Submitted to GLOBECOM 98, Sydney, Australia, November 1998
Area: Communication Systems Integration and Modeling

VirtualClock With Priority Buffer: A Resource Sharing Algorithm

Jianmei Chen=, Changcheng Huang=, Michael Devetsikiotis[, Ioannis Lambadaris[

=Nortel Technology
PO Box 3511 Station C
Ottawa, Ontario K1Y 4H7, Canada

[Department of Systems & Computer Engineering
Carleton University
Ottawa, Ontario K1S 5B6, Canada
Tel: (613) 520-5730, Fax: (613) 520-5727
E-mail: mike@sce.carleton.ca

Abstract

We propose a new algorithm called the VirtualClock with Priority Buffer (VCPB) in this paper. VCPB combines VirtualClock algorithm with a new priority buffering strategy and therefore provides complete isolation of different traffic classes in a sharing environment. The isolation provided by VCPB allows different QoS requirements be satisfied at the same time. Furthermore the performance of various traffic classes sharing the same bandwidth and buffer is typically upper bounded by their performance in homogeneous traffic environments. Therefore the traditional effective bandwidth approach can be applied as in homogeneous cases. Simulation results show that, in contrast to FIFO or traditional VirtualClock service, VCPB can provide proper separations both in mean delay and in cell loss rate while maintaining maximum sharing of buffer and bandwidth.

This research was supported by grants from theTelecommunication Research Institute of Ontario (TRIO)

1 Introduction

Bandwidth and buffer are the two types of resources to be statistically shared in data networks. The mechanism used for bandwidth management is generally called scheduling. Scheduling controls the order in which individual cells or packets are serviced. Various scheduling algorithms have been proposed to provide fair sharing of the total bandwidth. In the Round-Robin (RR) scheduling approach, the total bandwidth is divided into time frames with equal or variable length; a fixed number of slots within each frame are assigned to an individual user; and different users can have different numbers of slots depending on their bandwidth requirements. Weighted Round-Robin (WRR) and Hierarchical Round-Robin (HRR) [10] are the two variants of RR. Another typical scheduling approach is based on virtual finishing times. In this approach, each time a packet or cell arrives, a timestamp which is computed based on virtual finishing time is tagged. Packets or cells are serviced in the order of their timestamps. Examples of this approach are the Fair Queueing algorithm (FQ) [5], the Weighted Fair Queueing algorithm (WFQ)[5], the VirtualClock algorithm[6] and the Packetized Generalized Processor Sharing (PGPS) [7]. An excellent survey of various scheduling algorithms can be found in [14]. Although it has been shown in [7] that, under the worst case scenarios, PGPS can deliver a guaranteed service, very few studies have been done on the performance of these scheduling algorithms under predictive service except the study in [4].

In [4], de Veciana and Kesidis extended the effective bandwidth approach to also include individual statistical constraints. Instead of using a shared buffer or FIFO queueing discipline, they proposed segregating statistically identical streams with similar QoS requirements in separated buffers, thereby eliminating the disruption within the shared buffer while sharing the total bandwidth via a scheduling algorithm. Although this may simplify the mathematical analysis, it also removes the benefit of shared buffer pool and results in low utilization of buffer.

For predictive service, a scheduling algorithm alone is not enough to provide different QoS required by individual traffic streams. Although a fair scheduling algorithm can provide guaranteed bandwidth for each stream and therefore provide separations in mean delay, it does not control the order of cell losses. The cell loss rate for each stream depends on buffering mechanism. Two of the most general schemes for buffer management are shared buffer pool and per-flow allocation [1]. The shared buffer pool scheme is sometimes called First Come First Use (FCFU) buffer management or Completely Sharing (CS) scheme. The per-flow allocation scheme allocates certain buffer space to each stream. Depending on the percentage of the buffer reserved for each stream, it can be further divided into two schemes: Complete Partitioning (CP) and Partial Sharing (PS) [12]. FCFU can maximize buffer utilization but can not provide separations in cell loss rate. In contrast, CP can provide separations in cell loss rate but results in low utilization. The problem with PS is the difficulty of calculating the buffer size to be reserved for each stream. While using priority strategy can improve FCFU, it does not protect low priority streams.

To solve the above problem, we propose a new algorithm called the VirtualClock with Priority Buffer (VCPB) in this paper. VCPB combines VirtualClock algorithm with a new priority buffering strategy and therefore provides complete isolation of different traffic classes in a sharing environment. The isolation provided by VCPB allows different QoS requirements be satisfied at the same time. Furthermore the performance of various traffic classes sharing the same bandwidth and buffer is typically upper bounded by their performance in homogeneous traffic environments. Therefore the traditional effective bandwidth approach can be applied as in homogeneous cases. Simulation results show that, in contrast to FIFO or traditional VirtualClock service, VCPB can provide proper separations both in mean delay and in cell loss rate while maintaining maximum sharing of buffer and bandwidth.

2 VirtualClock with Priority Buffer (VCPB) Algorithm

In VCPB, the VirtualClock algorithm is used as a bandwidth enforcement mechanism while a special priority loss strategy is used as a buffer management mechanism. Traditional priority strategy approaches typically use priority both as a buffer and as a bandwidth scheduling mechanism. As a result, higher priority traffic streams not only have higher priority buffer access but also higher priority bandwidth access. Thus, lower priority traffic streams cannot have guaranteed bandwidth allocations.

In addition to decoupling bandwidth management from buffer management, another key aspect of the VCPB algorithm that differentiates it from previous approaches is that a higher loss priority is assigned to the traffic class with the lower cell loss rate. Assuming that the cell loss requirements for each class are widely spaced (i.e. differ by at least one order of magnitude), this priority strategy will effectively eliminate the disruptions within a shared buffer in terms of cell loss rates. For example, it is easy to see that, no matter what kind of buffering mechanism being used, a traffic stream with a cell loss rate of, say 10^{-9}, has little influence on another traffic stream with a higher cell loss rate, say 10^{-6}, although they share the same buffer. But this is not true in reverse. By providing a higher priority to the traffic stream with 10^{-9} cell loss rate, we can protect it from the disruption of the traffic stream with 10^{-6} cell loss rates. Therefore, both can maintain their own cell loss rates. Some typical application requirements are shown in Table 1 [13] where the cell loss rates are either equal or at least one order of magnitude different from class to class (The same priority can be assigned to classes with equal cell loss rates). This validates our assumption above. In the following paragraphs, we will give a more detailed

Generic service type	**Virtual bandwidth**	**Tolerable error rate**	**Acceptable maximum delay**	**Tolerable delay variation**	**Maximum burst length**
High-quality real-time voice	≤ 0.20 Mb/s	$\le 10^{-3}$	~ 300 ms	≤ 30 ms	O(bytes)
Real-time video	≥ 4 Mb/s	$\le 10^{-6}$	~ 100 ms	~ 5 ms	O(kilobytes)
Web browsing	≥ 0.25 Mb/s	$\le 10^{-5}$	~ 100 ms	~ 10 ms	O(megabytes)
Multiparty network games	≥ 0.10 Mb/s	$\le 10^{-5}$	~ 50 ms	~ 5 ms	O(kilobytes to megabytes)
Working from home	≥ 1 Mb/s	$\le 10^{-4}$	~ 1 min	≤ 500 ms	O(megabytes)

Table 1: Common broadband to the home services and the required transmission characteristics

description of VCPB algorithm.

Suppose that J classes of ATM homogeneous traffic which commit to the requirements of cell loss and mean delay pair (P_j , D_j), are carried through an processor sharing node (PSN) with capacity C and a buffer pool with size B. Then the traditional call admission control (CAC) algorithm can be as following:

$$\sum_{j=1}^{J} \sum_{i=1}^{n_j(t)} \alpha_i^j \le C \qquad (1)$$

where $n_j(t)$ is the source number of traffic stream j at time t, α_i^j is the effective bandwidth of the i^{th} source of traffic stream j. Without a proper buffer and bandwidth isolation mechanism the above CAC algorithm is very difficult to be applied due to the difficulty of calculating effective bandwidth α_i^j under a heterogeneous environment.

The structure of the PSN node using VCPB algorithm is shown in Figure 1 where we assume that each type of homogeneous traffic has a "logical" FIFO queue. Since a FIFO queuing discipline is ideal for homogeneous traffic, it allows a number of sources to aggregate their traffic to obtain a lower overall delay jitter. Therefore each homogeneous traffic class j can be treated as a single traffic stream requiring a total bandwidth C_j. To provide a guaranteed bandwidth C_j, we use the VirtualClock scheduling algorithm in which the cell departing order is determined by the numerical order of their virtual finishing times. Let $a_i^j \in R^+$ be the time of the i^{th} cell arriving to FIFO queue j. The virtual finishing time of the i^{th} cell arriving to FIFO queue j can be calculated as:

$$F_i^j = max\{F_{i-1}^j, a_i^j\} + \frac{1}{C_j} \qquad \text{for} \qquad i \ge 1 \qquad and \qquad (2)$$

$$F_0^j = 0$$

The "logical" queues in Figure 1 are not physically separated. Let $B_j(t)$ be the queue size for the homogeneous traffic stream j, such that $\sum_{j=1}^{J} B_j(t) \le B$. Assume that the cell loss requirements P_j for each class j are widely spaced (i.e. differ by at least one order of magnitude). In VCPB we assign the priority of each class according to its cell loss requirement P_j. The higher the cell loss rate, the lower the cell loss priority. If $P_1 < \dots < P_j < \dots < P_J$, traffic class 1 has the highest priority, while traffic class *J* has the lowest. Different from the traditional VirtualClock, each cell will be associated with an extra parameter W_i^j when it arrives at the buffer, which indicates whether the cell is overflow or underflow with respect to its stream. If $F_{i-1}^{j} > a_i^j$ in Eq. (2), the cell is an overflow cell and $W_i^j = 1$, otherwise the cell is an underflow cell and $W_i^j = 0$. When the buffer is not full, every cell can access the buffer upon arrival. When the buffer is full, the cell's ability to access the buffer depends on its cell loss priority. If there is any cell in the buffer which has a lower loss priority than the newly arrived cell, and its parameter W_i^j indicates that it is an overflow cell, it can be pushed out of the buffer by the newly arrived cell; otherwise, the newly arrived cell will be lost. If a cell is pushed out of the buffer, the traffic stream to which this cell is attached must recalculate its Virtual Finishing Times without reference to the removed cell. The final algorithm can be implemented as follows where the differences between the VirtualClock algorithm [6] and the VCPB algorithm are highlighted.

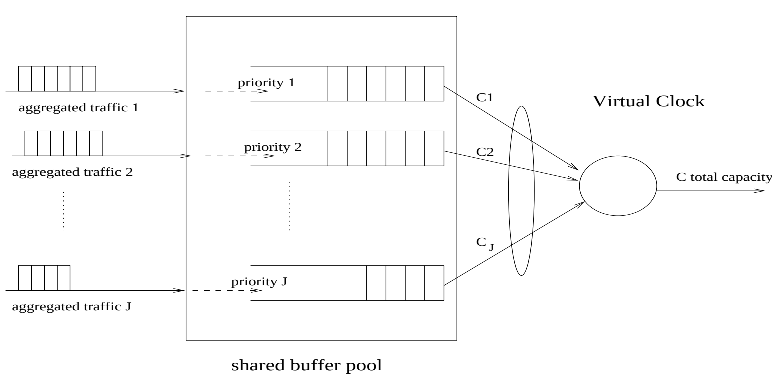

Figure 1: A processor sharing node with VCPB

Each homogeneous traffic stream has two variables: $VirtualClock_j$ and; $auxVC_j$ (which are consistent with the variables in the *VirtualClock* algorithm). Let $Vtick_j = 1/C_j$ *(second/cell).* The node will serve cells in the following manner:

- **i) Upon receiving the first cell from traffic class *j*:**

(a) $VirtualClock_j \leftarrow auxVC_j \leftarrow$ *real time*;

(b) $W = 0$;

(c) Stamp the cell with the $auxVC_j$ and ***W* value**;

(d) put the cell in the head of logical queue which belongs to traffic class *j*.

- **ii) Upon receipt of a cell from traffic class *j* when the buffer pool is not full:**

(a) $auxVC_j \leftarrow$ max (*real time*, $auxVC_j$);

(b) if *real time* $\geq auxVC_j$, $W = 0$; otherwise, $W = 1$;

(c) $VirtualClock_j \leftarrow (VirtualClock_j + Vtick_j)$, and $auxVC_j \leftarrow (auxVC_j + Vtick_j)$;

(d) Stamp the packet with the $auxVC_j$ value and ***W* value**;

(e) Put the cell at the end of the logical queue which belongs to traffic class *j*.

- **iii) Upon receipt of a cell from traffic class *j* when the buffer pool is full: Check the time stamp of the last cell in logical queue of class *J, J-1,........, j+1* (whose priority is lower than traffic class *j*), starting from class *J*.**

If we find that the last cell in the logical queue of class *i* is associated with $W = 1$ when $i \geq (j+1)$, stop checking and do the following:

(a) push out the last cell in logical queue *i*, $VirtualClock_i \leftarrow (VirtualClock_i - Vtick_i)$, $auxVC_i \leftarrow$ time stamp of the last cell current in the logical queue *i;*

(b) $auxVC_j \leftarrow$ max (*real time*, $auxVC_j$);

(c) if *real time* $\geq auxVC_j$, $W = 0$; otherwise, $W = 1$;

(d) $VirtualClock_j \leftarrow (VirtualClock_j + Vtick_j)$, and $auxVC_j \leftarrow (auxVC_j + Vtick_j)$;

(e) Stamp the packet with the $auxVC_j$ value and *W* value;

(f) Put the cell at the end of the logical queue which belongs to traffic class *j*;

otherwise:

discard the newly arrived cell of traffic class *j*.

- **The processor always serves the cells at the head of logical queues, in the order of their time stamps.**

The difference between VirtualClock and VCPB is how to insert the cells in the queues. In the VirtualClock algorithm, the cell is inserted into the outgoing queue according to its timestamp. When the buffer is full, cells are lost in the order of their timestamps. In the VCPB algorithm, whether the cells are lost is not only decided by its time stamp or the time it arrives, but also by its priority. The VCPB algorithm maintains several logical queues. Newly arrived cells can push out the overflow-tagged cells with lower loss priority when the buffer is full.

The bandwidth management mechanism in the VCPB algorithm guarantees the bandwidth allocation of C_j. The priority strategy in buffer management works in such a way that each traffic stream seems to own the whole buffer size *B* due to our unique priority assignment mechanism. Therefore performance prediction for each traffic stream can be based on the bandwidth C_j *and* buffer size *B* as in homogeneous case. Although it may be difficult to determine the exact effective bandwidth value for each class in a heterogeneous environment, the performance under heterogeneous environment should be bounded by the performance predicted in a homogeneous environment with the same bandwidth C_j *and* buffer size *B* due to the extra multiplexing gain under heterogeneous environment. Our simulation will show that, in most cases, the bound is tight. Therefore we can treat the effective bandwidth derived in the homogeneous environment as the effective bandwidth in the heterogeneous environment. It should be noted that, our new priority buffer management can be associated with any scheduling algorithms which provide guaranteed bandwidth allocation. The result will be the same. In this paper, we use VirtualClock only as an example.

3 Simulation Setup and Results

Although analytical approach can be applied to study certain queueing systems, it quickly becomes too complicated and intractable when the source model and the congestion control scheme become complex. In this paper, we will use simulation to evaluate the resource management schemes proposed above. In particular we will show simulation results contrasting the performance achieved by FIFO and VirtualClock with the VCPB algorithm we proposed.

Our simulation structure is shown in Figure 2. It contains two parts: a traffic generator and a network simulator (using OPNET [11]). To simplify the simulation scheme, only two classes of traffic will be generated by the traffic generator. Specifically they are ON-OFF traffic and video traffic. While we model ON-OFF source using the Bellcore ON-OFF model [8], we model video source using TES processes [9].

Sources within each traffic class are identical (with the same parameter settings). Therefore, the performance of homogeneous aggregated traffic stream can be considered to be the same as the performance of individual sources of this traffic class. Also, the call admission control procedure for individual sources is performed within homogeneous traffic.

In the follows, when we say that the simulation is homogenous, we are referring to the

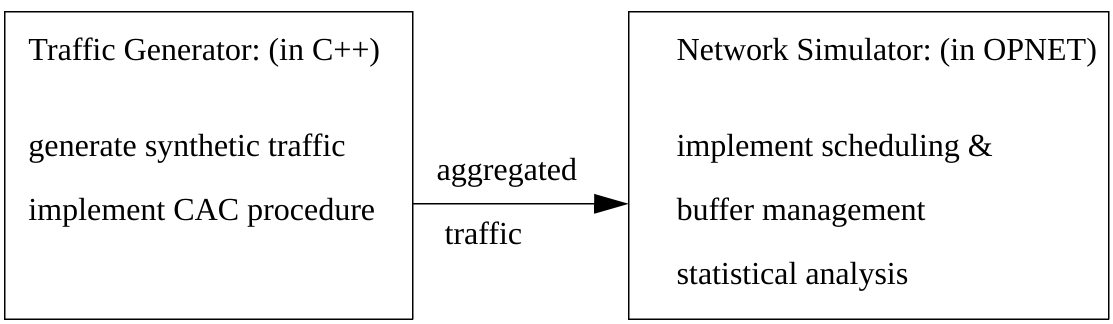

Figure 2: Simulation Structure

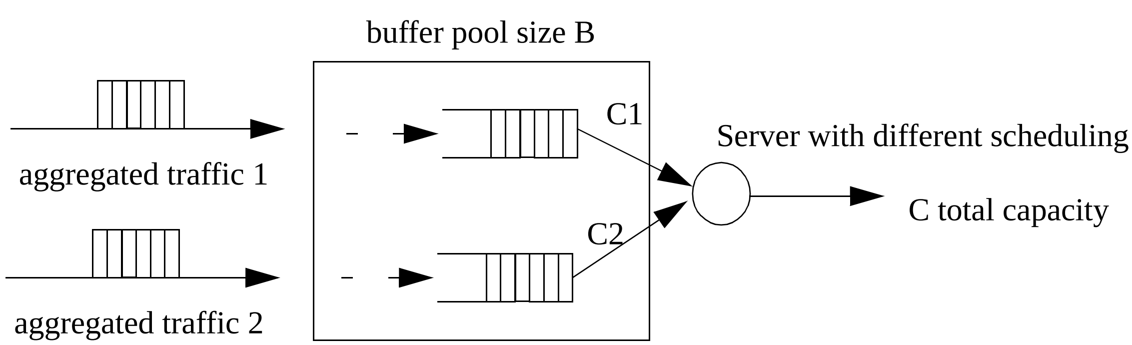

Figure 3: Network simulator structure in OPNET

structure shown in Figure 4. The network will have a total capacity C_j ($j = 1, 2$) and a total buffer space *B*. In contrast, the heterogeneous case refers to the structure similar to that shown in Figure 3. Total bandwidth C is shared among all traffic classes such that $C = C_1 + C_2$, and total buffer size is still *B*. We use mean delay and cell loss probability as our performance metrics.

We have simulated two profiles of mixed traffic loads in the heterogeneous traffic simulations. The first profile is: 50% ON-OFF traffic and 50% QTES-modeled video traffic. In this paper this traffic load ratio will be referred to as “balanced traffic load”. The required bandwidth for the ON-OFF traffic is the same as that for the QTES-modeled video traffic. The two traffic classes have the same utilization. The second profiles is: 25% ON-OFF traffic and 75% QTES-modeled video traffic (the unbalanced case). Due to the length limit, only the results of the balanced traffic load will be shown in this paper although they give the same conclusion. Interested reader can find more details in [15].

3.1 Performance Comparison of Two Traffic Models

Firstly, we examine the performance of different traffic streams in homogeneous environment. Results of cell loss probabilities together with their 95% confidence intervals are shown in Figure 5. For the same size of buffers as shown in Figure 5, the cell loss probability for the QTES-modeled video is much higher and the decrease with increasing buffer size is much slower compared to that in the ON-OFF traffic due to its bursty characteristic.In addition to cell

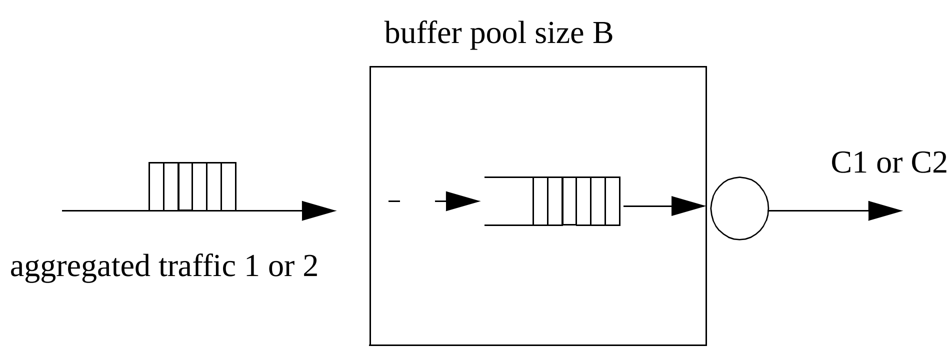

Figure 4: Homogeneous traffic structure

Traffic Type	Mean Delay	95% Confidence Interval
ON-OFF	2.94×10^{-5}	$(+/-)2.0 \times 10^{-7}$
Video	8.78×10^{-4}	$(+/-)4.9 \times 10^{-5}$

Table 2: Comparison of mean delay (ON-OFF traffic and QTES-modeled video traffic) in balanced traffic load case

loss probabilities, we give the values of mean delay in Table 2 where it is shown that the delays of the two models are also significantly different. Therefore the two traffic models can represent two classes of traffic.

3.2 Performance of FIFO Queueing discipline (FIFO Scheduling and FCFU Buffer Management)

Simulation results confirm that the two types of traffic suffer the same mean delay and the same cell loss probabilities in the heterogeneous case even though they have different

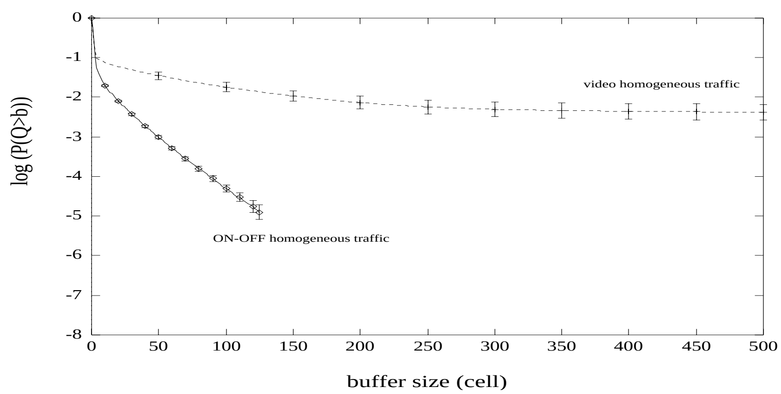

Figure 5: Queue tail distribution comparison of ON-OFF traffic and QTES-modeled traffic in balanced traffic load case

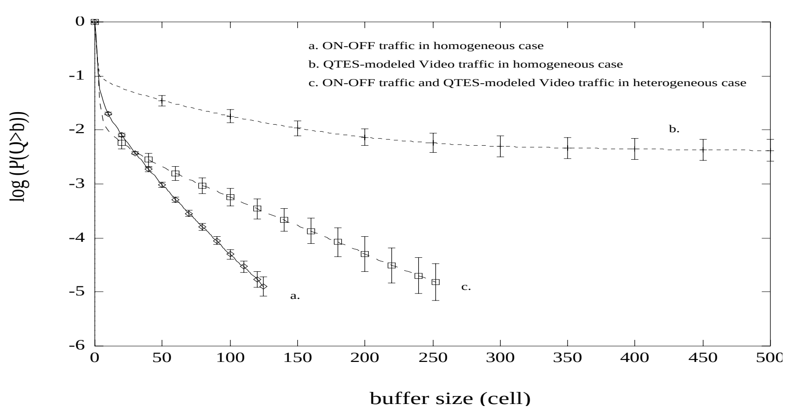

Figure 6: Queue tail distribution of aggregated traffic compared with homogeneous case in balanced traffic load case

performance in the homogenous case. In Figure 6, the cell loss rates of the ON-OFF traffic and the QTES-modeled video traffic are both equal to the cell loss rates of the aggregated traffic. QTES-modeled video traffic improves cell loss performance by benefiting from multiplexing with ON-OFF traffic, which has a low cell loss rates in the homogeneous case. However, the ON-OFF traffic is sacrificed. Its performance in terms of cell loss rates is degraded even though there should be a multiplexing gain for both traffic streams.

Traffic Type	**Mean Delay**	**95% Confidence Interval**
ON-OFF	1.31×10^{-5}	$(+/-) 1.9 \times 10^{-6}$
Video	1.72×10^{-5}	$(+/-) 1.8 \times 10^{-6}$

Table 3: Comparison of mean delay (ON-OFF traffic and QTES-modeled video traffic) while using FIFO discipline in balanced traffic load case

Mean delay has been calculated separately for each traffic as shown in Table 3. Although the values of the two types of traffic are not exactly the same due to the accuracy of the simulation, they can be considered to be very close in terms of the 95% confidence interval value. In comparison with the values of the homogeneous case in Table 2, the mean delay of both traffic types is not degraded. FIFO queueing allows a number of sources to aggregate their traffic to obtain a lower overall delay jitter. Therefore, QTES-modeled video traffic which has a higher mean delay in the homogeneous case has a lower mean delay when it is multiplexed with ON-OFF traffic. But there is no guarantee that the mean delay for the ON-OFF traffic will also be lower compared to that in the homogeneous case since it is multiplexed with the traffic with the higher mean delay. In summary, the FIFO queueing discipline cannot satisfy different QoS requirements.

3.2.1 Performance of VirtualClock Scheduling with FCFU Buffer Management

In this simulation scheme, we use the VirtualClock as a scheduling algorithm instead of using FIFO scheduling while the buffer management mechanism remains unchanged..

In Table 4, ON-OFF and QTES-modeled video traffic are seen to have different delays. The QTES-modeled video traffic suffers a larger delay because it is more bursty than the ON-OFF

Traffic Type	Mean Delay	95% Confidence Interval
ON-OFF	7.24×10^{-6}	$(+/-)6.8 \times 10^{-7}$
Video	2.07×10^{-5}	$(+/-)1.8 \times 10^{-6}$

Table 4: Comparison of mean delay (ON-OFF traffic and QTES-modeled video traffic) while using Virtual-Clock scheduling with FCFU buffer in balanced traffic load case

traffic as shown in homogeneous case. When using VirtualClock as a scheduling algorithm among different traffic classes, traffic classes will have the same guaranteed bandwidth usage as they have in the homogeneous traffic case, and in addition they might obtain additional bandwidth when they are multiplexed together. Therefore, the mean delay values are smaller when compared with those in Table 2. That is the benefit of multiplexing gain. Furthermore, the mean delay for the QTES-modeled video traffic is larger in Table 4 compared to Table 3; this shows that QTES-modeled video traffic consumes more bandwidth under a FIFO scheduling scheme than it declared.

However, as it becomes clear from the results, the cell loss probabilities for both traffic streams still cannot be separated. They will remain unchanged to what is shown in Figure 6, because the same buffer management mechanism (FCFU) is used. Therefore, this algorithm still cannot completely satisfy different QoS requirements.

3.2.2 Performance of The VCPB Algorithm

It is not a surprise that mean delays with VCPB are the same as in Table 4, since the VirtualClock algorithm is also used as the scheduling mechanism in the VCPB algorithm. For cell loss probability, different cell loss rates are obtained instead of a uniform cell loss rate due to the priority strategy in buffer management as shown in Figure 7. Therefore, a different QoS can be provided in the networks using the VCPB algorithm as a resource management method.

Case Type	Mean Delay	95% Confidence Interval
Homogeneous	2.94×10^{-5}	$(+/-)2.0 \times 10^{-7}$
Heterogeneous	7.24×10^{-6}	$(+/-)6.8 \times 10^{-7}$

Table 5: Comparison of mean delays for ON-OFF traffic with VCPB algorithm in balanced traffic load case

In comparing the performance of each traffic class under the VCPB algorithm with that in the

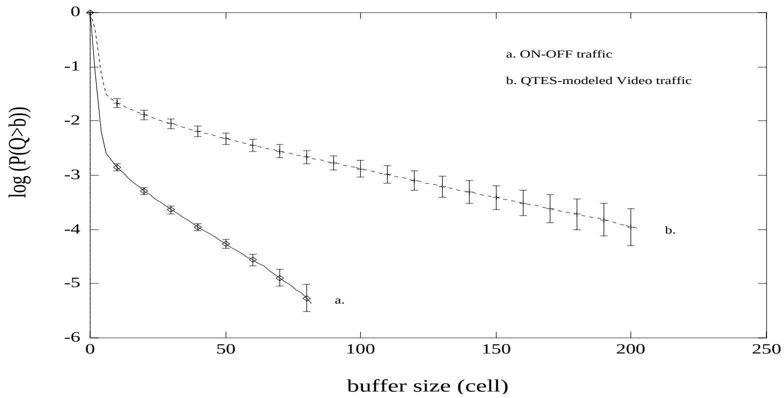

Figure 7: Queue tail distribution for ON-OFF traffic and QTES-modeled video with VCPB algorithm in balanced traffic load case

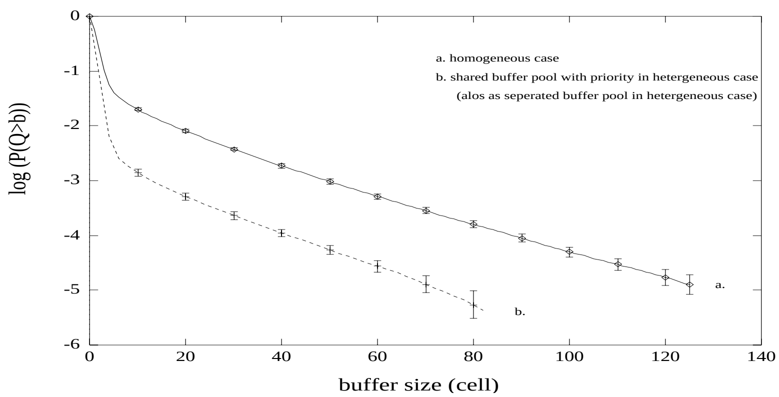

Figure 8: Queue tail distribution for ON-OFF traffic with VCPB algorithm in balanced traffic load case

Case Type	Mean Delay	95% Confidence Interval
Homogeneous	8.78×10^{-4}	(+/-)4.9×10^{-5}
Heterogeneous	2.07×10^{-5}	(+/-)1.8×10^{-6}

Table 6: Comparison of mean delays for video traffic with VCPB algorithm in balanced traffic load case

each traffic class under VCPB algorithm is bounded by its performance in the homogeneous environment. The mean delay of each traffic class is smaller compared to that in the homogeneous case due to the VirtualClock algorithm. As for cell loss rate, since the higher loss priority is given to the ON-OFF traffic which has lower cell loss rate, it seems that the buffer is totally used by ON-OFF traffic from its view-point. Hence, the cell loss rate of ON-OFF traffic can be bounded by that in the homogeneous case. For QTES-modeled video, its cell loss rate also decreased due to the bandwidth sharing which offsets the trivial effects of lower buffer access priority.

4 Conclusion

In this paper, we proposed VCPB as a new resource management scheme which can meet individual QoS requirements while preserving maximum resource sharing. In addition to the bandwidth sharing mechanism introduced by VirtualClock algorithm, VCPB also implemented a buffer management scheme based on a new priority strategy. This allows VCPB to commit both delay and cell loss requirements for different traffic streams in a sharing environment. Furthermore, we have shown that, for VCPB, the performance of various traffic classes in heterogeneous environments is upper bounded by their performance in homogeneous traffic environments. Therefore, we can predict the QoS of a traffic class in heterogeneous environments based on their QoS in homogeneous environments which can be easily characterized by effective bandwidth approach. Traffic descriptors (e.g. effective bandwidth) can now be treated as the attributes of a traffic class which are not dependent on other traffic classes to be multiplexed and therefore can be pre-calculated. Simulation results show that FIFO queue can not provide separations in both delay and cell loss rates. While VirtualClock can provide separations in delay, it can not provide separations in cell loss rates. In contrast, VCPB can provide separations in both delay and cell loss rates.

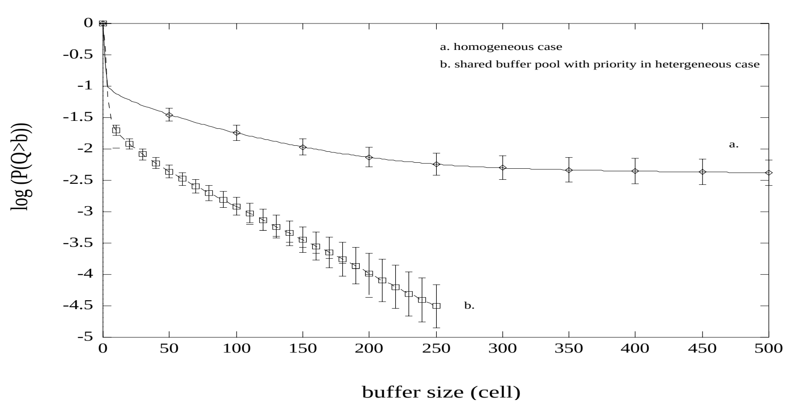

Figure 9: Queue tail distribution for QTES-modeled video traffic with VCPB algorithm in balanced traffic load case

References:

[1] C. Lefelhocz, B. Lyles, S. Shenker and L. Zhang, "Congestion Control for Best-Effort Service: Why We Need a New Paradigm", *IEEE Network*, vol. 10, no. 1, January/February 1996, pp. 10-19

[2] G. de Veciana, G. Kesidis and J. Walrand, "Resource Management in Wide-Area ATM Networks Using Effective Bandwidths", *IEEE JSAC*, vol. 13, no. 6, August 1995, pp. 1081 - 1090

[3] G. Kesidis, J. Walrand and C.-S. Chang, "Effective Bandwidths for Multiclass Markov Fluids and other ATM Sources", *IEEE/ACM Trans. on Networking*, vol. 1, no. 4, August 1993, pp. 424-428

[4] G. de Veciana and G. Kesidis, "Bandwidth Allocation For Multiple Qualities of Service Using Generalized Processor Sharing", *IEEE Globecom'94 Conference Record*, San Francisco, November 1994, pp. 1550-1554

[5] A. Demers, S. Keshav and S. Shender, "Analysis and Simulation of a Fair Queueing Algorithm", *Proceedings of SIGCOMM Computer Communications Review*, vol. 19, no. 4, September 1989, pp. 1-12

[6] L. Zhang, "Virtual Clock: A New Traffic Control Algorithm for Packet-Switched Networks", *IEEE/ACM Trans. on Computer Systems*, vol.9, no. 2, May 1991, pp. 101-124.

[7] A. K. Parekh and G. Gallager, "A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Single Node Case", *IEEE/ACM Trans. on Networking*, vol.1, no.3, June 1993, pp. 334-357

[8] Bell Communications Research, "Broadband Switching System (BSS) Generic Requirements", *GR-1110-CORE*, Issue 1, September 1994.

[9] M. R. Ismail, I. Lambadaris, M. Devetsikiotis and A. R. Kaye, "Modeling Prioritized MPEG Video Using TES and a Frame Spreading Strategy for Transmission in ATM Networks", *Proc. IEEE INFOCOM '95*, Boston, April 1995.

[10] G. Kesidis, *ATM Network Performance, A First Course*, Chapter 3, 1996.

[11] MIL3, Inc., *OPNET 2.5A Manual*, 1989-1994.

[12] G. Wu and J. W. Mark, "A Buffer Allocation Scheme for ATM Networks: Complete Sharing Based on Virtual Partition", *IEEE/ACM Trans. on Networking*, vol.3, no.6, December 1995, pp. 660-670

[13] B. Khasnabish, "Broadband to the Home (BTTH): Architectures, Access Methods, and the Appetite for it", *IEEE Network*, January/February 1997, pp. 58-69

[14] A. Varma and D. Stiliadis, "Hardware Implementation of Fair Queuing Algorithms for Asynchronous Transfer Mode Networks", *IEEE Communication Magazine*, December 1997, pp. 54-68

[15] J. Chen, "Virtual Clock with Priority Buffer: A Resource Sharing Algorithm", Master's Thesis, Carleton University, August 1997