Per-VC Rate Allocation Techniques for ATM-ABR Virtual Source Virtual Destination Networks*

Rohit Goyal, Raj Jain, Xiangrong Cai, Sonia Fahmy, Bobby Vandalore
Department of Computer Information Science
The Ohio State University
2015 Neil Avenue, DL 395
Columbus, OH 43210

New Address: Raj Jain, Washington University in Saint Louis,
jain@cse.wustl.edu, http://www.cse.wustl.edu/~jain

Abstract

We describe issues in designing rate allocation schemes in ATM-ABR networks for virtual source / virtual destination (VS/VD) switches. We propose a rate allocation scheme for VS/VD switches that uses per-VC queuing and per-VC control. We analyze the performance of this scheme, and conclude that VS/VD can help in limiting buffer requirements of switches, based on the length of their VS/VD control loops. *VS/VD is especially useful in isolating terrestrial networks from the effects of long delay satellite networks by limiting the buffer requirements of the terrestrial switches.*

1 Introduction

Of these, the The Available Bit Rate (ABR) service class in ATM has been specifically developed to support data applications. Traffic is controlled intelligently in ABR using a rate-based closed-loop end-to-end traffic management framework [1]. Several switch algorithms have been developed [2, 4, 5, 6] to calculate feedback intelligently.

One of the options of the ABR framework is the Virtual Source/Virtual Destination (VS/VD) option. The virtual source virtual destination (VS/VD) behavior specified for the ATM Available Bit Rate Service allows ATM switches to split an ABR control loop into multiple control loops. Each loop can be separately controlled by the nodes in the loop. The coupling between adjacent ABR control loops has been left unspecified by the ATM forum standards, and is implementation specific. On one loop, the switch behaves as a destination end system, i.e., it receives data and turns around resource management (RM) cells (which carry rate feedback) to the source end system. On the next loop, the switch behaves as a source end system, i.e., it controls the transmission rate of every virtual circuit (VC) and schedules the sending of data and RM cells. Such a switch is called a "VS/VD switch". In effect, the end-to-end control is replaced by segment-by-segment control as shown in Figure 1.

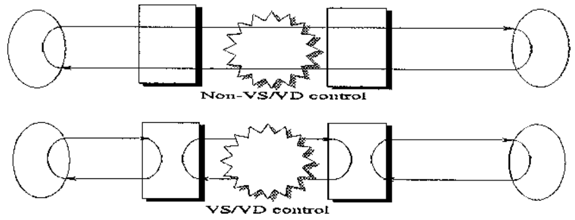


Figure 1: End-to-End Control vs VS/VD Control

VS/VD control can isolate different networks from each other. For example, two ABR networks can be isolated from a non-ATM network that separates them. Also, long latency satellite networks can be isolated from terrestrial networks so as to keep the effects of large latency to within the satellite loop.

VS/VD implementation in a switch, and the coupling of adjacent control loops present several design options to switch manufacturers. A VS/VD switch is required to enforce the ABR end-system rules for each VC. As a result, the switch must be able to control the rates of its VCs at its output ports. Per-VC queuing and scheduling can be used to easily enforce the rate allocated to each VC. With the ability to control per-VC rates, switches at the edge of the VS/VD loops can respond to congestion notification from the adjacent loop by controlling their output rates. Switches can also use downstream congestion information, as well as their internal congestion information, to provide feedback to the upstream loop. The ability to perform per-VC queuing adds an extra dimension of control for switch traffic management schemes. Rate allocation mechanisms can utilize the per-VC control at every virtual end system (VS/VD end point) for dimensioning of resources for each VS/VD loop. Not much work has been done in examining the options for VS/VD control in ATM switches.

In this paper, we present several issues in VS/VD switch design. We describe the basic architectural components of a VS/VD switch. We describe problems that may arise from naive implementations of feedback con-

[*] This research was sponsored in part by Rome Laboratory/C3BC Contract #F30602-96-C-0156.

0-7803-4984-9/98/$10.00 ©1998 IEEE.

trol schemes taken from non-VS/VD schemes. We then present a rate allocation scheme for feedback control in a VS/VD switch. We present simulation results with this scheme to show that VS/VD can help in switch buffer sizing, and isolation of users sharing a link.

2 A VS/VD Switch Architecture

Figure 2 illustrates the basic architecture of an output buffered VS/VD switch. The figure shows two output ports of the switch, and the data and RM cell flow of a VC going through the switch. Data and RM cells arrive at the input side of port 1. Data cells are switched to the appropriate destination port to be forwarded to the next hop. RM cells are turned around and sent back to the previous hop. For the VC shown in the figure, port 1 acts as the VD that accepts the data cells and turns around the RM cells, while port 2 acts as the VS for the next hop. Port 1 provides feedback to the upstream node in the VC's path by inserting congestion and rate information in the appropriate RM cell fields. Port 2 sends the data to the next hop, generates an RM cell every *Nrm* cells, and enforces all the source rules specified in the ABR end-system behavior. Port 1 also accepts and processes the turned around BRM cells returned by the downstream end system in the VC's path.

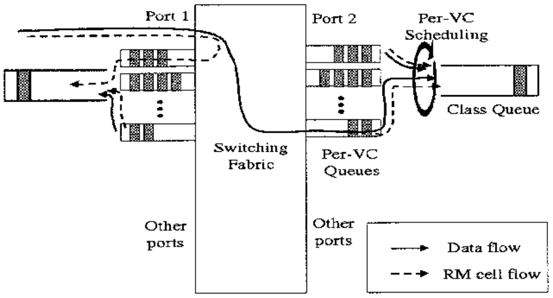


Figure 2: VS/VD switch architecture

Each port has a class queue for the ABR service category, as well as per-VC queues for each ABR VC.[1] Each per-VC queue contains the data cells and turned around RM cells for its VC. Each per-VC queue drains into the class queue at the ACR allocated to the corresponding VC. The class queue drains at the link rate of the outgoing link.

A scheduling mechanism ensures that each VC gets a fair share of the total link capacity. In principle, the scheduling policy must allow the VS to send at the rate that is allowed by a combination of the allocation policy and the end system behavior. However, when ACRs are overbooked, the scheduling policy must service the per-VC queues in some fair proportion of their ACR or MCR values. Details of scheduling policy design are a topic of future study.

3 Design Issues For Explicit Rate Allocation with VS/VD

Figure 3 shows a queuing model for a single port of an **output buffered non-VSVD switch** (node i). The port has one class queue for the ABR VCs. Cells from all the ABR VCs destined for the output port are enqueued in the class queue in a FIFO manner. Let the input rate of VC_j into node j be s_{ij}, and the input rate into the class queue be r_{ij}. In this case, since the node simply switches cells from the input to the output port, we have $s_{ij} = r_{ij}$. Let R_i be the output rate of the class queue at the given port of node i. Then R_i corresponds to the total bit rate of the link available to ABR. Let q_i be the queue length of the class queue. Let N be the number of ABR VCs sharing the link.

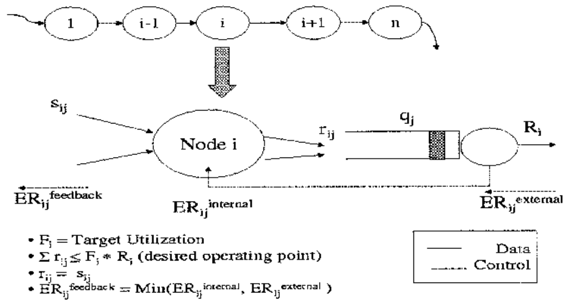

1
i-1
i
i+1
n
s_{ij}
Node i
r_{ij}
q_j
R_i
$ER_{ij}^{feedback}$
$ER_{ij}^{internal}$
$ER_{ij}^{external}$
• F_i = Target Utilization
• $\Sigma r_{ij} \leq F_i * R_i$ (desired operating point)
• $r_{ij} = s_{ij}$
• $ER_{ij}^{feedback} = Min(ER_{ij}^{internal}, ER_{ij}^{external})$
Data
Control

Figure 3: Queuing model for non-VS/VD switch

Many rate allocation algorithms use a parameter $F_i, (0 < F_i \leq 1)$ which is the target utilization of the link. The link rate is allocated among the VCs so that

$$\sum_{j=1}^{j=N} r_{ij} \leq F_i R_i$$

i.e., the goal of the switch is to bring the total input rate into the class queue to the desired value of $F_i R_i$. Let $ER_{ij}^{internal}$ be the ER calculated by the node based on the internal congestion in the node. This is the rate at which the switch desires VC_j to operate. Node i also receives rate allocation information from the downstream node $(i+1)$. This is shown in the figure as $ER_{ij}^{external}$. Node i provides feedback to the upstream node $(i-1)$, as

$$ER_{ij}^{feedback} = Min(ER_{ij}^{internal}, ER_{ij}^{external})$$

At node $(i-1)$, $ER_{ij}^{feedback}$ is received as $ER_{(i-1)j}^{external}$, and node $(i-1)$ performs its rate calculations for VC_j in a similar fashion.

The internal explicit rate calculation is based on the local switch state only. A typical scheme like ERICA [2], uses several factors to calculate the explicit rate. In particular, the ERICA algorithm uses the total input rate to the class queue, the target utilization of the link, and the number of VCs sharing the link to calculate the

[1] The class queue is not essential if per-VC queuing and scheduling are used, but we include it to illustrate a general architecture. The class queue can be removed without affecting the scheme presented in this paper.

desired operating point of each VC in the in the next feedback cycle, i.e.,

$$ER_{ij}^{internal} = \text{fn}(\sum_j r_{ij}, F_i R_i, N)$$

In steady state, the ERICA algorithm maintains $\sum_j r_{ij} = F_i R_i$, so that any queue accumulation due to transient overloads can be drained at the rate $(1-F_i)R_i$. As a result, the ERICA algorithm only allocates a total of $F_i R_i$ to the VCs sharing the link, and results in $100F_i\%$ steady state link utilization of the outgoing link.

The ERICA+ algorithm can achieve 100% steady state link utilization by additionally considering the queue length of the class queue when it calculates the internal rate for VC_j, i.e., for ERICA+,

$$ER_{ij}^{internal} = \text{fn}(\sum_j r_{ij}, g(q_i) R_i, N)$$

where $g(q_i)$, $(0 < g_{min} \leq g(q_i) \leq g_{max})$ is a function known as the *queue control function*, that scales the total allocated capacity R_i based on the current queue length of the class queue. If q_i is large, then $g(q_i) < 1$ so that $\sum_j r_{ij} = g(q_i)R_i$ is the target operating point in the next feedback cycle, and $(1 - g(q_i))R_i$ can be used to drain the queue to a desired value (q_i^{target}). The queue control function is bounded below by $g_{min} > 0$ so that at least some minimal capacity is allocated to the VCs. A typical value for the ERICA+ algorithm of g_{min} is 0.5. When the queue is small, $(q_i < q_i^{target})$, $g(q_i)$ may increase to slightly more than 1 so that sources are encouraged to send at a high rate. As a result, switches try to maintain a pocket of queues of size q_i^{target} at all times.

In the remainder of this section, ERICA and ERICA+ are used as a basis for our discussion. However, the discussion is general, and applies to any rate allocation scheme that uses the target utilization and queue length parameters in its rate calculations. The discussion presents some fundamental concepts that should be used in the design of rate allocation algorithms for VS/VD switches.

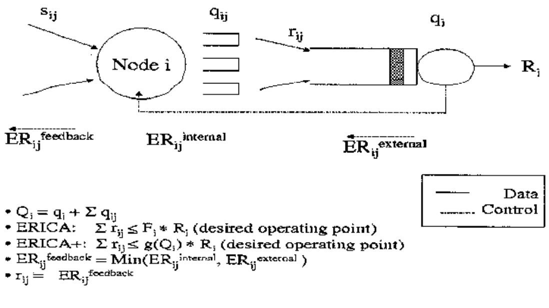

s_{ij}
q_{ij}
r_{ij}
q_i
Node i
R_i
$ER_{ij}^{feedback}$
$ER_{ij}^{internal}$
$ER_{ij}^{external}$
Data
Control
• $Q_i = q_i + \Sigma q_{ij}$
• ERICA: $\Sigma r_{ij} \leq F_i * R_i$ (desired operating point)
• ERICA+: $\Sigma r_{ij} \leq g(Q_i) * R_i$ (desired operating point)
• $ER_{ij}^{feedback} = \text{Min}(ER_{ij}^{internal}, ER_{ij}^{external})$
• $r_{ij} = ER_{ij}^{feedback}$

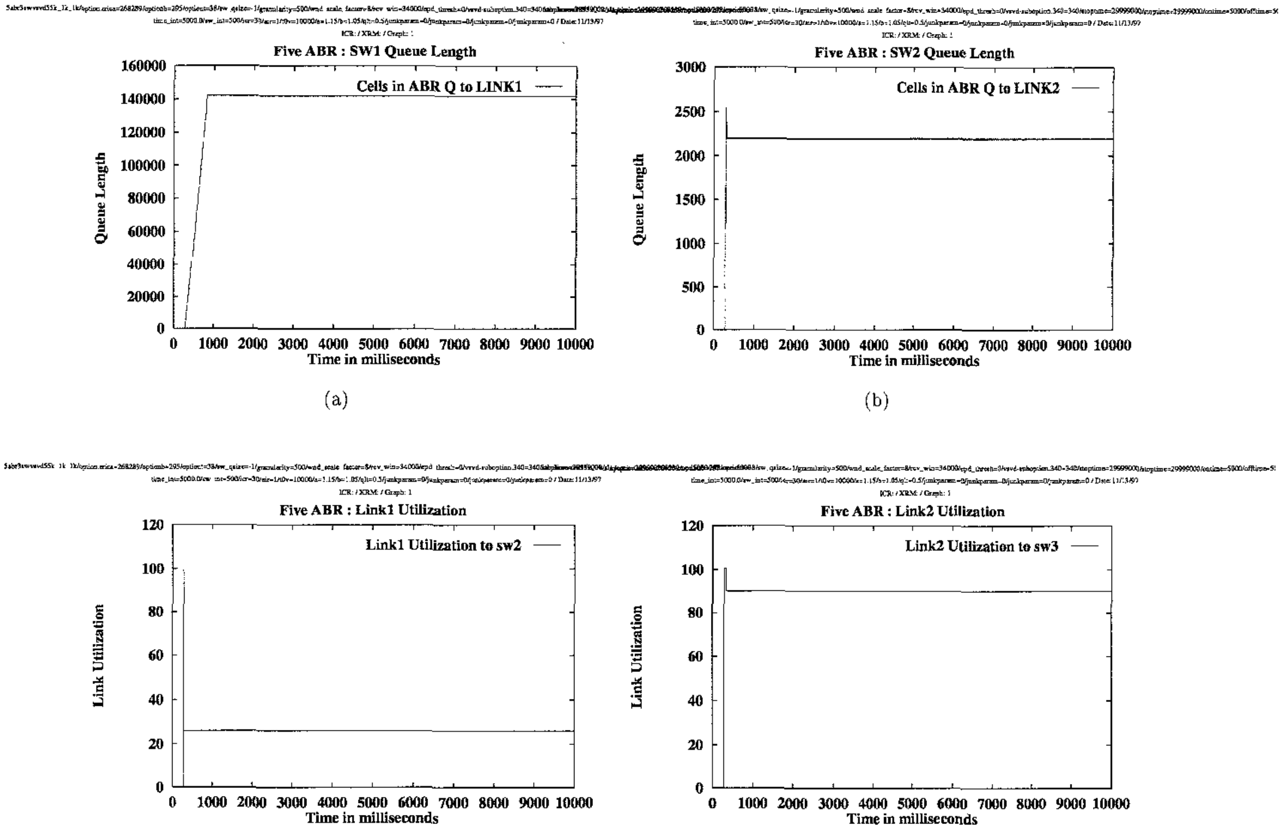
Figure 4: Simple queuing model for a VS/VD switch

Figure 4 illustrates a simple adaptation of ERICA and ERICA+ to a VS/VD switch [3]. The VS/VD switch can control the rates of the per-VC queues. r_{ij} is the rate at which VC_i's per-VC queue drains into the class queue. Like in ERICA, r_{ij} is set to the $ER_{ij}^{feedback}$ value calculated by the node. The explicit rate is calculated as before for both ERICA and ERICA+. ERICA+ uses the sum of the per-VC queues and the class queue for the queue control function. The key feature in this adaptation is that the output rate of the per-VC queue is set to the desired input rate at the class queue. This value is also fed back to the upstream hop of the previous loop. This simple approach can present problems in some cases.

Suppose that node i is the bottleneck node for VC_j, i.e., $ER_{ij}^{internal} < ER_{ij}^{external}$, and $ER_{ij}^{feedback} = ER_{ij}^{internal}$. As a result, VC_j of node $(i-1)$ sends at a rate of $ER_{ij}^{internal}$, i.e., the input rate to VC_j's per-VC queue is $s_{ij} = ER_{ij}^{internal}$. Also, the VC_j's queue drains at the rate $r_{ij} = ER_{ij}^{internal}$. Thus, the per-VC queue of VC_i can not recover from transient overloads and results in an unstable condition. This is shown in the simulation results in figure 5. The figure shows the queue lengths and percentage link utilizations for the configuration shown in figure 7 and described in section 5. Both switch 1 and switch 2 queues build up during the open loop control phase of the simulation. When the closed loop VS/VD control sets in, the queues cannot drain because the input and output rates of each switch are the same. When the queues build up, the link utilization of link 2 (the bottleneck link) should be 100%. However, the class queue in switch 2 is empty because the sum of the per-VC queues is only $F_i Ri$ with $F_i = 0.9$. As a result, the utilization of link 2 is 90% of the expected value.

The problem with the above scheme is that it ignores the existence of an ABR server at each VC-queue. The scheme uses the explicit rates calculated by the server at the class queue, and uses these as the output rates for the per-VC queues. As a result, the sum total of the output rates of the per-VC queues is limited to $F_i R_i$, hence limiting the drain rate of the class queue to the same value. The $(1-F_i)R_i$ capacity is thus never usable since $\sum_j r_{ij} \leq F_i R_i$.

Figure 6 shows a better model for a VS/VD switch. The presence of servers at the per-VC queues is explicitly noted, and the input rates to the per-VC queues are not the same as their output rates. Separate servers are shown before each queue, because these servers process the cells before they enter the queue. The servers at the per-VC queues also control the output rates of their respective queues. In the case of ERICA, the sum total of the input rates to the class queue is limited by $F_i R_i$. This allows the class queue to drain in case of transient overloads from the per-VC queues. The input to the per-VC queues (s_{ij}) is limited by $F_i r_{ij}$, allowing the per-VC queues to also use a rate of $(1 - F_i)r_{ij}$ to recover from transient overloads. Moreover, for an ERICA+ like scheme that uses queue length information to calculate available capacity, additional per-VC queue information is now used to control s_{ij} in relation



(a) (b)

(c) (d)

Figure 5: Performance of incorrect implementation of VS/VD

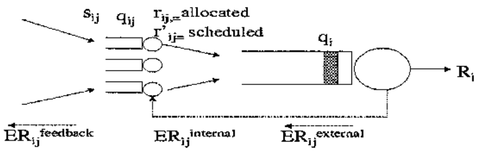
to r_{ij}. Thus, for ERICA+, the desired operating point is decided for the next feedback cycle such that

$$\sum_j r_{ij} \leq g(q_i)R_i$$

and

$$s_{ij} \leq g(q_{ij})r_{ij}$$

The feedback given to the previous loop is set to the desired per-VC operating point, which is the desired input rate to the per-VC queues. As a result, the per-VC feedback is further controlled by the VC's queue length. This can be used to further isolate the VCs on the same link from one another. Thus, if VC_j experiences a transient overload, only $ER_{ij}^{feedback}$ is reduced and the feedbacks to the remaining VCs are not affected by this temporary overload.



- ERICA: (desired operating point)
  - $\Sigma\, r_{ij} \leq F_i * R_i$
  - $s_{ij} \leq F_i * r_{ij}$
- $ER_{ij}^{internal}$ Min={Max(r'_{ij}/Overload, $F_i * R_i/N$), $ER_{ij}^{external}$ }
- r_{ij}= fn($ER_{ij}^{external}$, end system rules)
- $ER_{ij}^{feedback} = F_i * r_{ij}$

- ERICA+: (desired operating point)
  - $\Sigma\, r_{ij} \leq g(q_i) * R_i$
  - $s_{ij} \leq g(q_{ij}) * r_{ij}$
- $ER_{ij}^{internal}$ = Min{Max(r'_{ij}/Overload, $g(q_i) * R_i/N$), $ER_{ij}^{external}$ }
- r_{ij}= fn($ER_{ij}^{external}$, end system rules)
- $ER_{ij}^{feedback} = F_i * r_{ij}$

Figure 6: Queuing model for per-VC VS/VD switch

4 A Per-VC Rate Allocation Algorithm for VS/VD

The scheme presented in this section is based on the ERICA+ scheme for ABR feedback [2]. The basic switch model is shown in figure 6. The switch maintains an averaging interval at the end of which it calculates the rate allocations ($ER_{ij}^{internal}$) for each VC to provide feedback to the previous hop. $ER_{ij}^{internal}$ is calculated for each VC based on the following factors:

- The actual (measured) scheduled rate of the VC queue into the class queue or the link ($\hat{r}_{ij}$).
- The allocated rate (ACR) of the VC queue into the class queue or the link (r_{ij}).
- The queue length of the class queue (q_i).
- The output rate of the class queue (R_i). This is also the total estimated ABR capacity of the link.
- The number of active ABR VC's sharing the class queue (N).
- The external rate allocation received by each VC from the downstream hop ($ER_{ij}^{external}$).
- The queue control function $g()$.

A portion of the link capacity $g(q_i)R_i$ is divided in a max-min fair manner among the per-VC queues[2]. The remaining portion is used to drain the class queue formed due to transient overloads. Then, the per-VC feedback is calculated for the upstream hop based on the per-VC queue length (q_{ij}) and the allocated rate (ACR) of the per-VC queue (r_{ij}). This calculation allocates a fraction (that depends on the queue length) of r_{ij} to the previous hop as $ER_{ij}^{feedback}$ so that s_{ij} in the next cycle is less than r_{ij} thus allowing the per-VC queue to drain out any transient overloads.

The basic design is based on the following principle. A desired input rate is calculated for each queue in the switch, and this desired rate is given as feedback to "the previous server" in the network. In the case of the class queue, the previous server controls the per-VC queues of the same node. The previous server for the per-VC queue is the class queue of the upstream hop in the VS/VD loop.

The basic algorithm consists of the following steps.

- When a BRM cell is received, the ER in the RM cell is copied to $ER_{ij}^{external}$.
- When an FRM cell is received, it is simply turned around, and its ER is stamped with the value $ER_{ij}^{feedback}$.
- Rate calculations are performed only once every averaging interval as follows

$$\begin{aligned}
\text{Overload} &\leftarrow \frac{\sum_j \hat{r}_{ij}}{g(q_i)R_i} \\
ER_{ij}^{internal} &\leftarrow \text{Min}\{\text{Max}\left(\frac{\hat{r}_{ij}}{\text{Overload}}, \frac{g(q_i)R_i}{N}\right), \\
&\qquad ER_{ij}^{external}\} \\
r_{ij} &\leftarrow \text{fn}(ER_{ij}^{internal}, \text{end-system rules}) \\
ER_{ij}^{feedback} &\leftarrow g(q_{ij})r_{ij}
\end{aligned}$$

This results in s_{ij} in the next feedback cycle to be $g(q_{ij})r_{ij}$. The remaining features and options of the algorithm are the same as the ERICA+ algorithm [2].

5 Simulation Results

In this section we present simulation results to highlight the features of the VS/VD rate allocation scheme presented in this contribution, and its potential advantages over non-VS/VD switches. In particular, we are interested in comparing the buffer requirements of a VS/VD switch with those of a non-VS/VD switch.

5.1 Configuration

Figure 7 shows the basic configuration used in the simulations. The configuration consists of three switches separated by 1000 km links. The one way delay between the switches is 5 ms. Five sources send data as

[2] In the absence of a class queue, the function $g(q_i)R_i = F_iR_i$ where $F_i \leq 1$ is the target utilization of the link

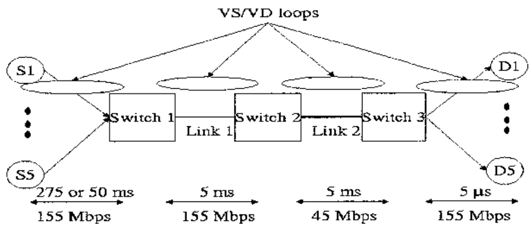


Figure 7: Five sources satellite configuration

shown in the figure. The first hop from the sources to switch 1 is a long delay satellite hop We simulated two values of one way delay - 275 ms (GEO satellite delay), and 50 ms (LEO satellite delay). The link capacity of link 2 is 45 Mbps, while all other links ar 155 Mbps links. Our simulations use infinite ABR sources. ABR Initial cell rates are set to 30 Mbps in all experiments. Thus, only link 2 is the bottleneck link for the entire connection.

5.2 Results

Figure 9 illustrates the difference in the maximum buffer requirements for a VS/VD switch and a non-VS/VD switch with the GEO satellite delay configuration. Switch 1 is connected to the satellite hop and is expected to have large buffers. Switch 2 is a terrestrial switch, and its buffer requirements should be proportional to the delays experienced by terrestrial links. Without VS/VD, all queues are in the bottleneck switch (switch 2). The delay-bandwidth product from the bottleneck switch to the end system is about 150,000 cells (155 Mbps for 550 ms). This is the maximum number of cells that can be sent to switch 2 before the effect of its feedback is seen by the switch. Figure 9(d) shows that without VS/VD, the maximum queue length in switch 2 is proportional to the feedback delay-bandwidth product of the control loop between the ABR source and the bottleneck switch. However, a terrestrial switch is not expected to have such large buffers, and should be isolated from the satellite network. In the VS/VD case, (figure 9 (a) and (b)), the queue is contained in switch 1 and not switch 2. The queue in switch 2 is limited to the feedback delay-bandwidth product of the control loop between switch 1 and switch 2. The observed queue is always below the maximum expected queue size of about 3000 cells (155 Mbps for 10 ms).

Figure 8 shows the corresponding result for the LEO satellite configuration. Again, with the VS/VD option, queue accumulation during the open loop period is moved from switch 2 to switch 1. The maximum queue buildup in switch 1 during the open loop phase is about 35000 (155 Mbps for 120 ms). Our simulations show that the corresponding link utilizations for link 1 and link 2 are comparable for VS/VD and non-VSVD. The ACRs allocated allocated to each source show that the resulting scheme is fair in the steady state.

This demonstrates that VS/VD can be helpful in limiting buffer requirements in various segments of a connection, and can isolate network segments from one another.

6 Summary and Future Work

In this paper, we have presented a per-VC rate allocation mechanism for VS/VD switches based on ERICA+. This scheme retains the basic properties of ERICA+ (max-min fairness, high link utilization, and controlled queues), and isolates VS/VD control loops thus limiting the buffer requirements in each loop. We have shown that VS/VD, when implemented correctly, helps in reducing the buffer requirements of terrestrial switches that are connected to satellite gateways. Without VS/VD, terrestrial switches that are a bottleneck, must buffer cells of upto the feedback delay-bandwidth product of the entire control loop (including the satellite hop). *With a VS/VD loop between the satellite and the terrestrial switch, the queue accumulation due to the satellite feedback delay is confined to the satellite switch. The terrestrial switch only buffers cells that are accumulated due to the feedback delay of the terrestrial link to the satellite switch.*

References

[1] ATM Forum, "ATM Traffic Management Specification Version 4.0," April 1996, available as ftp://ftp.atmforum.com/pub/approved-specs/af-tm-0056.000.ps

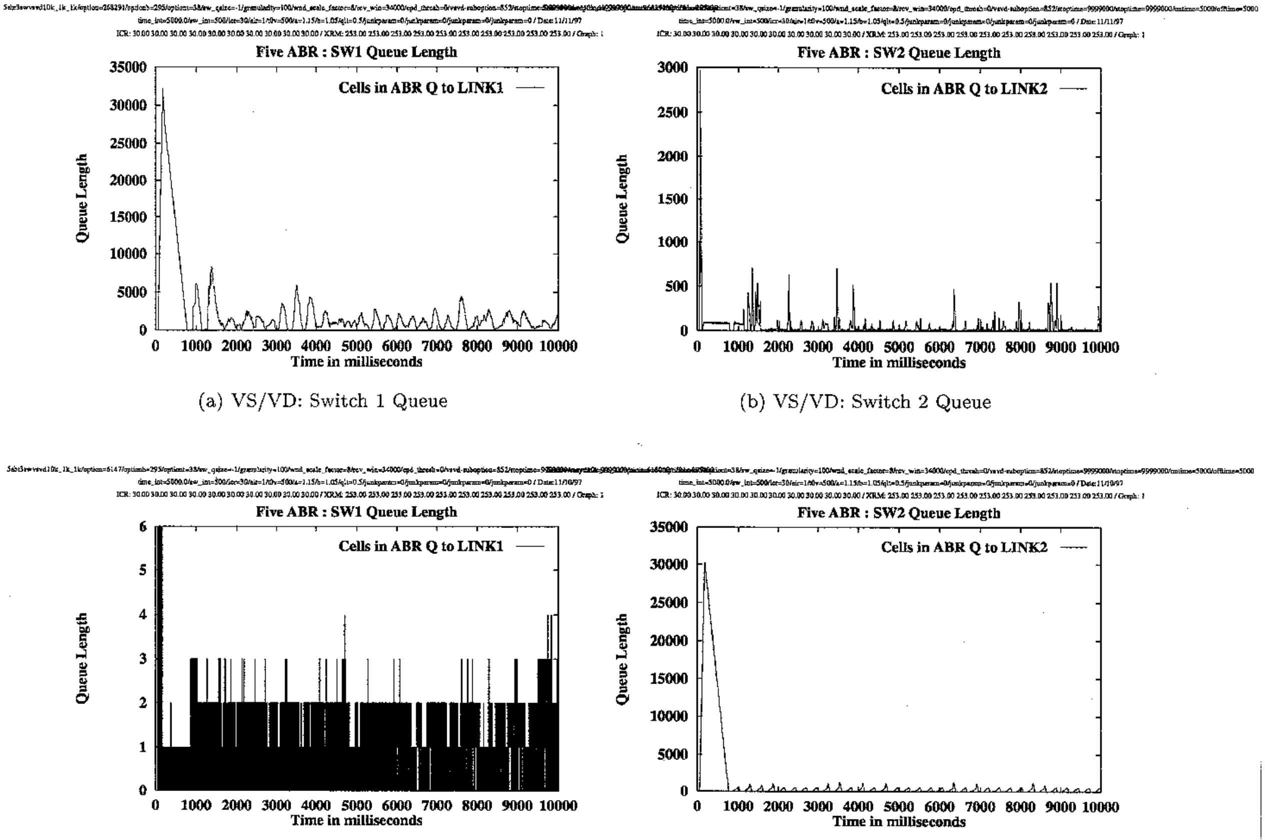
[2] R. Jain, S. Kalyanaraman, R. Goyal, S. Fahmy, and R. Viswanathan, "The ERICA Switch Algorithm for ABR Traffic Management in ATM Networks, Part I: Description," IEEE Transactions on Networking, submitted.

[3] Shivkumar Kalyanaraman, Raj Jain, Jianping Jiang, Rohit Goyal, and Sonia Fahmy, Seong-Cheol Kim, "Virtual Source/Virtual Destination (VS/VD): Design Considerations," ATM Forum/96-1759, December 1996.

[4] L. Roberts, "Enhanced PRCA (Proportional Rate-Control Algorithm)," *AF-TM 94-0735R1*, August 1994.

[5] L. Kalampoukas, A. Varma, K. K. Ramakrishnan, "An efficient rate allocation algorithm for ATM networks providing max-min fairness," Proceedings of the 6th IFIP International Conference on High Performance Networking, September 1995.

[6] Y. Afek, Y. Mansour, and Z. Ostfeld, "Phantom: A simple and effective flow control scheme," Proceedings of the ACM SIGCOMM, August 1996.



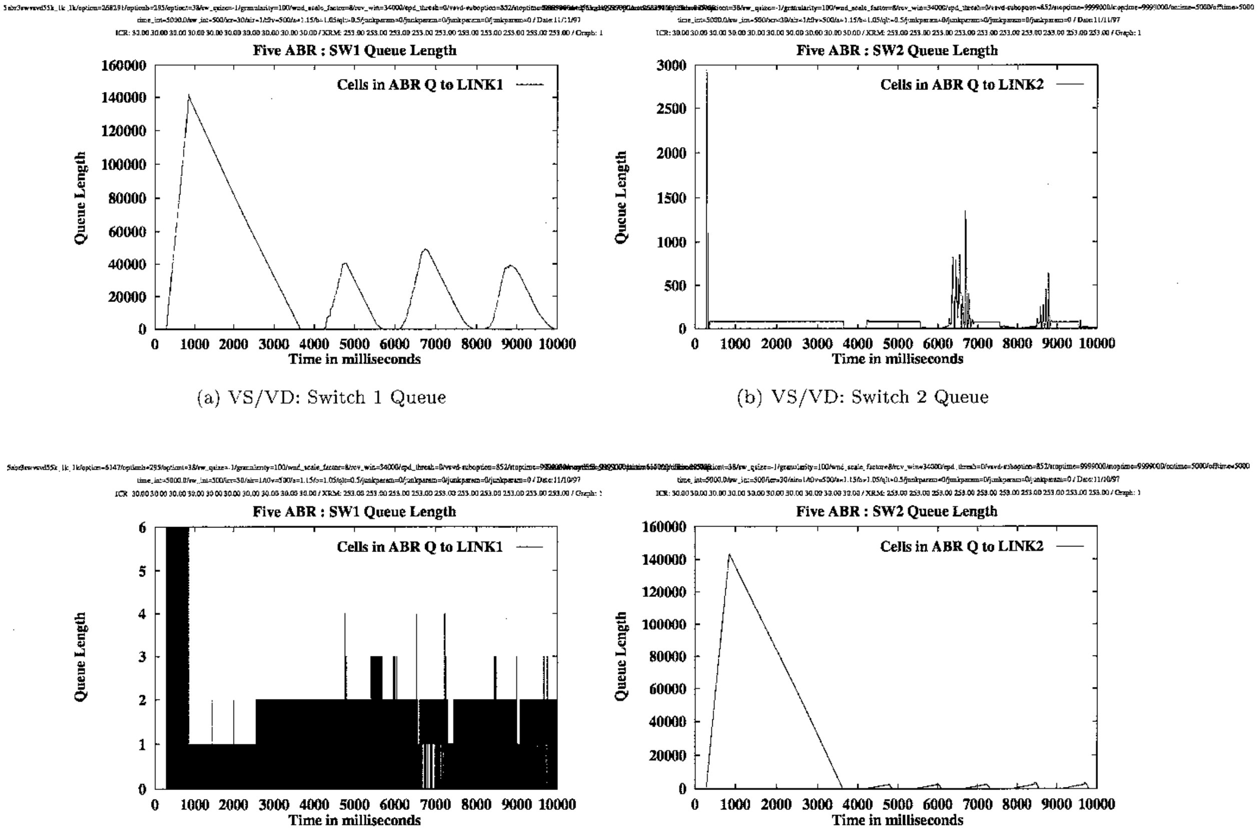
(a) VS/VD: Switch 1 Queue

(b) VS/VD: Switch 2 Queue

(c) Non-VS/VD: Switch 1 Queue

(d) Non-VS/VD: Switch 2 Queue

Figure 8: Switch Queue Length for VS/VD and non-VS/VD Case: LEO



(a) VS/VD: Switch 1 Queue

(b) VS/VD: Switch 2 Queue

(c) Non-VS/VD: Switch 1 Queue

(d) Non-VS/VD: Switch 2 Queue

Figure 9: Switch Queue Length for VS/VD and non-VS/VD:GEO