a_j)} \; U_j = \sqrt{\sum_j \phi_j^2 K_j a_j (1 - a_j)} \; U. \quad (12)$$

Now let us introduce the mean and variance of the extremal regulated connection processes (see section II):

$$m_j = e_0^j a_j, \quad v_j = (e_0^j)^2 \, a_j \, (1 - a_j). \quad (13)$$

Also let

$$\theta_j = \frac{\phi_j}{e_0^j} \frac{1}{\sum_k (\phi_k / e_0^k) K_k m_k} \quad (1 \leq j \leq J). \quad (14)$$

Then, from (11), for $k = 1, .., J$,

$$-\ln Pr \left[U \geq \frac{\theta_k C - 1}{\sqrt{\sum_j \theta_j^2 K_j v_j}} \right] \geq - \ln L_k. \quad (15)$$

From standard bounds [FEL66] on the tail behavior of the Gaussian distribution, we obtain from (15),

$$\frac{\theta_k C - 1}{\sqrt{\sum_j \theta_j^2 K_j v_j}} \geq \delta_k \quad (1 \leq k \leq J). \quad (16)$$

Here $\delta_k = \sqrt{2} Erfc^{-1}(2L_k) \approx \sqrt{-2 \ln L_k - \ln 2\pi}$ is a close, conservative approximation which is also asymptotically exact for $L_k \to 0$.

Equation (16) is solved by the procedure described in Appendix I to give

$$\phi_k \propto \theta_k e_0^k = \frac{e_0^k}{C} (1 + \delta_k \Delta), \quad (17)$$

where Δ is the unique positive solution of the following cubic equation:

$$\frac{\Delta^2 C^2}{\sum_j (1 + \delta_j \Delta)^2 K_j v_j} = \frac{C}{\sum_j (1 + \delta_j \Delta) K_j m_j}. \quad (18)$$

This is our main result for the approach based on the Gaussian approximation. Note that the parameter Δ depends on the connection populations $\{K_j\}$, thus making the weights similarly dependent, see (17).

The realizable region $\mathcal{R}$ of connection population vectors $\mathbf{K}$ is given by

$$\sum_j K_j a_j e_0^j (1 + \delta_j \Delta) \leq C. \quad (19)$$

which may be contrasted to (9). Since Δ depends on $\{K_j\}$, see (18), the left hand side of (19) is non-linear in $\{K_j\}$. Inevitably, the design of the weights and the implementation of CAC is somewhat complicated. Note however that the effect of the connection population is entirely encapsulated in the single parameter Δ. Hence the connection population may be periodically monitored, from which Δ may be calculated and the weights returned.

IV. Computing the Realizable Region: The Chernoff Approximation

A basic bound that holds for any random variable with well-defined moment-generating function (see for example [BUC90]) is

$$-\ln Pr[x \geq C] \geq Cs - F(s) \tag{20}$$

for all $s \geq 0$, where $F(s)$ is the log-moment-generating-function of the random variable x, defined as

$$F(s) = \ln \mathcal{E}[e^{sx}]$$

with $\mathcal{E}[\,]$ denoting expected value. In particular the tightest value of this bound is obtained thus

$$-\ln Pr[x \geq C] \geq \max_s \; Cs - F(s). \tag{21}$$

Equation (21) is the Chernoff bound, which is also asymptotically tight when x is the sum of a large number of independent random variables, where the number, as a proportion of C, is bounded as $C \to \infty$. These conditions apply in this case. Defining $M_k = C/e_0^k$ and $\delta_k = -\ln L_k$, we apply the Chernoff bound to the probabilities in (5) thus:

$$-\ln Pr[\sum_j \phi_j \sum_{i=1}^{K_j} \xi_i^j \geq \phi_k M_k] \geq$$

$$M_k s_k \phi_k - \sum_j K_j F_j(s_k \phi_j) \geq \delta_k. \quad (k = 1,..,J) \tag{22}$$

where

$$F_j(u) = \ln[1 - a_j + a_j e^u] \tag{23}$$

is the log-moment-generating-function of the binary activity indicators $\xi_\cdot^j$ We then see that, in a given direction η, see (6), we can rewrite (22) as

$$K \leq \frac{M_k s_k \phi_k - \delta_k}{\sum_j \eta_j F_j(s_k \phi_j)} \quad (k = 1,..,J) \tag{24}$$

for some $s_k \geq 0$, for every k. We thus maximize the realizable region along the given direction vector η by solving

$$K^{max}(\eta) = \max_\phi \min_k \max_{s_k} \frac{M_k s_k \phi_k - \delta_k}{\sum_j \eta_j F_j(s_k \phi_j)} \tag{25}$$

Equation (25) thus gives the values of the weights $\{\phi_k\}$ that yield the largest realizable region for given traffic mix η. The boundary of the realizable region is then given by $\mathbf{K} = \eta K^{max}(\eta)$ as η is varied over the positive orthant.

A. *Approximating the Realizable Region*

Equation (25) is difficult to solve analytically, and also numerically. In the following, we focus on approximate solutions which realize subsets of $\mathcal{R}$ together with the GPS weights.

In (24), we make the selection $s_1 = s_2 = ... = s_J = s$. Furthermore, note from (24) that there is no loss of generality in assuming $s = 1$, since a constant of proportionality applied to $\{\phi_j\}$ does not affect the solution. This modifies (24) to

$$K \leq \frac{M_k \phi_k - \delta_k}{\sum_j \eta_j F_j(\phi_j)} \qquad (k = 1,..,J) \tag{26}$$

which leads to the reduced problem

$$K^{max}(\eta) \geq \max_\phi \min_k K_k^{approx}(\phi, \eta) \tag{27}$$

where

$$K_k^{approx}(\phi, \eta) = \frac{M_k \phi_k - \delta_k}{\sum_j \eta_j F_j(\phi_j)} \tag{28}$$

It is easily shown using Lemma 1 in Appendix I that, for a given η, the maximizing weight vector in this reduced problem satisfies

$$K_k^{approx}(\phi, \eta) = \Omega \qquad (k = 1,..,J) \tag{29}$$

for some class-independent global parameter Ω. This then implies, from (28),

$$M_k \phi_k - \delta_k = \Delta \qquad (k = 1,..,J) \tag{30}$$

where Δ is yet another class independent parameter. We then obtain the solution

$$\phi_k = \frac{1}{M_k}(\delta_k + \Delta) \qquad (k = 1,..,J) \tag{31}$$

with Δ given by

$$\Delta = \arg\max_{D \geq 0} \frac{D}{\sum_j \eta_j F_j[\frac{1}{M_j}(\delta_j + D)]} \tag{32}$$

Equations (31) and (32) provide a simple approximate solution to the problem stated in (25). The maximization in (32), which is the key step in the approximation, is 1-dimensional, with a unique solution that follows from convexity. Thus the computation of Δ is straightforward.

It is interesting to compare the expressions for the weights $\{\phi_k\}$ in the Gaussian and Chernoff approximations as given by (17) and (31) respectively.

The performance of the approximate solutions obtained from (31) and (32) have been tested in several non-homogeneous examples. The results, which are reported in Section VI, show that the realizable region thus obtained closely approximates the optimal in all the examples.

In Appendix II, we give an iterative technique for solving (25), which is a refinement to the solution given by (31) and (32). The key distinguishing feature to this refinement is that the parameters $\{s_k\}$ in (25) are not required to be identical.

V. Adaptive Control of Weights and Call Admission

We now discuss algorithms to perform weight adaptation and Connection Admission Control (CAC) in real time. We assume that connections are not queued, i.e., connections are accepted or rejected immediately upon arrival. The algorithms depend strongly on the characterization of the realizable region derived above, and can be applied to either of the two approaches discussed above, namely, the Gaussian and Chernoff approximations. Here we restrict ourselves to the Chernoff-based approximate solution. A very similar procedure applies for the Gaussian approach.

We first consider a straightforward, greedy procedure that operates on the philosophy of avoiding weight changes unless necessary. (System solutions require that the frequency of weight changes is small.) Here "necessary" is to be interpreted in the sense that without a weight change the incoming connection cannot be accommodated.

In the following realization we exploit the fact that the realizable region is characterized by a single parameter Δ, which also determines all the weights. Hence we only need to adapt Δ, as opposed to the N-dimensional weight vector ϕ, which is a convenient feature for implementation. On substituting (31) in (26), we obtain the following compact characterization of the approximate realizable region for fixed weights $\{\phi_j\}$:

$$\sum_j K_j F_j(\phi_j) \leq \Delta \tag{33}$$

(In the case of the Gaussian approximation, the corresponding characterization is in (19).

Algorithm N-CAC

• Given the initial traffic vector $\mathbf{K}$, start with optimal Δ and ϕ.

• At each decision instant (connection arrival of class k, say), perform the following steps:

– If $\sum_j K_j F_j(\phi_j) + F_k(\phi_k) \leq \Delta$, admit the connection with weight ϕ_k.

– Else, solve for Δ^{new} and ϕ^{new} and using (31) and (32), and K^{max}

– If #(connections) +1 $\leq K^{max}$ admit the connection and reset ϕ to ϕ^{new}.

– Otherwise reject the connection.

Notice that this algorithm, named **N-CAC** to denote that the weight changes are necessity based, is very economical in terms of weight changes if the system operates within the realizable region. The number of weight changes may, however, increase rapidly as the system gets heavily loaded and the boundary of the realizable region is approached. In a practical implementation, further computational efficiency and reduction in weight changes may be obtained by imposing a threshold on changes in direction of vector η before re-computing weights. The algorithm as presented above is "optimal" in the sense that it makes the minimum possible number of weight changes while realizing the entire realizable region during operation.

We now discuss a new topic, namely, the connection blocking performance of the algorithm **N-CAC**. The connection blocking performance of the algorithm may be obtained from standard traffic models in which the state-space is the realizable region. For Poisson traffic, with blocked connections cleared, the solution to this model is *product-form* (see [COO81]), i.e. the blocking probability B_j for connections of class j is given by

$$B_j = \frac{\sum_{\mathbf{K}\in\mathcal{B}_j} \prod_m \frac{(\lambda_m/\mu_m)^{K_m}}{K_m!}}{\sum_{\mathbf{K}\in\mathcal{S}} \prod_m \frac{(\lambda_m/\mu_m)^{K_m}}{K_m!}} \tag{34}$$

where λ_m is the offered Poisson rate for connections of class m, $\frac{1}{\mu_m}$ is the mean holding time of such connections, $\mathcal{B}_j$ is the set of blocking states for connections of class j and $\mathcal{S}$ denotes the realizable region.

However, the algorithm has several drawbacks. First, importantly, as we will demonstrate later in our numerical results, the method is unfair towards connections requiring greater resources in terms of longer holding times or greater bandwidth. These connections are quickly "squeezed out" by the connections requiring less resources when the system is heavily loaded. Secondly, the weight changes are performed in a greedy manner, with no attention given to meeting class-dependent blocking targets. Thirdly, the algorithm sometimes exhibits flapping by reacting strongly to temporary traffic fluctuations.

We propose a new algorithm with superior performance, especially with regard to fairness. Here we suppose that we are given a "desirable" connection mix vector η^* is to be targeted when the system is heavily loaded. This vector may, for instance, be obtained by considerations of revenue maximization combined with a notion of fairness, such as meeting maximum blocking requirements for each class, for specified offered traffic. The algorithm is effectively insensitive to η^* when the system is lightly loaded, while gracefully enforcing the mix η^* as the load becomes heavier. Such mechanisms have been proposed before in other contexts using the method of "Virtual Partitioning" (VP) described in [MZI96], [BMI97], [KMI98] and references therein.

In correspondence with other VP based algorithms, our new algorithm is driven by "nominal allocations" to each connection class, and "reservations" against classes that exceed their nominal allocations. The algorithm then performs the following steps to admit an arriving connection of class k.

Algorithm VP-CAC

- For the current traffic mix η in the system, determine all classes $\mathcal{U} \equiv \{l : \eta_j < \eta_j^*\}$, i.e., the set of "underloaded" classes.
- If $k \in \mathcal{U}$, admit the connection if possible, with a weight change if necessary.
- If $k \notin \mathcal{U}$
  – Check if the offered connection can be admitted simultaneously with a connection of class $l \in \mathcal{U}$, with a weight change if needed. Repeat this computation for each $l \in \mathcal{U}$ in turn.
  – Admit the offered connection if the above step returns an affirmative answer. Reject otherwise.

As stated above, the algorithm reserves a capacity corresponding to that required to admit a single connection of any underloaded class in the system. Reservation for more than a single such call, for purposes of increased protection, may also be considered. However, as we will see later, it is typically unnecessary since reservation for a single connection enforces fairness effectively. As is to be expected, all the properties associated with robust and efficient resource sharing described in previous work on VP apply here. As intended, the algorithm achieves the goal of unfettered sharing at light loads and drifts towards the intended traffic mix η^* at heavy loads. Section VI gives simulation results to demonstrate these advantages.

VI. Experimental Results

The results are in two parts: the first part is on the realizable regions obtained by the Central Limit and Chernoff approaches, and the second part is on the simulated real-time behavior of the weight adaptation and CAC algorithms.

A. Computing the Realizable Region

We present numerical results on the realizable region for simple, diverse examples with two and three classes. The examples use combinations of source classes described in Table I. We consider the following 2-class examples:

- Case 1: Source classes 1 & 2.
- Case 2: Source classes 3 & 4.
- Case 3: Source classes 5 & 6.

The 3-class example has source classes 4, 5 and 6. The source characteristics of class 5 correspond to voice, while the remaining are for various classes of data.

We examine the realizable region for the following cases.

- No statistical multiplexing, i.e. $\sum_j K_j e_0^j \leq C$.
- The Mean-Value approximation, i.e. $\sum_j K_j a_j e_0^j \leq C$.
- The Central Limit approximation.
- The Chernoff approximation.
- The refined Chernoff approximation.

For each 2-class case, we plot the following:

- The realizable region (see Figure 2a.). The regions obtained from the Central Limit and Chernoff approximations are quite close. The mean-value approximation is too optimistic, as expected. The region for no statistical multiplexing is small and emphasizes the capacity gains from statistical multiplexing.
- The ratio of weights ϕ_1/ϕ_2 as a function of traffic mix (Figure 2b.) for the Central Limit and Chernoff-bound methods. This ratio is also in close agreement.

In the 3-class case, we only plot the realizable region, which has a boundary that is roughly linear (a diagonal plane through the cube) in this example (see Figure 4). This suggests that this example is analagous to the "effectively homogeneous" case for two classes that is discussed in [EMI99].

For comparison, we also plot the exact and approximate regions from the Chernoff bound method discussed in Section IV and Appendix II (see Figure 3). This figure also compares similar results obtained using the refined Chernoff approximation. Notice that the approximate region differs from the exact only slightly, and the iterative algorithm (**ITER**) gives regions very close to the optimal.

Source Class	e	$a = r/e$	L_k
1	1.48	0.1	10^{-3}
2	0.158	0.95	10^{-9}
3	0.62	0.24	10^{-3}
4	1.31	0.11	10^{-9}
5 (Voice)	0.0318	0.5	10^{-3}
6	0.48	0.31	10^{-9}

TABLE I

Traffic and QoS characteristics of the 6 connection classes used in the experiments.

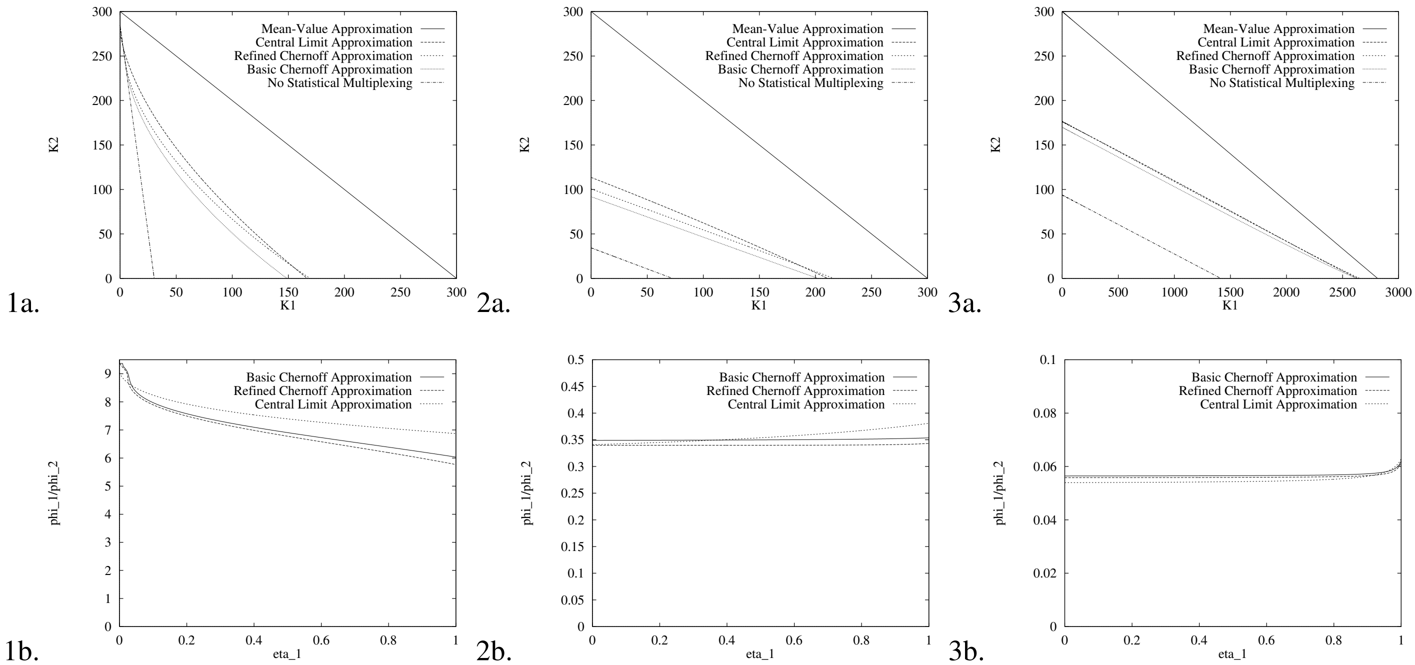

Fig. 2. Case 1, 2 and 3: a - Realizable regions. b - Ratio of weights ϕ_1/ϕ_2.

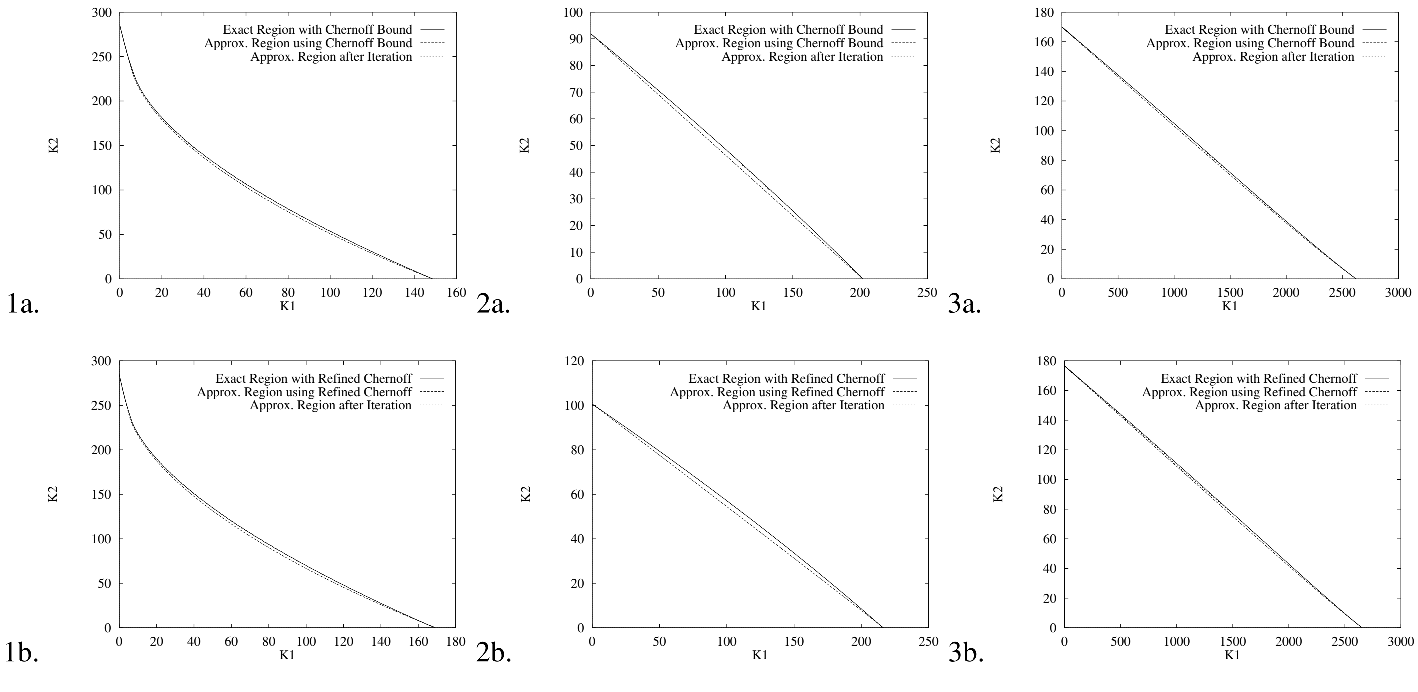

Fig. 3. Comparison of regions using exact, approximate and iterative methods: Case 1, 2 and 3, a - Chernoff Bound. b - Refined Chernoff approximation.

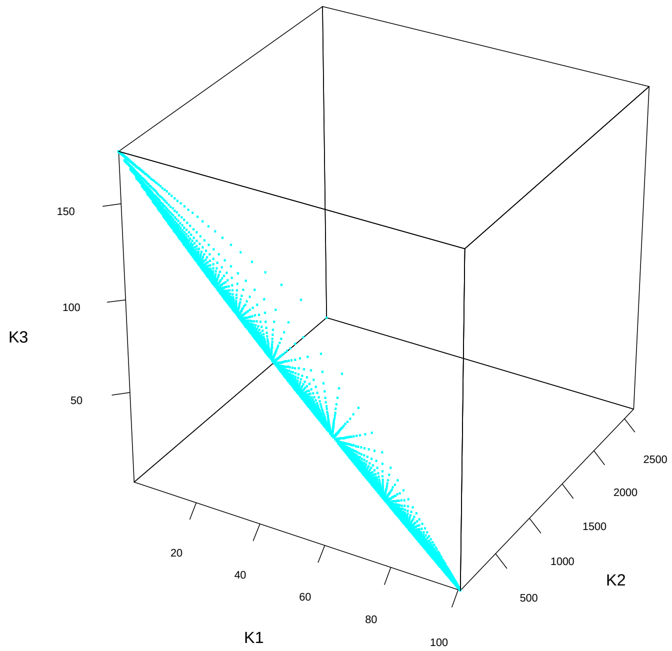

Fig. 4. 3-class Case: Realizable region.

B. Simulations of the Weight Adaptation and CAC Algorithms

We now present simulation results for our real-time control algorithms - namely **N-CAC** and **VP-CAC** (see section V). The simulations were run for 2 different QoS classes (class 1 and class 2) using Poisson connection traffic, i.e., Poisson connection arrivals and exponentially distributed holding times. The arrival rates λ_1, λ_2 were scaled

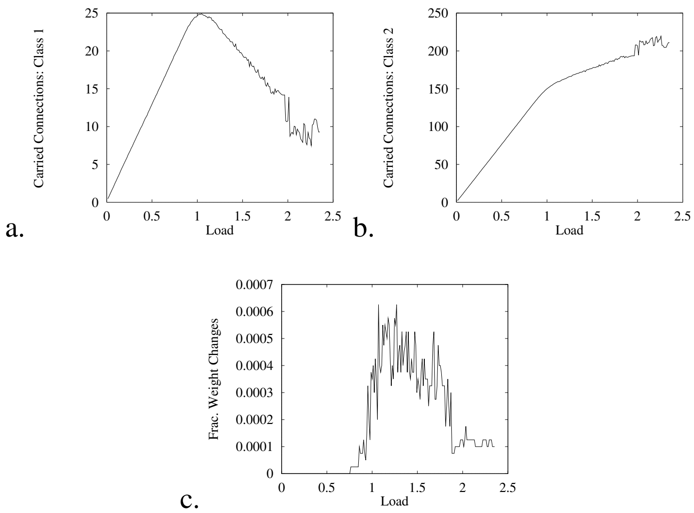

Fig. 5. N-CAC: a. and b. Carried connections as a function of system load for Class 1 and Class 2. c. Fraction of connections requiring weight change.

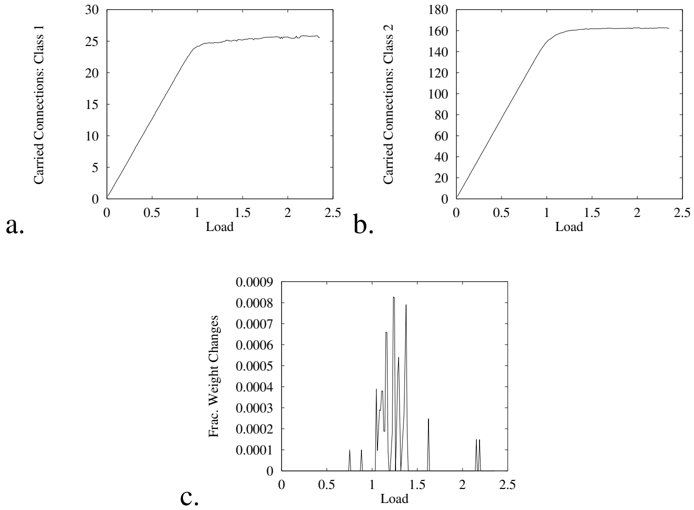

Fig. 6. VP-CAC: a. and b. Carried connections as a function of system load for Class 1 and Class 2. c. Fraction of connections requiring weight change.

in the proportion 2:1 and the holding times $\frac{1}{\mu_1}, \frac{1}{\mu_2}$ were fixed at a 3:1 ratio. We defined a nominal load measure

$$\frac{1}{C}\sum_{k=1}^{2}\frac{\lambda_k}{\mu_k}e_0^k a_k$$

with respect to which we plot all our performance measures. In the figures we show (i) the carried load for each class, and (ii) the number of weight changes required by the algorithm in steady state.

In the case of **VP-CAC**, the desired mix η^* was determined using the ratio of arrivals to departures for each QOS class. For each data point, the simulation was run long enough to eliminate the initial transient behavior, after which the statistics were collected.

The first set of plots, Figure 5 a. and b. show the carried load vs. the offered load for each class. As can be seen, as the offered load increased, for the **N-CAC** algorithm, the smaller bandwidth connections dominate the system, while the **VP-CAC** algorithm (with only a single connection of reservation) is able to keep the traffic mix stable.

Figure 5 c. shows the frequency of weight changes vs. offered load. This plot shows that the weight changes increase for **N-CAC** as the traffic increases. The VP-CAC (see Figure 6 c.), in contrast, only causes substantial weight changes at the boundary, as the system approaches overload. At very high loads, VP-CAC drives the system towards the desired traffic mix and the corresponding weights, thus providing greater stability in addition to the fairness advantage.

In addition to the above, our simulations provided insight into the transient behavior of VP-CAC. In a typical implementation, the choice of desired traffic mix may be determined by forecast of offered traffic. Periodically, a new desired mix is computed to correspond to expected changes in traffic. The GPS weights cannot be instantaneously changed to the weights corresponding to the new desired mix. The transient behavior of *VP-CAC* to changes in the targeted mix is surprisingly smooth and rapid, requiring few weight changes.

VII. Conclusions

We have presented techniques for computing the weights and the realizable region, together with CAC, in a GPS switch, which take advantage of statistical multiplexing. Key contributions include techniques to deal with arbitrary numbers of connection classes, and reservation-based weight and admission control techniques for GPS which achieve fairness among connection classes. Future work will address multiplexing in buffer management. The use of measurements for the calculation of weights and for the CAC algorithm is also a direction for future study.

References

[BMI97] S. Borst and D. Mitra. Virtual Partitioning for Resource Sharing by State-Dependent Priorities: Analysis, Approximations and Performance for Heterogeneous Traffic. In V. Ramaswami and P. E. Wirth, editors, *Teletraffic Contributions for the Information Age, Proc. ITC-15*, pages 1457–1468. Elsevier, 1997.

[BUC90] J. A. Bucklew. *Large Deviation Techniques in Decision, Simulation and Estimation*. Wiley-Interscience, New York, 1990.

[CSH97] J. Choe and N. B. Shroff. A Central Limit Theorem Based Approach to Analyze Queue Behavior in ATM Networks. In V. Ramaswami and P. E. Wirth, editors, *Teletraffic Contributions for the Information Age, Proc. ITC-15*, pages 1129–1138. Elsevier, 1997.

[COO81] R. B. Cooper. *Introduction to Queueing Theory*. North Holland, New York, 1981.

[CRU91] R. L. Cruz. A Calculus for Network Delay, Part I: Network Elements in Isolation. *IEEE Trans. on Information Theory*, 37:114–131, 1991.

[EMI99] A. Elwalid and D. Mitra. Design of Generalized Processor Sharing Schedulers Which Statistically Multiplex Heterogeneous QoS Classes. In *Proc. IEEE INFOCOM'99*, pages 1220–1230, 1999.

[EMW95] A. Elwalid, D. Mitra, and R. H. Wentworth. A New Approach for Allocating Buffers and Bandwidth to Heterogenous, Regulated Traffic in an ATM Node. *IEEE Journal on Selected Areas in Communications*, 13(6):1115–1127, Aug. 1995.

[FEL66] W. Feller. *An Introduction to Probability Theory and Its Applications*, volume II. John Wiley & Sons, New York, 1966.

[KES97] S. Keshav. *An Engineering Approach to Computer Networking*. Addison Wesley, Reading, MA, 1997.

[KES96] G. Kesidis. *ATM Network Performance*. Kluwer, Boston, MA, 1996.

[KMI98] K. Kumaran and D. Mitra. Performance and Fluid Simulations of a Novel Shared Buffer Management System. In *Proc. IEEE INFOCOM'98*, pages 1449–1461, 1998.

[LZT97] F. LoPresti, Z. Zhang, D. Towsley, and J. Kurose. Source Time Scale Optimal Buffer/Bandwidth Trade-off for Regulated Traffic in an ATM Node. In *Proc. IEEE INFOCOM'97*, pages 675–682, 1997.

[MZI96] D. Mitra and I. Ziedins. Virtual Partitioning by Dynamic Priorities: Fair and Efficient Resource Sharing by Several Services. In B. Plattner, editor, *Broadband Communications, Proceedings of the International Zurich Seminar on Digital Communications*, pages 173–185. Springer, 1996.

[PGA93] A. K. Parekh and R. G. Gallager. A Generalized Processor Sharing Approach to Flow Control - The Single Node Case. *IEEE/ACM Trans. Networking*, pages 344–357, June 1993.

[PGA94] A. K. Parekh and R. G. Gallager. A Generalized Processor Sharing Approach to Flow Control - The Multiple Node Case. *IEEE/ACM Trans. Networking*, pages 137–150, Apr. 1994.

[RRR98] S. Rajagopal, M. Reisslein, and K. W. Ross. Packet Multiplexers with Adversarial Regulated Traffic. In *Proc. IEEE INFOCOM'98*, pages 347–355, 1998.

[RRR99] M. Reisslein, K. W. Ross and S. Rajagopal. A Pragmatic Approach for Guaranteeing QoS to Regulated Traffic in Networks. In *Proc. IEEE INFOCOM'99*, pages 1061-1072, 1999.

[SWE95] A. Shwartz and A. Weiss. *Large Deviations for Performance Analysis: Queues, Communication and Computing*. Chapman and Hall, London, 1995.

[SVA98a] D. Stiliadis and A. Varma. Rate Proportional Servers: A Design Methodology for Fair Queueing Algorithms. *IEEE/ACM Trans. Networking*, 6:164–174, Apr. 1998.

Appendix I

We begin with (16) and introduce the traffic mix η, see (6), to obtain for $k = 1, 2, .., J$,

$$K \le \frac{\theta_k C - 1}{\delta_k^2 \sqrt{\sum_j \theta_j^2 \eta_j v_j}} \tag{35}$$

We are interested in K^{max}, the largest value of K satisfying (35). We pose the problem thus:

$$K^{max} = \max_{\theta} \min_{k} \frac{\theta_k C - 1}{\delta_k^2 \sqrt{\sum_j \theta_j^2 \eta_j v_j}} \tag{36}$$

We now appeal to the following lemma, which has a simple proof that we omit.

Lemma 1: Given N real-valued, continuous and differentiable functions $\{f_k(\mathbf{x}) : \ k \in \{1, 2, .., N\}\}$ on $\mathbf{x} \in \mathcal{R}^N$ satisfying

$$\frac{\partial f_k}{\partial x_k} > 0, \quad \frac{\partial f_k}{\partial x_l} < 0 \ for \ l \ne k \tag{37}$$

the optimal solution to

$$\mathcal{O} = \max_{\mathbf{x}} \min_{k} f_k(\mathbf{x})$$

satisfies

$$f_k(\mathbf{x}^*) = \mathcal{O}$$

for all $k \in \{1, 2, .., N\}$ where $\mathbf{x}^*$ is the optimizing vector.

Turning to (36) we see that the solution must satisfy

$$\frac{\theta_k C - 1}{\delta_k^2} = \Delta^2, \quad a \ constant, \tag{38}$$

for $k = 1, 2, .., J$. Hence

$$K^{max} = \frac{\Delta^2}{\sum_j \theta_j^2 \eta_j v_j} = \frac{\Delta^2 C^2}{\sum_j (1+\delta_j \Delta)^2 \eta_j v_j} \tag{39}$$

from (38).

Now, on introducing the traffic mix η in (14),

$$\theta_j = \frac{1}{K} \frac{\phi_j / e_0^j}{\sum_k (\phi_k / e_0^k) \eta_k m_k} \tag{40}$$

Hence,

$$\sum_j \theta_j \eta_j m_j = \frac{1}{K}. \tag{41}$$

From the above and (38),

$$K^{max} = \frac{C}{\sum_j (1+\delta_j \Delta) \eta_j m_j}. \tag{42}$$

From (39) and (42),

$$K^{max} = \frac{\Delta^2 C^2}{\sum_j (1+\delta_j \Delta)^2 \eta_j v_j} = \frac{C}{\sum_j (1+\delta_j \Delta) \eta_j m_j}. \tag{43}$$

This gives (18), while (17) is obtained from (38). It is straightforward to show that the above cubic equation in Δ has a unique positive solution. Finally, (19) follows from (42).

APPENDIX II

We propose an iterative technique for solving (25) based on the Chernoff approximation, which is a refinement to the solution given by (31) and (32). As mentioned before, the key distinguishing feature to this refinements that the parameters $\{s_k\}$ in (25) are not required to be identical. We begin by defining quantities which are in (24) and (25): for $k = 1, 2, ..., J$

$$G_k(\eta, \phi, s_k) \equiv \frac{M_k s_k \phi_k - \delta_k}{\sum_j \eta_j F_j(s_k \phi_j)} \tag{44}$$

$$H_k(\eta, \phi) \equiv \max_{s_k} \frac{M_k s_k \phi_k - \delta_k}{\sum_j \eta_j F_j(s_k \phi_j)}. \tag{45}$$

The following two properties of these functions straightforward to prove.

Lemma 2: The functions H_k are uniquely defined for any reasonable set of log-moment-generating functions $\{F_j\}$ and fixed positive values of all the parameters, i.e. there exists a unique s_k^* that solves the maximization problem in (45).

Lemma 3: For fixed η, the function H_k is an increasing function of ϕ_k and a decreasing function of $\{\phi_j : j \neq k\}$.

We now invoke Lemma 1 to state that the optimal solution to (25) satisfies

$$H_k(\eta, \phi^*) \equiv \frac{M_k s_k^* \phi_k^* - \delta_k}{\sum_j \eta_j F_j(s_k^* \phi_j^*)} = \Omega \tag{46}$$

for all classes k, with the superscript * referring to the optimizing values as before.

These properties of the optimal solution are exploited in the following iterative algorithm to solve for the optimizing Ω in (46). The algorithm on termination gives Ω, $\{\phi_k^*\}$, $\{s_k^*\}$.

Algorithm ITER

1. Start with initial feasible solution $\phi^{(0)}$, obtained from (31) and (32), and

$$s_k^{(0)} = \arg\max_s G_k(\eta, \phi^{(0)}, s).$$

2. At each step, labelled by n, determine

$$k = \arg\max_j H_j(\eta, \phi^{(n)}) = \arg\max_j G_j(\eta, \phi^{(n)}, s_j^{(n)}).$$

3. Update as

$$\phi^{(n+1)} = \phi^{(n)} - t\mathbf{e}_k$$

where

$$t = \arg\max_{\tau \geq 0} \min_k G_k(\eta, \phi^{(n)} - \tau \mathbf{e}_k, s_k)$$

where $\mathbf{e}_k$ is the unit vector with kth component set to 1.

4. Set

$$s_k^{(n+1)} = \arg\max_s G_k(\eta, \phi^{(n+1)}, s).$$

5. Stop when the H_k are all equal.

Note that **ITER** is guaranteed to converge at least to a local optimum, since the objective always increases at every step. The word "local" is due to the fact that we have not established uniqueness of the weights at the boundary of the realizable region, which would have guaranteed global optimality. We conjecture that this property holds in most realistic examples.

A final step which we have incorporated in our implementation of the algorithm **ITER** is the use of a well-known refinement to the Chernoff bound (see [BUC90]), which uses the second derivative of the log-moment-generating-functions. The results from this implementation are reported in Figure 3.