****

ATM Forum Document Number: ATM Forum/96-1759

****

Title: Virtual Source/Virtual Destination (VS/VD): Design Considerations

****

Abstract:

The ABR service specification allows a switch to act like a virtual destination as well as a virtual source. The switch divides an ABR connection into separately controlled ABR segments. However the coupling between the two adjacent ABR control segments is implementation specific. We study the issues in designing such a coupling between ABR segments and list a variety of implementation options. Only a small subset of the implementation options are viable. We present simulation results to support our arguments.

****

Source:

Shivkumar Kalyanaraman, Raj Jain, Jianping Jiang, Rohit Goyal, and Sonia Fahmy
The Ohio State University (and NASA)
Department of CIS

Seong-Cheol Kim
Samsung Electronics Co. Ltd.
Chung-Ang Newspaper Bldg.
8-2, Karak-Dong, Songpa-Ku
Seoul, Korea 138-160
Email: kimsc@metro.telecom.samsung.co.kr

****

Date: December 1996

****

Distribution: ATM Forum Technical Working Group Members (Traffic Management)

****

Notice: This contribution has been prepared to assist the ATM Forum. It is offered to the Forum as a basis for discussion and is not a binding proposal on the part of any of the contributing organizations. The statements are subject to change in form and content after further study. Specifically, the contributors reserve the right to add to, amend or modify the statements contained herein.

****

1 Introduction

The normal ATM ABR control loop is an end-to-end control loop in the sense that all control (resource management (RM) cells) flow from the source to the destination, where they are turned around back to the source. The traffic management specification TM4.0 allows the control loop to be segmented into shorter control loops. These control loops are implemented using switches with virtual source/virtual destination (VS/VD) capability. A VD switch receives data and turns around forward RM (FRM) cells as if it were a destination. Similarly, a VS switch controls the transmission rate of the virtual circuit (VC) and schedules the sending of FRMs, backward RM (BRM) cells and data as if it were a source. A single switch can act as VD on one side and VS on the other. It is also allowed to have a non-ABR or non-ATM network between a VD and the next VS. In effect, a VS/VD switch divides the ABR connection into separately controlled ABR segments. The end-to-end control is replaced by segment-by-segment control. The difference between end-to-end control and VS/VD is shown in figure 1.

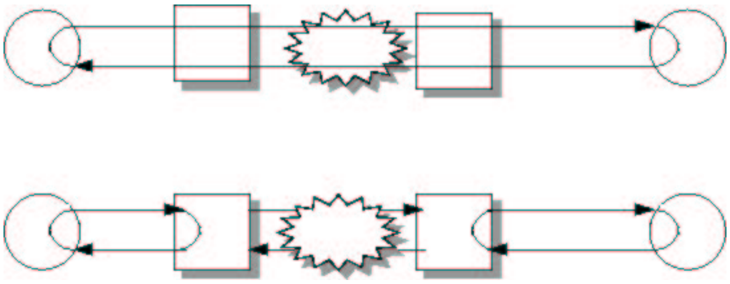


Figure 1: End-to-End Control vs VSVD Control

One use of segment-by-segment control is to isolate different networks from each other. One could envisage a proprietary network between two ABR segments which allows end-to-end ABR connection setup across the proprietary network and forwards ATM packets between the ABR segments. Another reason for using VS/VD is that shorter loops may potentially improve performance by giving feedback faster to the sources when new bursts are seen.

VS/VD requires the implementation of per-VC queueing and scheduling in the switch. However, if per-VC queueing and scheduling are already in place, the incremental cost to implement VS/VD is small. Hence, VS/VD may be used to improve the performance of switches implementing per-VC queueing and scheduling.

This contribution is first in a sequence of studies that we plan to do on VS/VD. The goals of this sequence are as follows:

- What changes to switch algorithms are required to operate in VS/VD environments?
- Do VS/VDs improve performance?
- There are no standards for passing informantion between VS and previous VD. Would multivendor VS/VDs interoperate? Or, are some rules required to be standardized?

- Are there any effects of control loops of different lengths?

In this first contribution, we attempt to answer only some of these questions. We show that VS/VDs, if implemented correctly, improve performance. The standardization issues are not addressed in this contribution. We study the possible changes to switch algorithms to be implemented in a VS/VD setup and evaluate the various implementation options. The last goal of the effect of control loops of different lengths is also for future study.

This contribution is organized as follows. We first present our model of the switch queue structure for the VS/VD switch and describe the data and RM cell flow in Section 2. Next we present the points at which the rate calculations are done in the basic ERICA switch scheme [2, 3] when implemented in a VS/VD switch in Section 3. We then enumerate all the possible VS/VD design option combinations in Section 4 and evaluate them in Section 5. The results and future work are summarized in Sections 7 and 8.

2 Switch Queue Structure

In this section we present a simple switch queue model for the non-VS/VD switch and extend it to a VS/VD switch by introducing per-VC queues We also present the data, FRM and BRM flow in our VS/VD switch queue structure.

2.1 Non-VS/VD Switch

A switch which does not implement VS/VD minimally requires FIFO queues for the different service classes: CBR, VBR and ABR. We refer to these as "per-class" queues. This shown in figure 2. The ABR rate allocation algorithm is used at each ABR class queue to calculate the rates for the contending ABR VCs.

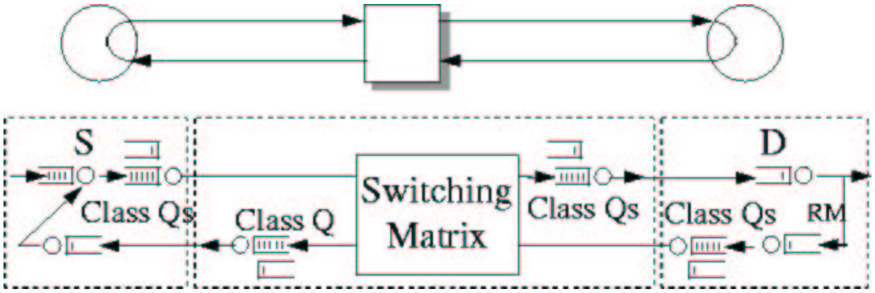


Figure 2: Per-class queues in a non-VSVD switch

The figure also shows a source end system, S, and a destination end system, D. These end systems have per-VC queues which feed into the connecting link. For simplicity, we have assumed a per-class queue at the entry to each link. This implies that the ABR VCs at the source feed into the ABR class queue, and the VBR VCs feed into the VBR class queue. Specifically, the ABR per-VC output rate is controlled to the Allowed Cell Rate (ACR) and the VBR rate is controlled by the leaky bucket mechanism. Scheduling is then done between the CBR, VBR, ABR and UBR queues for link access.

2.2 VS/VD Switch

The VS/VD switch implements the source and the destination functionality. Hence, like any source and destination, it requires per-VC queues to control the rates of individual VCs. We assume a switch queue structure where the per-VC queues feed into the per-class queues. Observe that the switch queue structure is now similar to the source and destination queue structure. This switch queue structure is shown in figure 3.

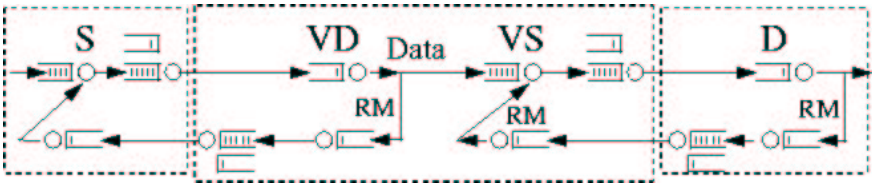


Figure 3: Per-VC and per-class queues in a VSVD switch

However, the switch also needs to implement the ABR rate allocation algorithm and calculate the allocations for VCs depending upon its available rate. Figure 3 shows a single unidirectional VC operating on a VS/VD switch. We now describe the data and RM cell flow in such a VS/VD switch.

2.3 Unidirectional Data Flow

The Virtual Destination (VD) forwards the data cells from the first segment (previous loop) to the per-VC queue at the Virtual Source (VS) of the second segment (next loop). The VS sends out the data cells along with new FRM cells as specifed in the source rules.

The VD turns around FRM cells as specified in the destination rules. Additionally, when the FRM cells are turned around, the switch may decrease the value of ER field to account for the rate available on the next and subsequent links.

When the VS at the next loop receives a BRM cell, the ACR of the per-VC queue at the VS is updated using the ER field in the BRM cell.

The model can be extended to multiple unidirectional VCs in a straightforward way. Figure 4 shows two unidirectional VCs, VC1 and VC2, between the same source S and destination D which go from Link1 to Link2 on a VS/VD switch. There is a separate VS and VD control for each VC. This and all subsequent figures will omit non-ABR queues.

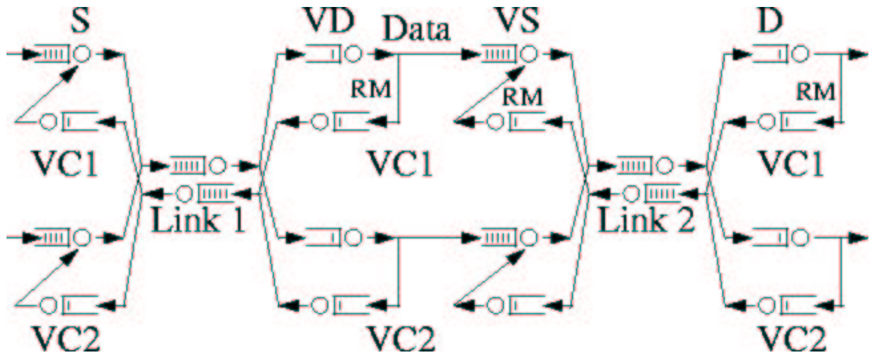


Figure 4: Multiple unidirectional VCs in a VSVD switch

2.4 Bi-directional Data Flow

In a bi-directional flow, each end system acts as a source for the flow in one direction and as a destination for the flow in the other direction. The VS/VD switch now has separate VSs and VDs for the two directions as shown in figure 5. The data on the incoming VD is forwarded to the outgoing VS, FRMs are turned around for the VS on the same segment, BRMs are processed by the VS on the same segment to update the corresponding ACRs.

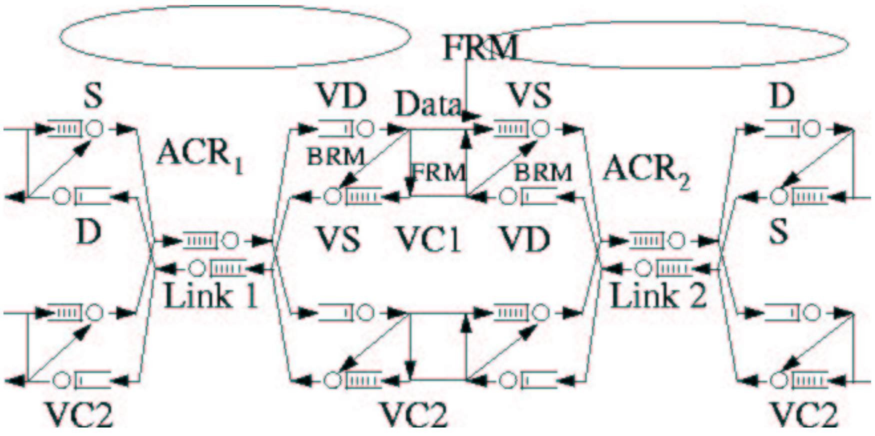


Figure 5: Multiple bi-directional VCs in a VSVD switch

We will discuss only forward (from left to right) flow of VC1. This flow has two ACRs: ACR_1 on Link1 and ACR_2 on Link2. All subscripts refer to the links. For example, FRM1 is an FRM on link 1 and so on.

The performance of ABR depends heavily on the switch algorithm used. We use the ERICA algorithm for rate allocation at the switches. This algorithm is briefly described next.

3 Basic ERICA Switch Scheme

In the most basic form of ERICA, the network manager sets a target utilization for each link. This is used to calculate the target rate for ABR as follows:

Target Rate = Target Utilization × *Link Rate - VBR rate - CBR rate*

The switch measures the input rate to the ABR class queue and the number of active ABR sources and uses them as follows:

To achieve fairness, the VC's Allocation (VA) has a component:

$$\mathrm{VA}_{\mathrm{fairness}} = \text{Target Rate/Number of Active VCs}$$

To achieve efficiency, the VC's Allocation (VA) has a component:

$$\mathrm{VA}_{\mathrm{efficiency}} = \text{VC's Current Cell Rate/Overload}$$

where,

$$\text{Overload} = \text{Input Rate/Target Rate}$$

Finally, the VC's allocation on this link (VAL) is calculated as:

$$\mathrm{VAL} = \mathrm{Max}\{\mathrm{VA}_{\mathrm{efficiency}}, \mathrm{VA}_{\mathrm{fairness}}\}$$

The explicit rate (ER) field in the RM cell is updated as:

$$\mathrm{ER} = \mathrm{Min}\{\mathrm{ER}, \mathrm{VAL}\}$$

There are many other steps in the algorithm which are described in [2]. We now describe where the rate calculations are done in a non-VS/VD switch and in a VS/VD switch.

3.1 Rate Calculations without VS/VD

The non-VS/VD switch calculates the rate for sources when the BRMs are processed in the reverse direction. Briefly,

$$\mathrm{VAL} = \mathrm{Fn}\{\text{Input Rate}, \text{VC's current rate}\}$$
$$\mathrm{ER} = \mathrm{Min}\{\mathrm{ER}, \mathrm{VAL}\}$$

At the source, ACR for the flow is calculated as follows

$$\mathrm{ACR} = \mathrm{Fn}\{\mathrm{ER}, \text{VC's current ACR}\}$$

3.2 Rate Calculations with VS/VD

Figure 6 shows the variables used in the rate calculations with VS/VD implemented at the switch. Specifically, the segment starting at Link2 returns an ER value, ER_2 in the BRM2. The FRM of the first segment FRM1 is turned around with an ER value of ER_1. The rate calculations are as follows:

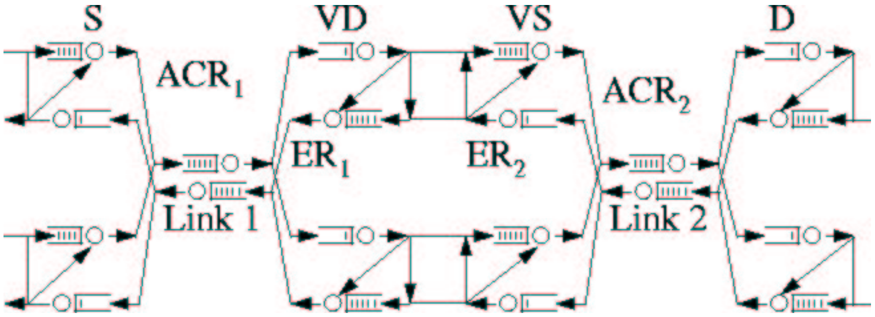


Figure 6: Rate calculations in VS/VD switches

- Destination algorithm for the previous loop:

$$\mathrm{ER}_1 = \mathrm{Min}\{\ \mathrm{ER}_1, \mathrm{VAL}_2, \mathrm{ACR}_2\ \}$$

$$\mathrm{VAL}_2 = \mathrm{Fn}\{\ \text{Input Rate}, \text{VC's Rate}\ \}$$

- Switch Algorithm for the next loop:

$$\mathrm{ER}_2 = \mathrm{Min}\{\ \mathrm{ER}_2, \mathrm{VAL}_2\ \}$$

$$\mathrm{VAL}_2 = \mathrm{Fn}\{\ \text{Input Rate}, \text{VC's Rate}\ \}$$

- Source Algorithm for the next loop:

$$\mathrm{ACR}_2 = \mathrm{Fn}\{\ \mathrm{ER}_2, \text{Current ACR}_2\ \}$$

We shall see in the next section that there are several ways of combining the feedback from the next loop, updating the ACR of the next loop, measuring VC rates and input rates to give feedback to the previous loop.

4 VS/VD Switch Design Options

With VS/VD, the rate of a flow in different control loops is different. Therefore, it is not obvious what some of the usual variables mean in such a situation. In particular, we need to answer the following questions:

- What is a VC's rate?
- What is the input rate?
- Do the congestion control actions at a link affect the current loop or the previous loop?
- When is the VC's allocation at the link (VAL) calculated?

4.1 Measuring the Rate of the VC

Figure 7 shows four methods of estimating the VC's rate for use in the switch algorithm calculations. These four methods are:

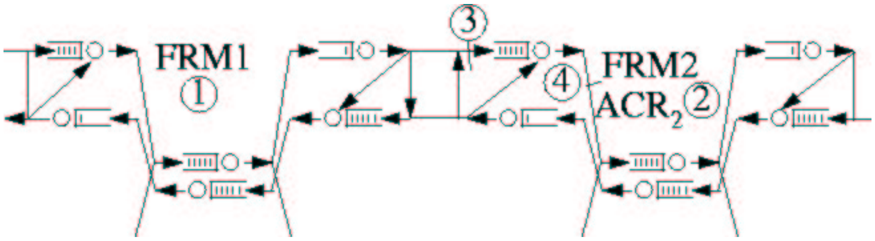


Figure 7: Four methods to measure the rate of a VC at the VS/VD switch

1. The rate declared in the current cell rate (CCR) field of FRMs in the previous loop (FRM1) is used as the VC's rate.
2. The rate declared in the CCR field of FRMs in the next loop (FRM2) is used as the VC's rate. This value is same as the ACR on the next loop (ACR_2).
3. The actual source rate in the previous loop can be measured. This rate is equal to the VC's input rate to the per-VC queue. This measured source rate can be used as the VC's rate.
4. The actual source rate in the next loop can be measured as the VC's input rate to the per-class queue (from the per-VC queue). This measured value can be used as the VC's rate.

Note that the actual source rates and ACRs of a flow are often very different. If a source does not have enough cells to send, its source rates may be below its ACR. This happens more often in a virtual source than in a real source.

4.2 Measuring the Input Rate at the Switch

Figure 8 shows two methods of estimating the input rate for use in the switch algorithm calculations. These two methods are:

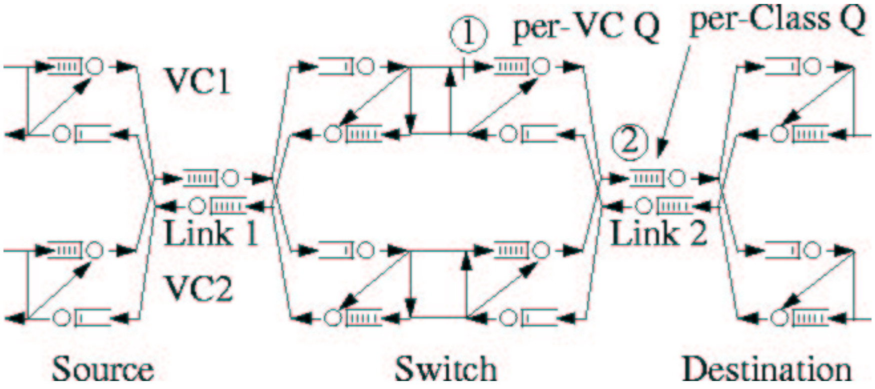


Figure 8: Two methods to measure the input rate at the VS/VD switch

1. The input rate is the sum of input rates to the per-VC queues.
2. The input rate is the input rate to the per-class queue.

4.3 Effect of Link Congestion on Neighboring Links

Congestion on a link can affect neighboring loops. For example, congestion of link2 can result in the following actions:

1. Change ER_1. This affects the rate of the previous loop only.
2. Change ACR_2. This affects the rate of the next loop only.
3. Change both ER_1 and ACR_2. This affects both the previous and the next loop. The next loop is affected instantaneously while the previous loop is affected after a feedback delay.

4.4 Frequency of Updating the Allocated Rate

The allocated rate is normally (in a non-VS/VD switch) calculated when a BRM cell is processed in the reverse direction in a switch or when an FRM is turned around in a destination. However, in a VS/VD switch, we have three options as shown in figure 9 and described below:

1. Calculate the allocated rate on receiving BRM2 only. Store the value in a table and use this table value when a FRM1 is turned around.
2. Calculate allocated rate only when FRM1 is turned around.
3. Calculate allocated rate both when FRM1 is turned around as well as when BRM2 is received. No table lookups are done.

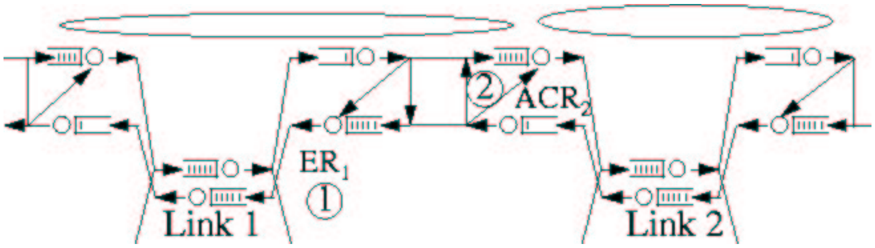


Figure 9: Three methods to update the allocated rate

4.5 Summary of VS/VD Switch Design Options

We have enumerated the various alternatives for each design issue. In summary, the options available are:

- What is the VC's rate: 4 options.
- What is the input rate: 2 options.
- Effect of link congestion: 3 options.
- Allocated rate update frequency: 3 options.

This gives a total of $4 \times 2 \times 3 \times 3$ or 72 combinations of options. Some of these combinations are incompatible and hence do not work. Hence, we identify the incompatible options and then conduct a performance evaluation of the compatible options.

5 Evaluation of VS/VD Switch Design Options

In this section, we first present analytical arguments to eliminate certain design combinations. We then evaluate the performance of the remaining (viable) design options through simulation.

5.1 VC Rate and Input Rate Measurement: Viable Options

5.1.1 VC Rate Measurement Techniques

When a source is given a new rate (via the ER field in BRMs), the effect will not be seen for some time since new feedback has to reach from the switch to the source and cells at the new rate have to come back to the switch. This delay from the switch to the source and from source to the switch is called the "feedback delay."

As discussed earlier, the rate of a VC can be read from CCR field of FRMs or measured. The measured rate is often lower than that in the CCR field. This difference has been a topic of several debates in the TM working group. To issues relating to use-it-or-lose-it (UILI) policies.

The actual VC's input rate can be measured either at the entry to the per-VC queues or at the entry to the per-class queues. The VC's rate to the per-VC queues can be measured either over an averaging interval, or between successive RM cells from the VC or a combination of both techniques. Exponential averaging [2] can be done to get a smoother distribution of rate values. We have not analysed the effect of measuring the source rate at the entry to the per-VC queues.

One easy way to measure VC rate is simply:

$$\text{Measured rate} = \text{Nrm/Inter-RM cell time}$$

The inter-RM cell time is known because it is measured as part of the source rules (SES Rule 5).

One can argue that the CCR field of RM cells should contain actual rate of the source and not allocated rate. This would make the rate allocation at the switch more precise. However, that topic is not a subject of this contribution.

5.1.2 Input Rate Measurement Techniques

As shown in figure 8, the input rate can be measured as the sum of the input rates of VCs to the per-VC queues or the aggregate input rate to the per-class queue. These two rates can be different because the input rate to the per-VC queues is at the previous loop's rate while the input to the per-class queue is related to the next loop's rate.

Figure 10 show a case where two adjacent loops run at very different rates (10 Mbps and 100Mbps) at least for one feedback delay.

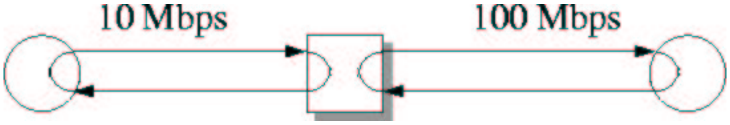


Figure 10: Two adjacent loops may operate at very different rates for one feedback delay

Given 4 options for VC's rate and two for input rate, there are 8 possible combinations as shown in Table 1. Note that we assume only one VC active in the network.

The table shows that four of these combinations may work satisfactorily. The other combinations use inconsistent information and hence may either overallocate rates leading to divergent queues or result in unnecessary oscillations.

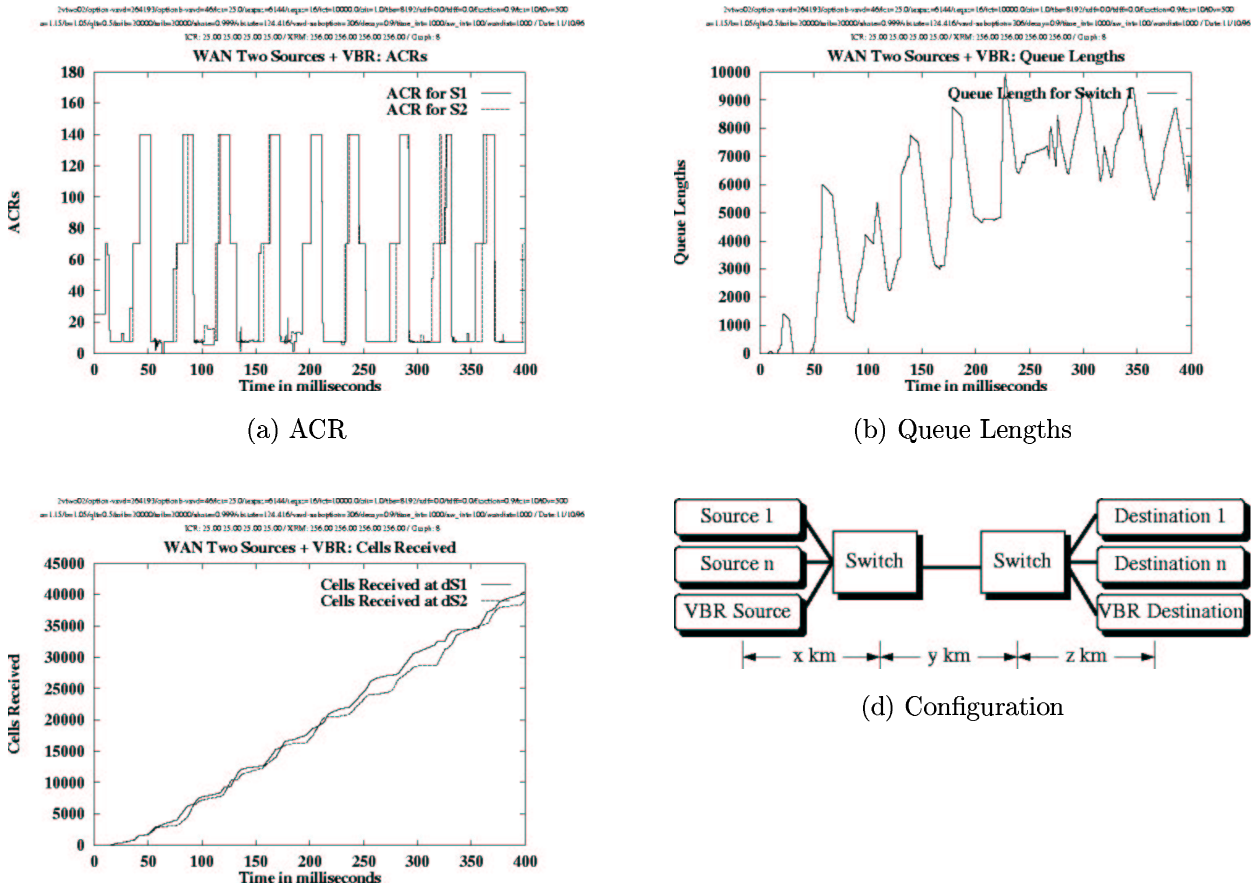
Further, the above table does not make any assumptions about the queue lengths at any of the queues. For example, when the queue lengths are close to zero, the actual source rate might be much lower than the declared rate in the FRMs leading to overallocation of rates.

Figure 11 shows simulation results with one such case (corresponds to entry 4 in table 1). The configuration used has two ABR infinite sources and one high priority VBR source contending

Table 1: Viable combinations of VC rate and input rate measurement

#	VC Rate Method	Σ VC rates (Mbps)	Input Rate Method	Input Rate Value	Design (OK/NO)
1.	From FRM1	10	Σ per-VC	10	OK
2.	From FRM1	10	per-class	10-100	NO
3.	From FRM2	100	Σ per-VC	10	NO
4.	From FRM2	100	per-class	100	OK
5.	At entry to per-VC queue	10	Σ per-VC	10	OK
6	At entry to per-VC queue	10	per-class	10-100	NO
7.	At entry to per-class queue	100	Σ per-VC	10	NO
8.	At entry to per-class queue	100	per-class	100	OK

for the bottleneck link's (LINK1) bandwidth. The VBR has an ON/OFF pattern, where it uses 80% of the link capacity when ON. The ON time and the OFF time are equal (20ms each). The VS/VD switch overallocates rates when the VBR source is OFF. This leads to ABR queue backlogs when the VBR source comes ON in the next cycle. The queue backlogs are never cleared, and hence the queues diverge. In this case, the fast response of VS/VD is detrimental because the rates are overallocated.



(a) ACR

(b) Queue Lengths

(c) Cells Received

(d) Configuration

Figure 11: 2-Source+VBR Configuration: Rejected Option = 306

We have not yet evaluated entry 5 of the table (measurement of VC rate at entry to the per-VC queues). Hence, out of the total of 8 combinations, we have two viable combinations: entry 1 and entry 8 of the table.

5.2 Link Congestion and Allocated Rate Update Frequency: Viable Options

In Section 4.3, we listed three options for link congestion. Of these, the second option ("next loop only") does not work because the congestion information is not propogated back to the sources of the congestion. This leaves two alternatives. The last option ("both loops") is attractive because, when ACR_2 is updated, the switches in the next loop experience the load change faster, and save a few iterations of the switch algorithm. This reduces the convergence time in such configurations.

The allocated rate update has three options: either update upon receiving BRM in VS and enter the value in a table to be used when an FRM is turned around, update upon turning around FRM at VD, or update both at FRM (VD) and at BRM (VS) without use of a table. The last option recomputes the allocated rate a larger number of times, but can potentially allocate rates better because we do not use old information.

The allocated rate update and the link congestion effects interact as shown in figure 12. The figure shows a tree where the first level considers the link congestion (2 options), i.e., whether the next loop is affected or not. The second level lists the three options for the allocated rate update frequency. The viable options are those highlighted in bold at the leaf level.

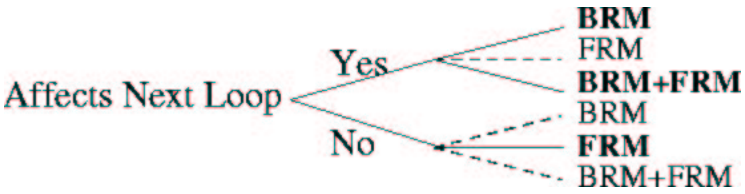


Figure 12: Link congestion and Allocated Rate Update: Viable Options

In particular, if the link congestion does not affect the next loop, the allocated rate update at the FRM turnaround is all that is required. The allocated rate at the BRM is redundant in this case.

Further, if the link congestion affects the next loop, then the allocated rate update has to be done on receiving a BRM, so that ACR can be changed at the VS. This gives us two possibilities as shown in the figure (BRM only, and BRM+FRM).

Hence, we have three viable combinations of link congestion and the allocated rate update frequency.

5.3 Summary of Viable VS/VD Options

The total (reduced) number of viable options is six which is a combination of:

- VC's rate and Input Rate: 2 alternatives
- Link congestion and update frequency: 3 alternatives

We list these alternatives in table 2.

Table 2: Summary of viable VS/VD design alternatives

Code #	VC Rate Method	Input Rate Measured at Queue	Link Congestion Effect	Allocated Rate Updated at
41	Read from FRM1	per-VC	previous loop only	FRM1 only
52	Measured at per-class	per-class	both loops	FRM1 only
329	Read from FRM1	per-VC	both loops	FRM1 only
340	Measured at per-class	per-class	both loops	FRM1 and BRM2
393	Read from FRM1	per-VC	both loops	BRM2 only
404	Measured at per-class	per-class	both loops	BRM2 only

6 Simulation Results

We use four metrics to evaluate the performance of these key alternatives:

- **Response Time:** is the time taken to reach near optimal behavior on startup.
- **Convergence Time:** is the time taken for oscillations to diminish.
- **Throughput:** Total data transferred per unit time.
- **Maximum Queue:** The peak queue length reached before convergence.

The difference between response time and convergence is illustrated in Figure 13.

6.1 Response Time

Without VS/VD all response times are close to the round-trip delay. With VS/VD, the response times are close to the feedback delay from the bottleneck. Since VS/VD reduces the response time during the first round trip, it is good for long delay paths. The quick response time (10 ms in the parking lot configuration which has a 30 ms round trip time) is shown in figure 14.

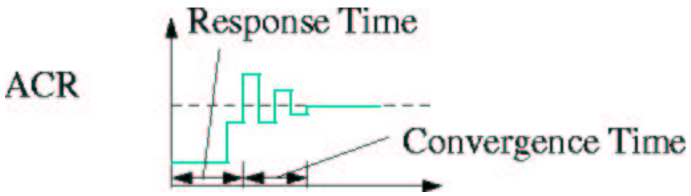
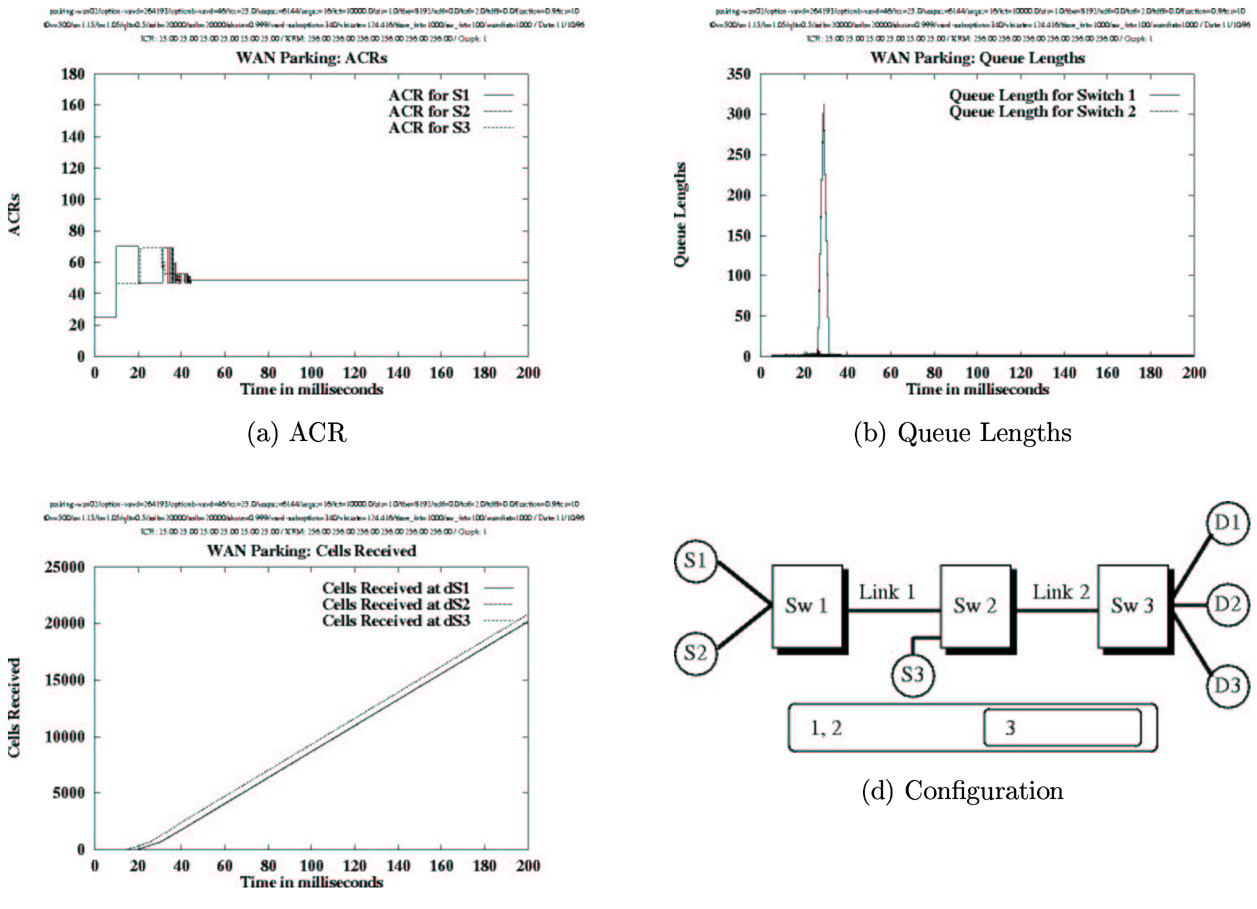


Figure 13: Response Time vs Convergence Time



(a) ACR

(b) Queue Lengths

(c) Cells Received

(d) Configuration

Figure 14: Parking Lot Configuration: Option = 340

Response time is also important for bursty traffic which “starts up” at the beginning of every active period after the corresponding idle period.

6.2 Cells Received at the Destination

The number of cells received at the destination is a measure of the throughput achieved. These values are given in table 3. The top row is a list of the configuration codes (these codes are explained in table 2). In addition, the final column lists the throughput values for the case when a non-VS/VD switch is used. In the 2 Src+VBR and upstream configurations, the simulation was run for 400 ms (the destination receives data from time = 15 ms through 400 ms). In the parking lot configuration, the simulation was run for 200 ms. The upstream configuration is shown in figure 15.

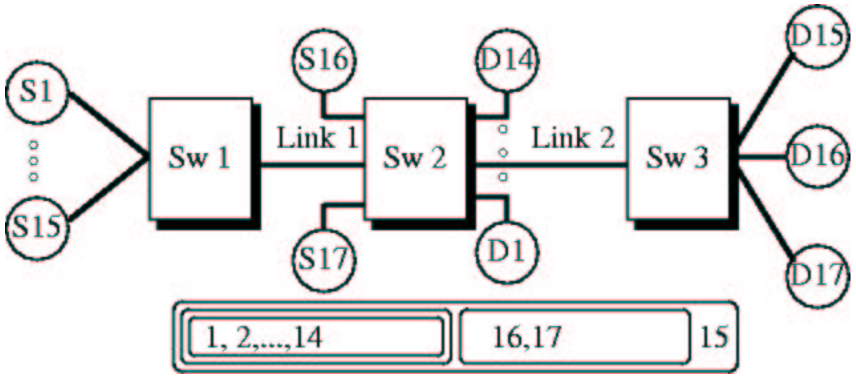


Figure 15: Upstream configuration

Table 3: Cells Received at Destination per source (in kcells)

VSVD Option #	329	393	52	340	41	404	No VS/VD
Configuration							
2 Src + VBR	31	31	32.5	34	32	33	30
Parking	22	22	23	20.5	23	20.5	19.5
Upstream	61	61	61	60	61	61	62

In general, there is little difference between the alternatives in terms of throughput. However, there is a slight increase in throughput when VS/VD is used over the case without VS/VD switch.

6.3 Convergence Time

The convergence time is a measure of how fast the system reaches steady state. It is also sometimes called “transient response.” The convergence times of various options are listed in table 4. The transient configuration is shown in figure 16. In this figure, the second source is the transient source that is active for only a part of the simulation time.

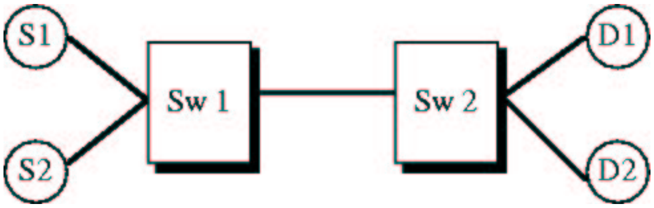


Figure 16: Transient configuration

Table 4: Convergence Time in ms

VSVD Option # Configuration	329	393	52	340	41	404	No VS/VD
Transient	50	50	65	**20**	55	25	60
Parking	120	100	170	**45**	125	50	140
Upstream	95	75	75	**20**	95	**20**	70

The convergence time of VS/VD option 340 is the best. Recall that this configuration corresponds to measuring the VC rate at the entry to the per-class queue, input rate measured at the per-class queue, link congestion affecting both the next loop and the previous loop, the allocated rate updated at both the FRM1 and BRM2.

6.4 Maximum Transient Queues

The maximum transient queues give a measure of how askew the allocations were when compared to the optimal allocation and how soon this was corrected. The maximum transient queues are tabulated for various configurations for each VS/VD option and for the case without VS/VD in table 5.

Table 5: Maximum queue in kcells

VSVD Option # Configuration	329	393	52	340	41	404	No VS/VD
2 Src + VBR	**1.2**	1.4	2.7	1.8	2.7	1.8	2.7
Transient	1.4	1.1	1.4	**0.025**	1.3	1.0	6.0
Parking	1.9	1.9	1.4	**0.3**	3.7	0.35	2.0
Upstream	0.025	0.08	0.3	**0.005**	1.3	**0.005**	0.19

The above table shows that VS/VD option 340 has very small transient queues in all the configurations and the minimum queues in a majority of cases. This result, combined with the fastest response and near maximum throughput behavior affirms the choice of option 340 as the best VS/VD implementation.

Observe that the queues for the VS/VD implementations are in general less than or equal to the queues for the case without VS/VD. The queues are very small if the correct implementation (like option 340) is chosen.

6.5 Other Observations

Due to the improved convergence time in VS/VD, some cases that diverged with basic ERICA converge with VS/VD.

7 Conclusions

In summary, we conclude that:

- VS/VD can be added to switches which implement per-VC queueing at little cost. The addition can potentially yield improved performance in terms of response time, convergence time, and smaller queues.

- VS/VD reduces the response time during the first round trip. Hence VS/VD is good for long delay paths

- VS/VD does improve convergence time. Some cases that diverged with basic ERICA converge with VS/VD.

- VS/VD results in a slight increase in throughput due to reduced response time and reduced convergence time.

- The effect of VS/VD depends upon the switch algorithm used and how it is implemented in the VS/VD switch. The convergence time and transient queues can be very different for different VS/VD implementations of the same basic switch algorithm.

- In VS/VD, ACR and actual rates of VCs are very different. The switch cannot rely on the CCR field. Measuring each VC's rate at the switch is highly desirable.

- The VC's rate should be measured at the output of the VS. We have not analyzed the implementations where the measurement is at the input of the VS.

- The sum of the input-rates to per-VC queues at a VS is not the same as the input rate to the link.

- Input rate to the link must be measured as the input rate to the per-class queue.

- On detecting link congestion, the congestion information should be forwarded to the previous loop as well as the next loop. This is done by including the link bottleneck information in the per-VC ACR at each VS. This method reduces the convergence time by reducing the number of iterations required in the switch algorithms on the current and downstream switches.
- It is best for the the rate allocated to a VC to be calculated both when turning around FRMs at the VD as well as after receiving BRMs at the next VS.

8 Future Work

The VS/VD provision in the ABR traffic management framework can potentially improve performance of bursty traffic and reduce the buffer requirements in switches. The VS/VD mechanism achieves this by breaking up a large ABR loop into smaller ABR loops which are separately controlled. However, further study is required in the following areas:

- Effect of VS/VD on buffer requirements in the switch.
- Quantitative effect of the reduced convergence time.
- Effect of different control loop lengths
- Advantage of VS/VD within a carrier network vs an ER-only network
- Scheduling issues with VS/VD.
- Effect of different switch algorithms in different control loops.
- Effect of non-ABR clouds and the standardization issues involved.
- Information to be transferred through a non-ABR cloud for interoperability.

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] Raj Jain, Shiv Kalyanaraman, Rohit Goyal, Sonia Fahmy, and Ram Viswanathan, "ERICA Switch Algorithm: A Complete Description," *AF-TM 96-1172*, August 1996, http://www.cis.ohio-state.edu/~jain/atmf/atm96-1172.ps

[3] Raj Jain, Sonia Fahmy, Shivkumar Kalyanaraman, and Rohit Goyal, "ABR Switch Algorithm Testing: A Case Study with ERICA," ATM Forum/96-1267, October 1996, http://www.cis.ohio-state.edu/~jain/atmf/atm96-1267.ps