QUALITY OF SERVICE ANALYSIS OF SHARED BUFFER MANAGEMENT POLICIES COMBINED WITH GENERALIZED PROCESSOR SHARING

George Lapiotis
Applied Research
Telcordia Technologies
lapiotis@research.telcordia.com

Shivendra Panwar
Center for Advanced Technology in Telecommunications
Polytechnic University
panwar@catt.poly.edu

Abstract. An objective of the next generation network is the accommodation of services with different QoS requirements. This indicates that the network should provide special mechanisms in order to prioritize the access to network node resources, such as link capacity and buffer space. We studied the performance of sharing buffer space and link capacity between sessions, the traffic of which is modeled by independent general Markov-Modulated Fluid Process (MMFP) sources. For scheduling we use the Generalized Processor Sharing (GPS) policy, and improve previous upper bounds on the queue occupancy distributions. As an example of combining GPS with buffer management we apply our results to complete buffer sharing with virtual partitioning (VP+GPS). We also derive results on the resource allocation trade-off, with applications in traffic management and admission control.

1 Introduction

An important objective for the next generation network is the accommodation of a variety of services with different traffic characteristics and Quality of Service (QoS) requirements. This indicates that the network should provide efficient traffic management mechanisms in order to control access to resources. In this paper we consider combined shared buffer management and scheduling schemes whose objectives are to (i) effectively control the QoS of sessions that may be classes of traffic or individual connections competing for resources (bandwidth, buffer space) at network nodes, and (ii) efficiently utilize the resources.

This work was supported in part by the NY State Center for Advanced Technologies in Telecommunications (CATT), and Sumitomo Electric Industries (USA).

The work was done when G. Lapiotis was a research fellow in the NY State CATT, Polytechnic University.

There is a broad spectrum of traffic management frameworks that attempt to allocate the access to bandwidth and buffers in network nodes so as to satisfy the QoS requirements of different applications. In a connection-oriented context (e.g. ATM, RSVP subnets) there is resource reservation upon call admission. Scheduling is often used to allocate to each virtual connection its nominal bandwidth, and buffer management is used to control the loss, especially when shared buffers are used. In IP networks a recent approach to QoS provisioning is simple service differentiation achieved by flow aggregation into two or more service classes performed at ingress routers. Proposals in that direction include simple packet dropping and strict priority service or more sophisticated scheduling schemes [1]. Scheduling also plays a central role in the context of hierarchical link sharing approach and Class Based Queueing (CBQ) routers [2], in providing QoS capabilities to the Internet. A similar approach is taken in the new QoS-oriented IEEE 802.1p standard that addresses QoS issues in Local Area Networks.

It is therefore evident that both scheduling and buffer management are fundamental control tools of network designers for traffic management in network nodes, such as concentrators, switches, routers, etc. By using these controls the nodes are being designed to accommodate several types of priority classes ranging from two, by simple flow aggregation, to thousands, by managing the traffic at the connection/microflow level [3]. A first-order service categorization would distinguish between real-time or guaranteed service applications and elastic ones (see, for example, [4, 5]) with subclasses for both cases. A fundamental problem that needs to be addressed is which schemes should be used to jointly allocate the bandwidth and buffer to these traffic classes so that their QoS demands are met.

In this paper we propose and study combined

0-7803-5796-5/99/$10.00 © 1999 IEEE

scheduling and shared buffer management techniques to allocate bandwidth and buffers to sessions which share a common output link and a buffer. There are two main advantages in using such schemes. First, while scheduling directly controls bandwidth allocation, managing access to a shared buffer can guarantee that each session has the chance to enter the buffer and use its nominal share of bandwidth [3]. Second, joint buffer-bandwidth allocation is expected and shown in this paper to better utilize system resources and consequently accommodate more traffic in the network. The important issue that should then be addressed is which are the shared buffer and bandwidth resource sharing schemes that will optimize the network utilization under the constraint of meeting the individual connection or flow QoS requirements. In this work we consider that Generalized Processor Sharing (GPS) is used as the scheduling discipline across sessions destined to the same output port (a session represents a traffic stream of any granularity including aggregate flow, priority class, or connection). GPS is a work conserving scheduling discipline in the which the n input sessions share a deterministic server with total rate c. A set of parameters $\{\varphi_i\}_{1\leq i\leq n}$, called the GPS assignment, determine the share of service rate that each session receives as follows: the minimum service rate guaranteed to input session i is equal to $c_i = (\varphi_i / \sum_{j=1}^{n} \varphi_j)c$. The residual service of unbacklogged sessions is distributed to the active sessions in proportion to their weights. Also, the input traffic of session i is considered an infinitely divisible fluid that can be described by a continuous process $r_i(t)$. Thus, GPS can be considered a continuous limiting case of the Weighted Round Robin service discipline.

Most previous work on GPS analysis is mainly focused on (i) infinite or partitioned buffers per session, and (ii) very general arrival processes, with deterministic or stochastic settings. These results are general and are expected to give loose upper bounds of the distributions, because the dynamics of the arrival processes are not captured [7, 6]. This results in low utilization especially in the case of deterministic settings, as was observed in [8]. It would therefore be more preferable to use some approach that can provide statistical QoS guarantees. As an example, in [6] the sources are described by Markov Modulated Fluid Processes (MMFP), that feed an infinite buffer. A central contribution of the analysis in the present work is the development of a new analytical method for a GPS system that can be used for both infinite, and more importantly, finite queue cases. In the latter case, the analysis of complete buffer sharing with virtual partitioning [11] is integrated with GPS scheduling (VP+GPS). We initially consider the problem with two queues, and then derive the general case for multiple queues. In the latter case we improve the previous upper bound approximations of [6]. The rest of the paper is organized as follows. In section 2, we describe the method for two sessions. In section 3, the techniques are extended for multiple sessions. In section 4 we propose to jointly allocate bandwidth and buffers using VP+GPS. Section 5 includes numerical results and we conclude in section 6.

2 The Model for Two Sessions.

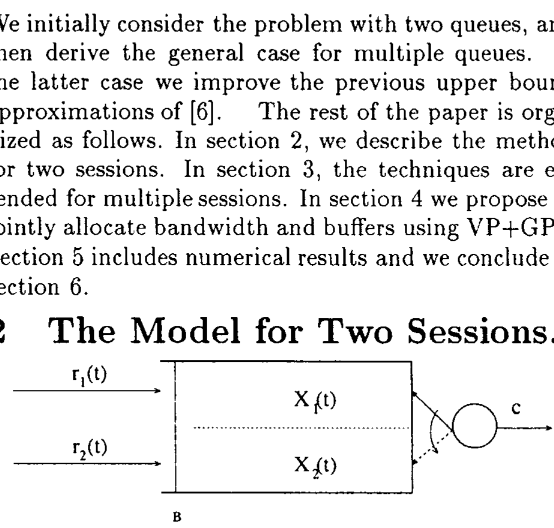

Figure 1: Bounded buffer with GPS service. Case of two MMFP source inputs.

In this section we consider two input traffic sessions described by the processes $r_1(t)$, $r_2(t)$ (see fig. 1), which are modeled by two independent general Markov Modulated Fluid Process (MMFP) sources characterized by the generator matrix $\mathbf{A}$ and diagonal rate matrix $\mathbf{\Lambda}$ for input $r_1(t)$, and the corresponding matrices $\mathbf{B}$, $\mathbf{M}$ for input $r_2(t)$. The fluid contents of the two logical queues formed are denoted by $X_1(t)$, and $X_2(t)$. In the above, the assumption is made that traffic source 1 (2) has N_1 (N_2) states. Also, λ_l is the arrival rate when the source 1 rate process is in state $S_1(t) = l$, and μ_m is the arrival rate when the source 2 rate process is in state $S_2(t) = m$. The combined total arrival rate of the two sessions can be described by a global K-state MMFP process ($K = N_1 N_2$) whose state at time t is $S(t) \stackrel{\text{def}}{=} (S_1(t), S_2(t)) \in \mathcal{G}$, where $\mathcal{G} = \{(i,j) : i = 1, ..., N_1; j = 1, 2, ..., N_2\}$. The total system has an output server of rate c. The minimum guaranteed service fraction for class 1 is a, and for class 2 is 1-a. The transmission capacity allocated to each of the two traffic streams is described by the processes $C_1(t)$ and $C_2(t)$ where

$$C_1(t) = \begin{cases} c - r_2(t), & \text{if } X_2(t) = 0 \\ c_1 = ac, & \text{if } X_2(t) > 0 \end{cases}, \tag{1}$$

and $C_2(t)$ is symmetrically defined. To proceed first define the steady state probability distribution functions, whose domain is illustrated in figure 2: $\mathrm{F}_{ij}^{W}(x_1, x_2)$ denotes the joint cdf of buffer occupancy in area W ($x_1 > 0, x_2 > 0$), at global source state $S(t) = (i, j)$. $\mathrm{F}_{ij}^{10}(x_1)$ denotes the joint cdf of buffer occupancy X_1

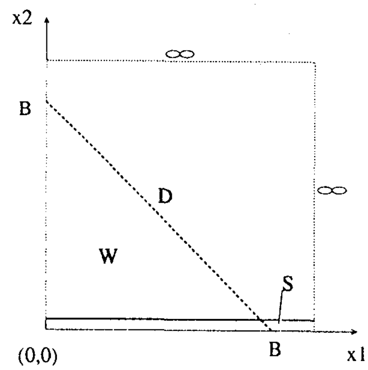

Figure 2: Derivation of system equations.

restricted on the axis $x_2 = 0$. $\mathrm{F}_{ij}^{12}(x_1)$ denotes the joint cdf of X_1 restricted in area W ($X_2 > 0$). $\mathrm{F}_{ij}^{20}(x_1)$ and $\mathrm{F}_{ij}^{21}(x_1)$ are symmetrically defined for $X_2(t)$. The exact closed-form solutions of the partial differential equation systems that describe the dynamics of the system in area W and on the x_1 and x_2 axis are very difficult to obtain because they are highly coupled [12, 13]. In the next section we describe a technique to approximate the system solution, that can be applied to both the infinite and finite queue assumption, by applying appropriate boundary conditions in each case. We summarize the development and main results. The interested reader can find details in [12, 13]. As a first step, we proceed with an approximate solution of the problem in area W, with an initial assumption that in this area the variables $X_1(t)$ and $X_2(t)$ are independent and their joint distribution can be expressed as,

$$\mathbf{F}^W(x_1, x_2) = \mathbf{F}_1^W(x_1) \otimes \mathbf{F}_2^W(x_2), \qquad (2)$$

where $\mathbf{F}_i^W(x_i)$ is the steady state vector cdf of $X_i(t)$, $i \in \{1,2\}$, and "$\otimes$" denotes the Kronecker product. This approximate joint distribution is expected to become more accurate for large B, and is only used to derive an initial estimate. In a second step of the analysis, in which the interaction between $X_1(t)$ and $X_2(t)$ is explicitely taken into account, this system is solved to obtain $\mathbf{F}^{10}(x_1)$, that is the distribution of $X_1(t)$ along the line $x_2 = 0$. Similarly, we can obtain $\mathbf{F}^{20}(x_2)$. The solution can be written as

$$\mathbf{F}_l^{10}(x_1) = \sum_{\sigma} \psi_\sigma \tau_{l\sigma} e^{r_\sigma x_1} + \mathbf{U}(x_1), \qquad (3)$$

where c_σ are constants determined by boundary conditions different for the infinite and finite buffer cases, (r, τ) are eigenpairs that typically arise when solving systems of this type, and $\mathbf{U}(x_1)$ is the term representing the excitation from area W. A similar solution stands for $\mathbf{F}^{20}(x_2)$. The constrained cdf-s $\mathbf{F}^{12}(x_1)$ and $\mathbf{F}^{21}(x_2)$ can be obtained taking into account $\mathbf{F}^{10}(x_1)$ and $\mathbf{F}^{20}(x_2)$ in boundary conditions. In a third step the individual queue cdf-s $\mathbf{F}_1(x_1)$ and $\mathbf{F}_2(x_2)$ can readily be obtained by using the simple fact that $\mathbf{F}_1(x_1) = \mathbf{F}^{10}(x_1) + \mathbf{F}^{12}(x_1)$ (similarly for $\mathbf{F}_2(x_2)$). The cdf-s can then be manipulated to extract useful performance statistics (loss, delay, etc.). We compare our results with simulations in section 5.

3 The Model for Multiple Sessions

In this section we provide an approach to treat the general case of N traffic sessions sharing a common buffer and served with GPS scheduling. As in the two-queue case the first step is to get approximate distributions for the infinite buffer case. This is equivalent to having an individual unbounded buffer for each session. For this problem we propose a technique that substantially improves the upper bound approximation presented in [6]. We then obtain approximate results when the buffer is finite by adapting the N-queue problem to a two-dimensional one with one *isolated* queue and the remaining system of $N-1$ queues treated as a single logical queue. The results of the two-queue case in the previous sections can then be used to approximately model each individual queue of the system in isolation.

As mentioned in the introduction, the infinite buffer problem has attracted considerable attention. We consider the problem in the stochastic setting, i.e., our goal is statistical QoS provisioning to individual sessions. Although deterministic upper bounds may be desirable for some guaranteed QoS services, we expect that many, if not most, applications can profit from statistical multiplexing gains and better resource utilization that stochastic bounds can provide. Initially a decomposition approach [7, 6] is adopted to derive the upper bounds for the tail distribution of individual queues for the N-queue case. We then provide a new expression for the service process of each decomposed queue that is shown by numerical results to substantially improve the upper bound approximation of the tail distribution.

Consider N general MMFP sources that are inputs to an infinite buffer system with GPS service. The corresponding GPS assignments are $\varphi_1, \varphi_2, ..., \varphi_N$, where without loss of generality we assume $\sum_{i=1}^{N} \varphi_i = 1$. The service process $s_k(t)$ of logical queue k is time-variant and at any time instant time t it can be determined by the residual service of the remaining $N-1$ queues and the GPS assignments φ_i, $i = 1, ..., k-1, k+1,$

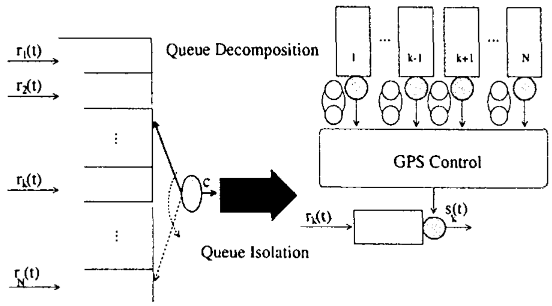

Figure 3: Queue Decomposition and Isolation in a GPS system with N queues.

...,N. In the general queue decomposition method the GPS server of rate c is divided into a set of N servers with service rates $c_1, c_2, ..., c_N$, so that instead of having a GPS system with N sessions sharing a server, there is a decomposed system consisting of N separate queues, each of which has a dedicated server. The decomposition is such that $\sum_{i=1}^{N} c_i = c$ and $\lambda_i < c_i$, $\forall i \in 1, 2, ..., N$. Using this decomposition one obtains an output MMFP approximation of each of the N queues, which is then used to partition backlogged and unbacklogged queues at each state of the overall system. Define by $B(t)$ the set of backlogged queues, at any time instant t. Also, define the set of unbacklogged queues $\bar{B}(t)$, which is the complement of $B(t)$, i.e., the union of the two sets at any time is the set of all N logical queues in the system. Then the instantaneous service process $s_k(t)$ seen by an individual queue k is given by

$$s_k(t) = c_k + \frac{\varphi_k}{\varphi_k + \sum\limits_{l \in B(t)} \varphi_l} \sum_{l \in \bar{B}(t)} (c_l - r_l(t)), \qquad (4)$$

where $c_k = \varphi_k c$ is the minimum guaranteed rate of session k, and c the total system service capacity. Equation (4) is comprised of c_k increased by the residual service seen by queue k when there are unbacklogged queues in the system. The first term in the product on the right side is the fraction of the total residual service assigned to queue k according to the GPS allocation rule. The second term in the product represents the total available residual service in the system, and is equal to the sum of the residual service of individual unbacklogged queues. Notice that in the fluid model even if a queue l is empty (unbacklogged), part of its available service capacity may be consumed for an instantaneous input rate $r_l(t)$, which is in this case necessarily less than c_l. It can be shown [13] that by using (4) we obtain an upper bound on the buffer occupancy and delay distribution of each individual session. The specifics of analytical methods for both the infinite and finite queue cases are here omitted for brevity (see [13]). We next introduce an example of a traffic management scheme that combines buffer management and GPS scheduling. The analysis of this scheme is based on application of the techniques we outlined in the previous sections.

4 An Example: VP+GPS

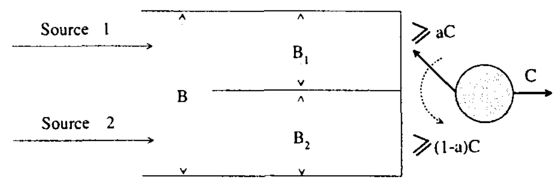

Figure 4: Complete Buffer Sharing with Virtual Partitioning.

Virtual Partitioning (VP) is a buffer management policy that controls the cell or packet access to the shared buffer when it overflows. In the two logical queues case, it involves a threshold that virtually partitions the buffer in two virtual buffers when it is full: B_1 buffer units are allocated to session 1, and $B_2 = B - B_1$ to session 2, where B is the total available buffer size. The VP rule is as follows (figure 4): (1) any incoming cell/packet has access to the buffer when there is free memory, and (2) when the buffer is full, a class 1 (2) packet can push-out (replace) a packet of session 2 (1), only if its current content X_1 (X_2) in the buffer is below B_1 (B_2). out of the buffer. This scheme provides buffer sharing, with isolation and fairness at the same time. Isolation is provided by guaranteeing a nominal buffer allocation B_i for each session i, and fairness is provided by penalizing a misbehaving session that occupies buffer space continuously at the cost of the other session. Combined with GPS, VP also guarantees that each session will be able to receive its nominal bandwidth share. In summary, the VP+GPS scheme provides for complete sharing, fairness, and isolation in the allocation of bandwidth and buffer resources. Additionally, the VP buffer allocation B_i, along with the GPS service allocation c_i, allows for an additional parameter to guarantee QoS for the two sessions in a finite buffer system. We provide numerical results in the following section.

5 Numerical Investigations

In order to verify the accuracy of our approximation we conducted a numerical investigation to compare analysis with simulation for the tail distributions $F_i(x_i)$, $i = 1, 2$ of the two sessions. Results are included here for the finite shared buffer case of two queues, and for the tail distribution of queue occupancy in the case of N sessions with infinite buffer. In all the simulation cases under study the simulation run lengths were such that the 95% confidence interval of an estimated value is within 20% of the value. Although the analysis accounts for any number of multistate MMFP sources in each session, we use two-state sources for simplicity. Each input traffic stream i consists of a two-state MMFP source (on-off source) characterized by the state transition rates a_i and input rate λ_i (on state) for source 1, and b_i and input rate μ_i (on state) for source 2. The average duration of the on state is kept equal to 1 in all cases, i.e. $1/a_2 = 1/b_2 = 1$. The total server capacity $c = 1$. To give some intuition we define the source activity factor ρ as the fraction of time a source is in the ON state, and the normalized source load u as the ratio of average source rate to server capacity ($\rho_1 = a_1/(a_1 + a_2)$, and $\rho_2 = b_1/(b_1 + b_2)$). In figure 5 the tail probability of both sessions is estimated and simulated with $\rho_1 = \rho_2 = 0.4$. The GPS assignment is

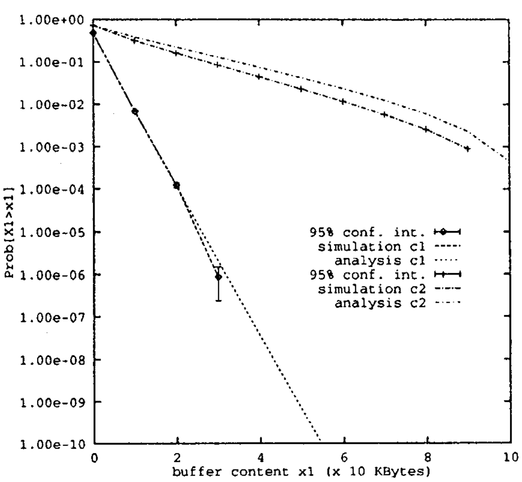

Figure 5: Tail distribution of sessions 1 and 2, with finite shared buffer $B = 10$, GPS assignment $c_1 = 0.8, c_2 = 0.2$.

$(\varphi_1, \varphi_2) = (0.8, 0.2)$. In order to give a more practical interpretation to these results we may equate the server capacity of one fluid unit per time unit to the standard OC-3 rate of 155 Mbps. By using a time unit of 0.27 msecs, one unit of fluid then corresponds to 100 ATM cells, so $B = 10$ is equivalent to 1000 cells. As more service is assigned to the first source the difference of the tail probabilities of the two sessions increases. Although this is a simple qualitative effect the analysis can help quantify it.

In figure 6 we demonstrate our results on tail distributions of the queue occupancy in the case of multiple sessions ($N = 3$). We also compare to simulations and

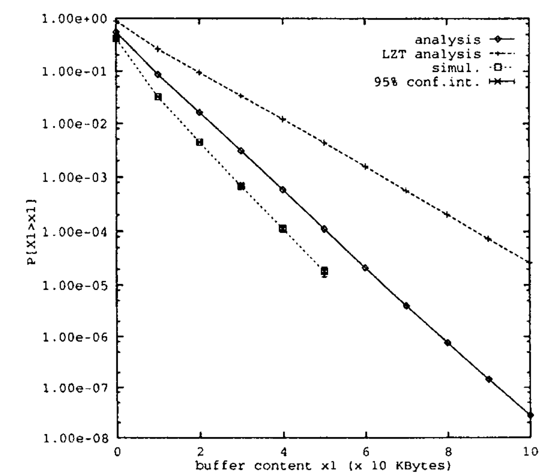

Figure 6: Comparative results of our approach, simulations, and the LZT bound for the case of three queues. We provide the tail distribution of an isolated session for asymmetric sources.

results of the method in [6] (LZT stands for the initials of the authors). We model three sessions with source parameters $\rho_1 = 0.5$, $\rho_2 = \rho_3 = 0.1$, and $u_1 = 0.7$, $u_2 = u_3 = 0.2$. We observe the tail distribution of the isolated queue of session 1. The tail distribution we predict is an upper bound of the simulation result, and a tighter bound than the one in the approach of [6]. The effect of improvement is more pronounced in cases where sessions 2 and 3 are bursty, and also when the number of sources increases. The intuition behind this is that in this case the queue of session 1 receives more residual service when the other queues are unbacklogged. This fact is efficiently captured in expression (4) for the modulated service process of an isolated queue. In figure 7 we compare the VP+GPS scheme to (i) a system sharing buffer among the two sessions with VP but fixed capacity allocation for each session ("VP only"), and (ii) a system sharing service using GPS but fixed partitioned buffers for each session ("GPS only"). The corresponding system parameters, i.e., source parameters, and all partitions are kept the same for all systems. We use two bursty sources with activity factor ρ=0.3. The loss of VP+GPS is less by about an order of magnitude, which indicates that it is a scheme that can efficiently allocate resources and has significant gains that allow more sessions to be accomodated in the system.

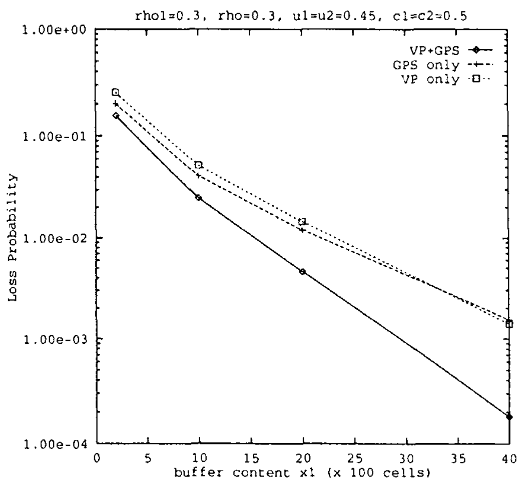

Figure 7: Comparison of VP+GPS to VP with partitioned bandwidth allocation and (V P only), and to GPS with partitioned buffer allocation (GPS only): $\rho_1 = \rho_2 = 0.3$, $u_1 = u_2 = 0.45$.

5.1 A Comparative Study with a Fixed Capacity Allocation System

Using our analysis we can explicitly demonstrate the significant improvement in resource allocation during admission control when using capacity scheduling with the GPS policy and shared buffering (with and without VP), as compared to a segregated buffer system with fixed capacity assignment for each priority class. We assume a scenario in which each source of the previous examples belongs to a traffic class with different loss requirements, $L_1 = 10^{-6}$ and $L_2 = 10^{-3}$. We then examine the minimum capacity that satisfies the loss requirement for a wide range of buffer sizes in three systems under consideration. In the first system each traffic stream i uses a dedicated buffer of size B_i and capacity C_i. To obtain the optimal capacity allocation for the overall system of buffer size B, we solve the minimization problem:

$$\textbf{Minimize } f(C_1, C_2) = C_1 + C_2$$
$$\textbf{Subject to: } L_1(C_1, B_1) = 10^{-6},\ L_2(C_2, B_2) = 10^{-3},$$
$$\text{and } B_1 + B_2 = B.$$

For each value of B the optimization procedure finds the optimal buffer (actual) partitioning (B_1, B_2) that minimizes the overall capacity C.

In the second and third system we use GPS scheduling with assignment (c_1, c_2) and complete sharing of a buffer of size B, with or without VP, respectively. For the overall system of buffer size B we solve the minimization problem:

$$\textbf{Minimize } f(C) = c_1 C + c_2 C = C$$
$$\textbf{Subject to: } L_1(C_1 = c_1 C, B) = 10^{-6},$$
$$L_2(C_2 = c_2 C, B) = 10^{-3},$$
$$\text{and } c_1 + c_2 = 1.$$

In this case the optimization procedure finds the minimum required capacity C as well as the optimal GPS assignment (c_1, c_2). For the case of VP we also determine the optimal virtual partition (B_1, B_2). Comparative numerical results were obtained using Matlab library functions for optimization and are illustrated in figure 8. There are two optimal capacity-buffer curves for fixed losses L_1 and L_2, one for the segregated buffer/capacity system, one for the complete buffer sharing GPS system without VP and one for the VP+GPS system.

In all three cases, as the buffer size increases less capacity is required to meet the loss requirements of the two priority classes. For all buffer sizes the shared buffer GPS systems require less capacity. Specifically, for buffer sizes from 500 to 2500 cells ($B = 5$ to 25) savings in capacity range from 35% to 10%. This is a region of typical buffer sizes of interest, roughly covering the first quarter of the x_1-axis in fig. 8. Observe also that in this region the optimal curves are much steeper. However the slope of the shared GPS system is smaller, which means that by reducing the available capacity by some amount, much less additional buffer is needed for the shared GPS system to meet the loss requirement, than in the segregated system. Alternatively, for any available capacity the shared buffer GPS system can meet the loss requirements using a smaller buffer size. This is a tradeoff that is useful in buffer sizing. Using VP improves GPS mainly for smaller buffer sizes where buffer management has a great impact on the system losses. The shared GPS system optimal curve also provides a means to compare how different implementations of the packetized GPS (see [14] for a survey) approach the ideal fluid analysis performance. It is also possible to derive optimal bandwidth-buffering curves for fixed mean or 95^{th} percentile delay of one or both priority classes. Preliminary results show that, as expected, when both delay and loss QoS constraints need to be met, the minimum capacity is determined by loss for small buffer sizes and delay for larger buffer sizes.

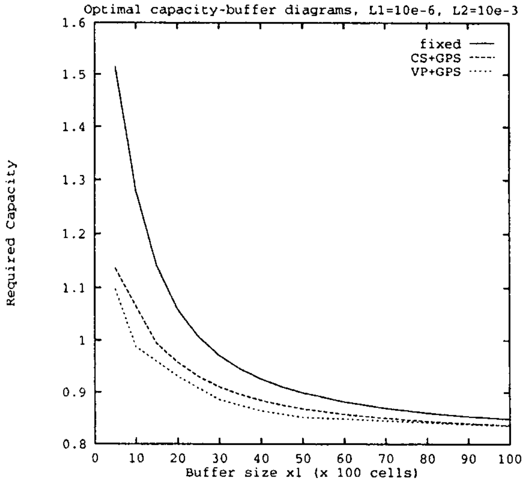

Figure 8: Comparison of the shared buffer GPS system with a segregated buffer fixed capacity allocation system of the same total buffer size. Two cases of buffer management are included for the shared buffer system: (i) no buffer ac cess control, (ii) buffer access control using VP.

6 Conclusions and Future Work

In this paper, we provide analytical techniques to approximate the performance of combining buffer management and bandwidth sharing using GPS scheduling between two or more sessions. Case studies for traffic loads of interest show that the analytical results for the case of two queues sharing a buffer approximate simulation results adequately for engineering purposes. We also validated our new upper bound approximation for the queue occupancy tail distributions in the case of $N > 2$, and confirmed the tightness of our bounds as compared to previous methods and simulations. Current work includes further validation of our method for multiple sessions sharing a finite buffer. Since this method mainly relies (i) on the accuracy of the tail distributions, and (ii) on the solution for two queues, we expect similar trends. The techniques presented in this work can be used by system designers for buffer dimensioning and by network managers for tuning buffer thresholds and scheduling parameters of switches/routers for near real time control. Our optimization results indicate that there can be a significant saving in the bandwidth allocation to sessions, by exploiting the buffer-bandwidth tradeoff.

References

[1] Yoram Bernet, et al., "A Framework for Differentiated Services," *Internet Draft*, February, 1999.

[2] Sally Floyd and Van Jacobson, "Link Sharing and Resource Management Models for Packet Networks," *IEEE/ACM Transactions on Networking*, Vol. 3, No. 4, August, 1995.

[3] V. P. Kumar, T. V. Lakshman, D. Stiliadis, "Beyond Best Effort: Router Architectures for the Differentiated Services of Tomorrow's Internet," *IEEE Commun. Magazine*, May, 1998.

[4] Mark. W. Garrett, "A Service Architecture for ATM: From Application to Scheduling," *IEEE Network Magazine*, pp.6-14, Vol. 10, No. 3, May-June, 1996.

[5] D. Clark, S. Shenker and L. Zhang, "Supporting Real-time Applications in an Integrated Services Packet Network: Architecture and Mechanism" *Proc. Sigcomm'92*, pp. 14-26, August, 1992.

[6] F. L. Presti, Z. -L. Zhang and D. Towsley, "Bounds, Approximations and Applications for a Two-Queue GPS System", *Proc. IEEE Infocom'96*, pp. 1310-1317, 1996.

[7] Z.-L. Zhang, D. Towsley, and J. Kurose, "Statistical analysis of Generalized Processor Sharing Scheduling Discipline." *IEEE J. on Sel. Areas in Comm.,*, Vol. 13, No. 6, pp. 1-10, Aug. 1995.

[8] Fabio M. Chiussi and Vijay Sivaraman, "Achieving High Utilization in Guaranteed Service Networks using Early-Deadline-First Scheduling", Proceedings of *IWQos'98*, Nova, California, May 1998. pp. 209-218.

[9] Sally Floyd and Van Jacobson, "Random Early Detection Gateways for Congestion Avoidance," *IEEE/ACM Transactions on Networking*, Aug. 1993.

[10] O. Yaron and M. Sidi, "Generalized Processor Sharing networks with Exponentially Bounded Burstiness Arrivals," *Proc. IEEE Infocom '94*, pp. 628-634, June 1994.

[11] Guo-Liang Wu and Jon W. Mark, "A buffer allocation scheme for ATM Networks: Complete sharing based on virtual partition," *IEEE/ACM Trans. on Networking*, Vol. 3, No. 6, Dec. 1995.

[12] George Lapiotis and Shivendra Panwar, "Approximate Loss and Delay Analysis of Two Priority Classes Sharing a Finite Buffer with GPS Scheduling," *Proc. of the 6^{th} International Conference on Telecommunication Systems, Analysis and Modeling*, Nashville, 1998.

[13] George Lapiotis "Stochastic Analysis of Joint Buffer Management and Service Scheduling in High-Speed Network Nodes," Ph.D. Dissertation, Polytechnic University, January 1999.

[14] A. Varma and D. Stiliadis, "Hardware Implementation of Fair Queueing Algorithms for ATM Networks," *IEEE Commun. Mag.,*, Vol. 35, No. 12, December 1997.