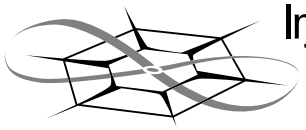
The University of Kansas



Information and
Telecommunication
Technology Center

Technical Report

Performance Evaluation of an Edge Switch ABR Algorithm

Haritha Pindi, Nail Akar, Victor Frost

ITTC-FY97-TR-13200-01

June/July 1997

Project Sponsor:
Sprint Corporation

Abstract

Edge switches form the exterior of the Edge-Core Network. The series-D ER algorithm is an Explicit Rate ABR congestion control algorithm developed by Nortel/Fore systems to eb implemented in the edge switches of the Edge-Core Network. This simualtion study was conducted to evaluate the prformance of the algorithm in terms of the fairness achieved, throughput/link utilization and the queue sizes of the switches for a variety of netwrok environments that are likely to occur in practice. This work is part of the ongoing work to evaluate the interoperability and performance of the ABR flow control algorithms that Sprint expects to use in its Edge-Core ATM network (i.e algorithms from NEC and FORE/Nortel). Although the Edge-Coe architecture calls for the use of different switches for the Edge and for the Core, we felt that it is useful to observe the behavior of each of the algorithm in isolation in order to better understand their behavior in heterogenous test cases. In this simulation study we considered the FORE/Nortel's Series_D algorithm.

Sponsored by Sprint Corporation
Project # 13200

Contents

1 Introduction

This report outlines a series of validation and performance tests run for the evaluation of the Series -D ER algorithm. The simulation model was developed in OPNET by Nortel, the manufacturers of the Vector switch in which the algorithm is going to be implemented. The initial models were modified and revised in order to do both the validation and the performance tests.

The validation tests were conducted initially to verify the operation of the models and to gain insight into the various parameters associated with the functionality of the algorithm.

Based on the initial results a series of performance tests were defined and executed where the algorithm was tested in different network configurations. The goal of the tests was to assess the algorithm's performance under a variety of network topologies that were likely to occur in practice. The long term goal of this simulation study is to analyze the performance and interoperability of the algorithm together with other algorithms (NEC algorithm) in Sprint's Edge/Core Network architecture.

The report is organized in the following order. Section 2 gives an overview of the algorithm, Section 3 describes the simulation model, Section 4 describes the parameters used in the simulation, Section 5 gives a description of the performance metrics used to evaluate the algorithm, Section 6 describes the various network topologies used for the simulation and the results of the simulation. The report ends with a conclusion and some ideas for future work.

2 Overview of the Algorithm

The series-D ER algorithm is an Explicit Rate ABR congestion control algorithm developed by FORE/NORTEL. The ABR buffer at each port of a switch can be partitioned into M regions using M-1 buffer thresholds, M being a design parameter. The objective of this flow control scheme is to maintain the ABR buffer occupancy within a predetermined "middle" region or the "operating region". If the buffer occupancy falls below this region, the control scheme will try to push the buffer occupancy upward by allowing faster input rate, and similarly if the buffer occupancy lies above the region, the scheme will push it downward by reducing the input rate. Each region can be in one of these five modes: the constant increase, default increase, normal mode, default decrease or constant decrease mode. The capability to select the mode is provided by software. Based on the queue region and the rate of change of the queue length the following two parameters are computed periodically every update interval of N cells.

1. MAIR = Additive Increase Rate
2. MRF = Multiplicative Reduction Factor

A Mean Allowed Cell Rate (MACR) which is a per-port parameter is computed based on the MAIR and the MRF values. The scheme keeps track of the number of FRM cells received for each port to decide when the values of MACR can be increased. The ER computation is based on the MACR value and is done whenever a backward RM cell is received. ER takes into account the per-VC buffer occupancy. The pseudo code is given in [1]. A specific case in which region 1 is constant increase region, region 2 in the default increase mode, region 3 in the default decrease mode and region 5 in the constant decrease mode has been selected (figure 1).

3 Simulation Model

The key components of the various simulation set ups are the Traffic Source model, the Switch model and the Destination model. The models have been developed in OPNET[2], a communication network modeling and simulation package.

Source Model: The ABR source is modeled as a persistent source and it's behavior is exactly as specified for the ABR Source behavior by the ATM Forum[3]. The inter-cell times is a constant and is determined by the allowed cell rate of the source. The source will initially send an RM cell and then Data cell. An RM cell is sent for every 31 data cells i.e, every 32nd cell is an RM cell. The source has been modeled in a way to take the number of users(sources) as a parameter and will accordingly invoke a child process for each source to be activated. The start time is also provided as a parameter to enable staggered connections of sources.

Switch Model: The ABR algorithm is implemented here. The switch uses per VC queues and Round Robin servicing for these queues. The model does not account for the detailed switch architecture. Only the ABR buffer has been modeled.

Destination Model: The destination model discards the cell if it is a data cell and turns around the RM cells. The destination model's behavior is based on the specifications by the ATM Forum[3].

4 Simulation Parameters & Assumptions

We have used the following parameters for our simulations and these are in accordance with the recommendations from Nortel.

Table 1: Source Parameters

Parameter	Expansion	Value
ICR	Initial Cell Rate	7064.14 cells/s=3 Mbps
MCR	Minimum Cell Rate	0
PCR	Peak Cell Rate	353207cells/s = 150 Mbps
TCR	Tagged Cell Rate	10 cells/s
RIF	Rate Increase Factor	1.0
RDF	Rate Decrease Factor	0.0625
ADTF	ACR Decrease Time Factor	0.5 s
CDF	Cutoff Decrease Factor	0.125
FRTT	Fixed Round Trip Time	8000 μs
CRM	Missing RM cell count	32000
Mrm	Controls bandwidth allocation between FRM, BRM and data cells	2 cells
Nrm	Number of cells between FRM cells	32 cells
Trm	Upper bound on Inter-FRM Time	100 ms
TBE	Transient Buffer Exposure (determines the maximum number of cells that may suddenly appear at the switch during the first round trip before the closed loop phase of the control takes effect).	16777215 cells

- **Source Parameters** Table 1 shows the source parameters used in the simulations.

  Note that we have used only homogeneous sources i.e, all the sources have the same set of values for the parameters associated with them. Also note that the value chosen for the paramter CRM did not match with the value calculated using CRM=TBE/Nrm. This was not done intentionally. However, since we had assumed on our simulations that the backward RM cells were not encountering any congestion and also the delays chosen were also not long enough to trigger the rule 6 of the source behavior [3]. So this had no effect

Table 2: Queue Region Thresholds

Parameter	Value
Q1	50 cells
Q2	300 cells
Q3	1000 cells
Q4	1500 cells

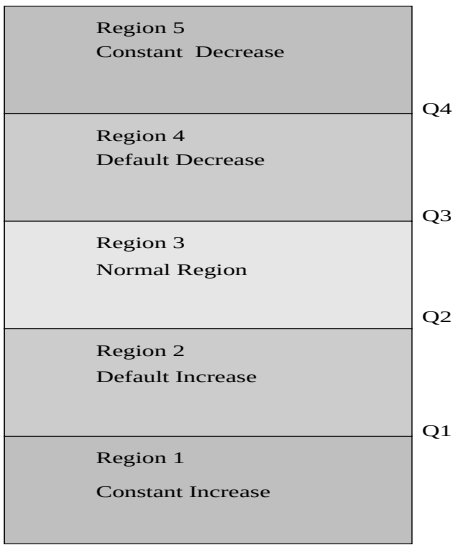


Figure 1: ABR buffer

on the results.

- **Switch Parameters** Tables 2, 3, 4 and 5 give the switch parameters used in our simulations.

  We have assumed in our simulations that there is congestion only in the forward direction. The backward RM cells were assumed not to be encountering any congestion.

Table 3: Common Switch Parameters for both LAN and WAN

Parameter	Expansion	Value
TR	Target Rate	150Mbps
N	Update Interval	32
PFA	Per-VC Fixed ACR	15.5Mbps
MACR0	Initial value of MACR	147.25Mbps
AVF	Average Factor	1/16
max_queue_size	-	infinity

Table 4: Switch Parameters for LAN

Parameter	Value
min_MRF	0.375
default_MRF	0.8
max_MAIR	1 Mbps
const_MAIR	0.5Mbps
default_MAIR	0.2Mbps
scale_MRF3	2
scale_MRF4	4
scale_MRF5	8
scale_MAIR2	64
scale_MAIR3	32
scale_MAIR4	4
q_{vc}(low)- Per-VC low threshold	5
q_{vc}(med)- Per-VC medium threshold	150
q_{vc}(high)- Per-VC high threshold	300
IFA(Increasing Factor for ACR)	1.5

Table 5: Switch Parameters for WAN

Parameter	Value
min_MRF	0.5
default_MRF	0.75
max_MAIR	0.5 Mbps
const_MAIR	0.25 Mbps
default_MAIR	0.1 Mbps
scale_MRF3	1
scale_MRF4	2
scale_MRF5	4
scale_MAIR2	16
scale_MAIR3	8
scale_MAIR4	2
q_{vc}(low)	20
q_{vc}(med)	150
q_{vc}(high)	300
IFA	1.25

5 Performance Metrics

The performance metrics considered in this study include the following

- **Fairness**: Intuitively fairness means that the bandwidth of the link be equally shared among all the sources. According to the max-min fairness definition[4], a bandwidth allocation is max-min fair if for every VC, one cannot increase its bandwidth without decreasing the bandwidth of VCs of equal or lower bandwidth. With this definition, all VCs bottle-necked at a given link get equal portions of its available bandwidth. Let C denote the link capacity, let n denote the number of VCs traversing that link. Let k, where $k \leq n$, denote the number of VCs bottle-necked at some other link in the network. Let B denote the sum of the bandwidths used by the k VCs bottle-necked elsewhere. Then

the fair-share (max-min fair allocation) for VCs bottle-necked at the given link is:

$$\frac{C - B}{n - k}$$

The source ACR plots are representative of the fairness achieved by the algorithm.

- **Throughput/Utilization**: Throughput represents the amount of bandwidth on a link that is actually used. The throughput of a particular VC refers to the average bandwidth received by that VC. Link utilization is the total throughout divided by the link capacity and is a measure of the percentage of the link bandwidth utilized. The link utilizations of the bottleneck links have been plotted.

- **Queue Sizes** : The ABR buffer size is an important performance metric for the algorithm under study as the sole aim of the algorithm is to maintain the buffer in a predetermined normal region. The queue sizes of the bottleneck switches are presented under the Results section for the various configurations considered.

6 Test Configurations and Results

This section describes our efforts in analyzing the performance of the algorithm in three different network environments. A network environment is defined by the topology, length of the feedback delays and the traffic models.

The algorithm's performance was tested in networks with single complete [1] bottleneck and in networks with multiple bottlenecks, some partial [2] and some complete for both the LAN and WAN scenarios. Only ABR traffic was used during this phase of the study.

6.1 Two Node Configuration (Single Complete Bottleneck Topology)

Figure 2 shows the simplest topology for testing the behavior of the algorithm in complete bottleneck state. Here Link A is the bottleneck and the congestion control algorithm under test runs in switch 1. We turned off the congestion control algorithm in switch 2, making it a simple de-multiplexer rather than an actual switch. The simulation time for all of the following cases in this configuration is 0.3 seconds.

6.1.1 Case 1 : LAN (all links are 10 km), all sources start at t = 0 i.e RTT = 0.3 ms

Figures 3 to 6 show the plots for the source ACR's, MACR values computed in the switch, the queue sizes and the link utilizations respectively. Since there are 10 sources, the fair-share for each of the sources is 10% of the link capacity which is approximately 15 Mbps. The ACR plot shows that all of the sources attain a steady value of 15Mbps demonstrating that the algorithm works fair enough for this configuration. The MACR plot is plotted as percentage of the target rate. The MACR plots indicate that the MACR values approach 10% of the target rate which

[1] A link is said to be a complete bottleneck if it is a bottleneck for all of the VCs passing through it.

[2] A link is said to be a partial bottleneck if it is a bottleneck for only a subset of the VCs passing through it.

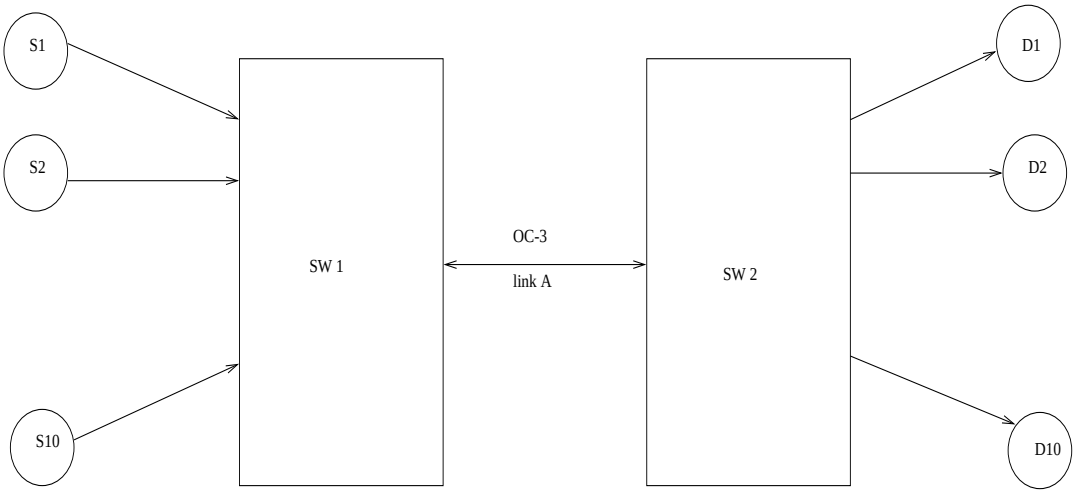


Figure 2: Two Node Configuration

is as expected for this case with 10 sources. The queue size plot shows that the ABR buffer attains a steady state value of approximately 300 cells which falls in the normal region (300 to 1000 cells) set for our simulations. So we see that the goal of the algorithm to maintain the ABR buffer in the normal region is satisfied in this case. The link is utilized to 100%.

6.1.2 Case 2 : LAN (all links are 10 km i.e RTT = 0.3ms), Staggered Connections.

In this case, the sources are started at 5ms apart starting from source 1 at time t=0. Source 4 to Source 8 send only 2 Mb of data and then leave the system. Figures 7 through 10 show the plots for this case. This case is used to test how fast the algorithm responds to a change in the available bandwidth and allocate fair shares of the available bandwidth to the sources. This case more closely represents a practical situation where the sources come and go. From the source ACR plots (Figure 6)

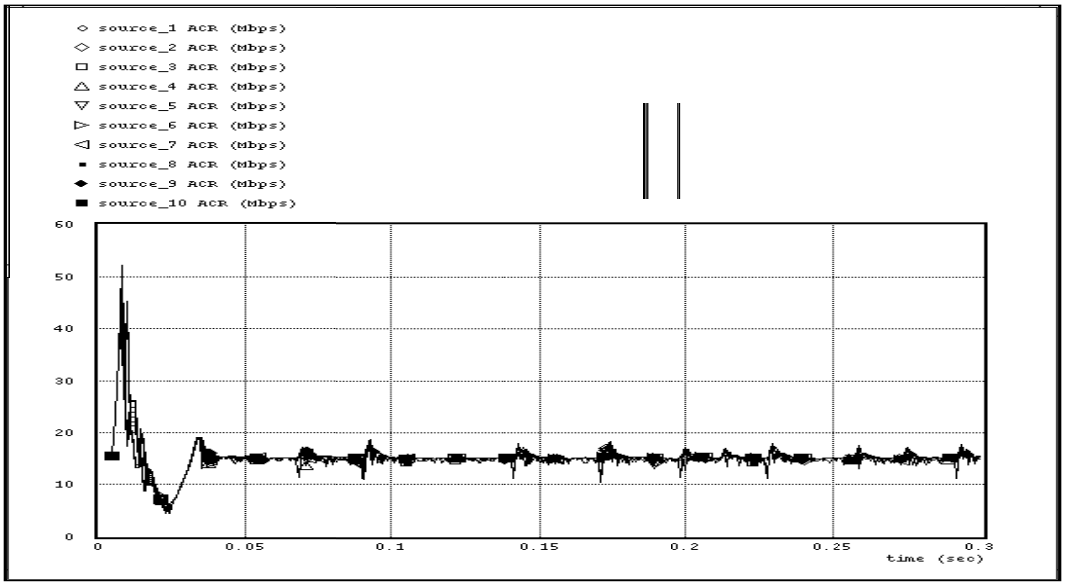


Figure 3: Two Node Configuration Case 1 : Source ACR plots

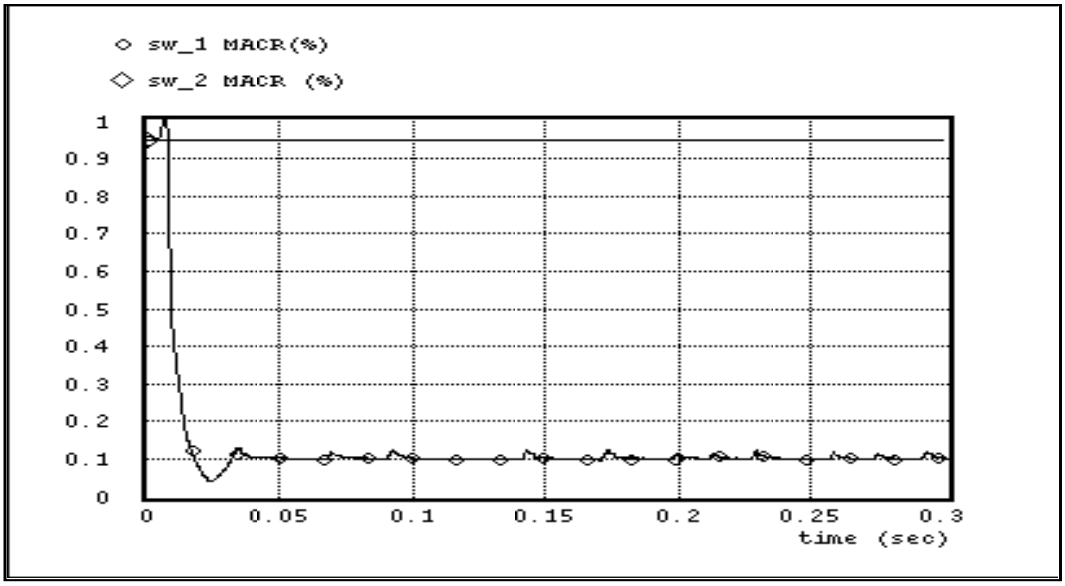


Figure 4: Two Node Configuration Case 1: Switch MACR plots

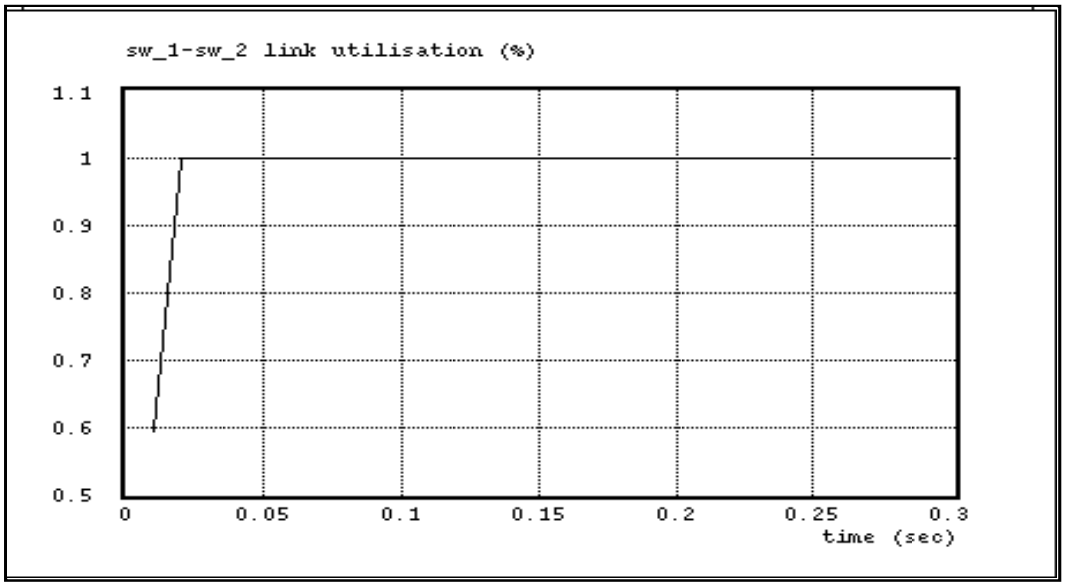


Figure 5: Two Node Configuration Case 1: Link Utilization plots

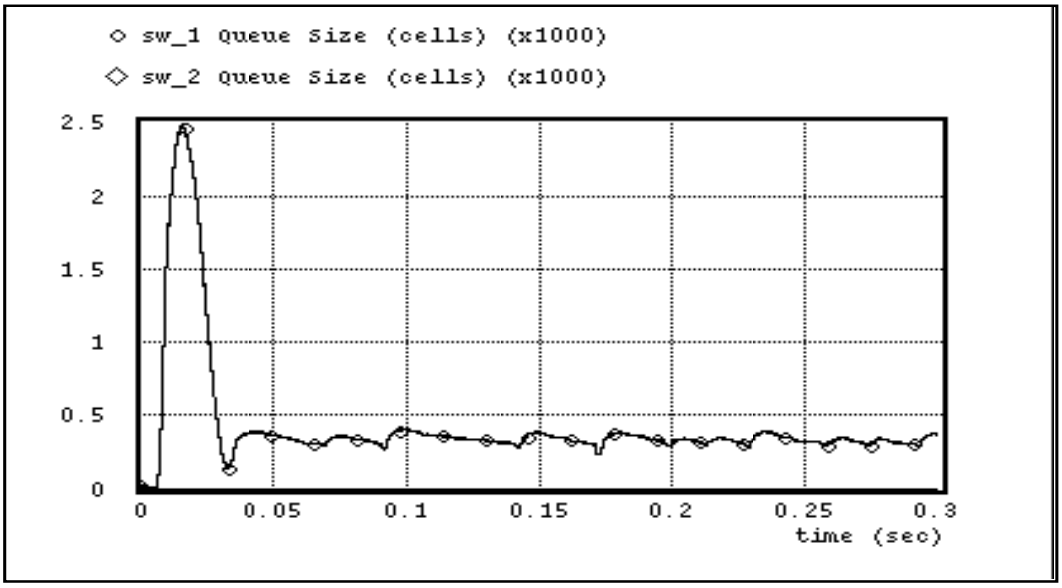


Figure 6: Two Node Configuration Case 1: Queue Size plots

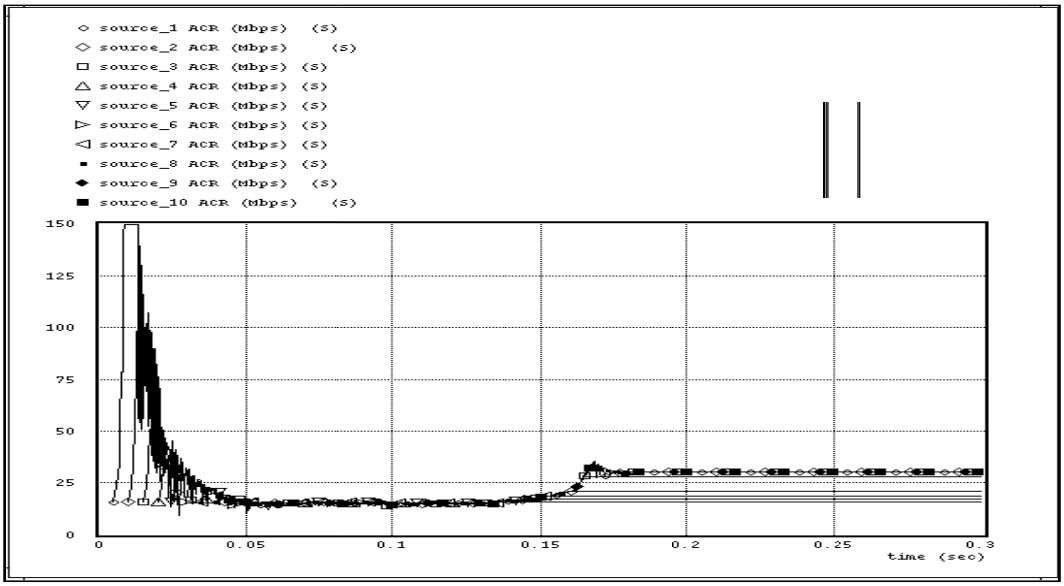


Figure 7: Two Node Configuration Case 2: Source ACR plot

we see that when all the 10 sources are active each of the sources gets a share of 15 Mbps which is as expected (10% of the link capacity). After a while sources 4 to 8 are deactivated, so the available capacity increases and the remaining sources are now receiving 20% of the link capacity(30 Mbps). The algorithm is therefore responding fast enough to make a reallocation of the bandwidths. The MACR plot also shows similar characteristics. The Queue size plot in this case also shows that the ABR buffer is well maintained in the normal region. The sudden fall in the queue size at 0.15 seconds is because of some of the sources leaving the system at that time. However note that in this case the time taken to reach a steady value is more than in the previous case which is obvious as 5 of the sources are leaving the system causing a temporary instability. The link utilization is seen to be 100% in this case also.

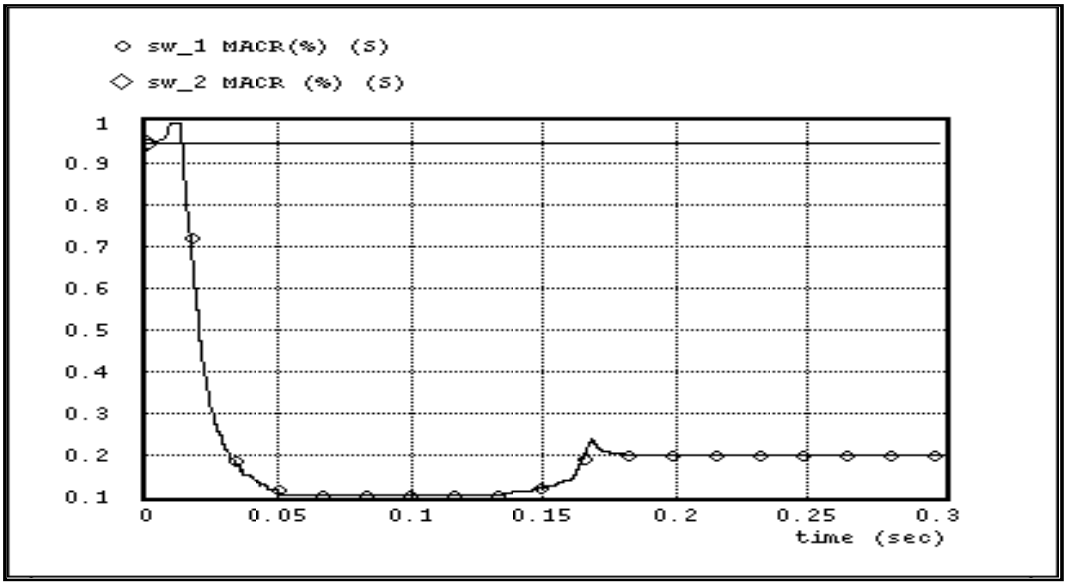


Figure 8: Two Node Configuration Case 2: Switch MACR plots

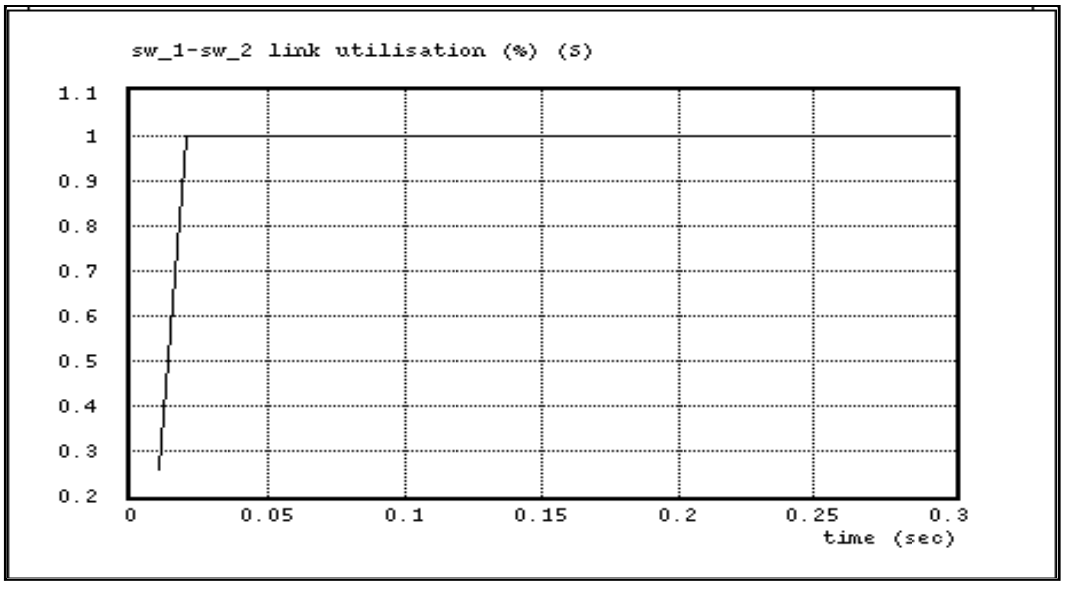


Figure 9: Two Node Configuration Case 2: Link Utilization plots

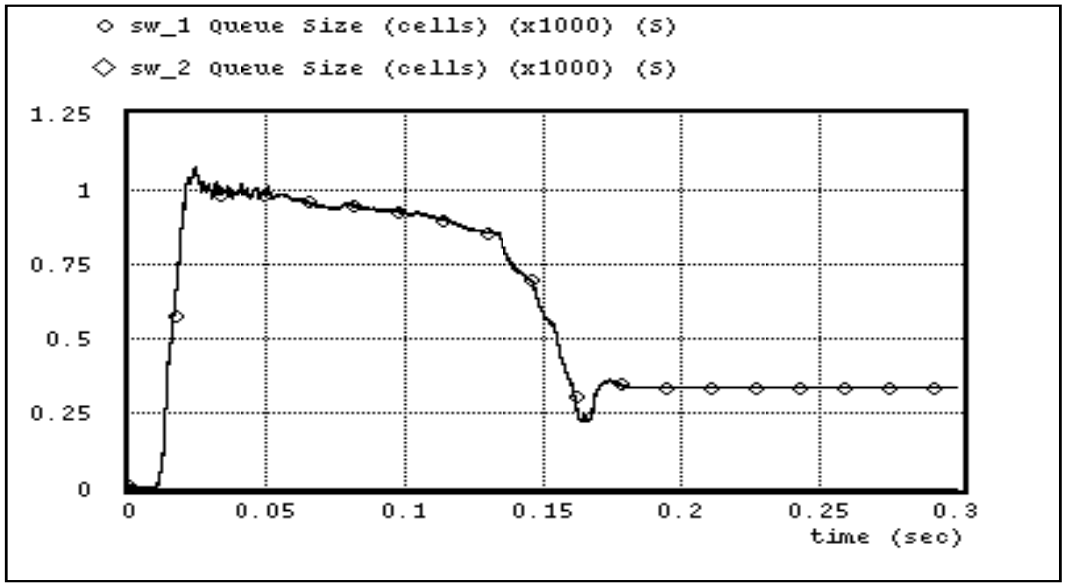


Figure 10: Two Node Configuration Case 2: Queue Size plots

6.1.3 Case 3 : LAN same as case 2 except that the algorithm's parameter Scale_MRF3 = 0.5

The plots for this case are shown in figures 11 through 14. The plots are similar to the previous case. This shows that the algorithm is not very sensitive to the parameter Scale_MRF3[1].

6.1.4 Case 4 : WAN Scenario. Inter-Switch Distance = 1000 km . Source-Switch Distance = 250km i.e RTT = 1.5ms, All sources start at t=0

Figures 15 through 18 show the plots for this case. Note that the queue sizes (figure 17) and the ACR plots (figure 14) exhibit a rather oscillatory behavior and it takes a longer time to reach a steady state. This is because of the increase in the feedback delay.

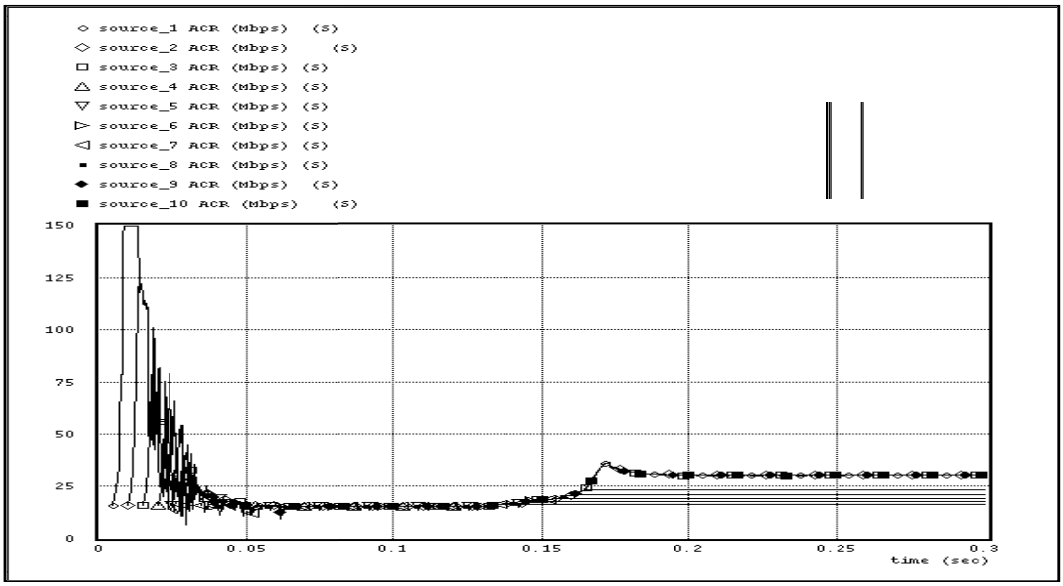


Figure 11: Two Node Configuration Case 3: Source ACR plot

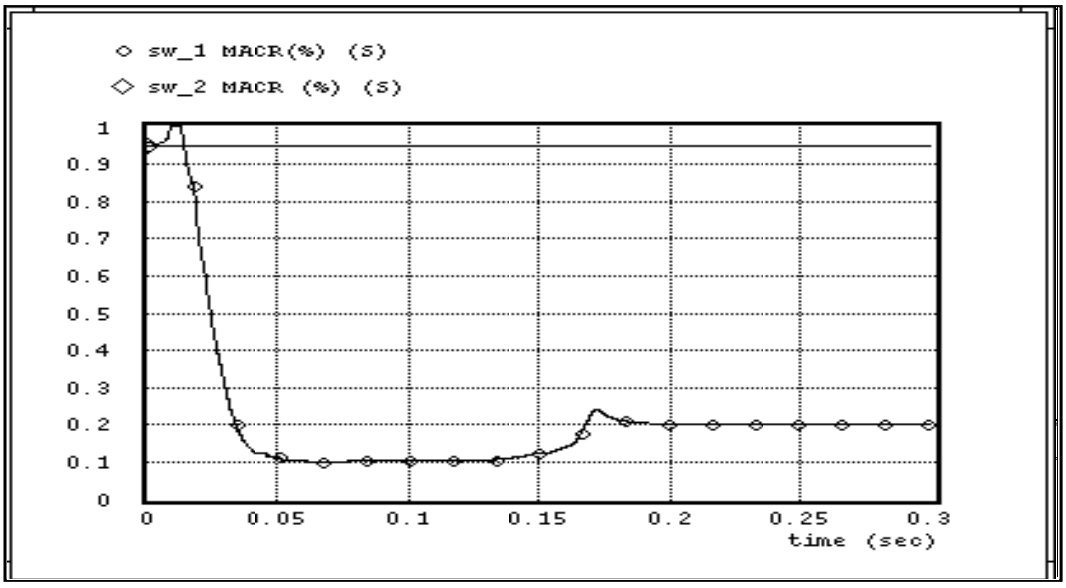


Figure 12: Two Node Configuration Case 3: Switch MACR plots

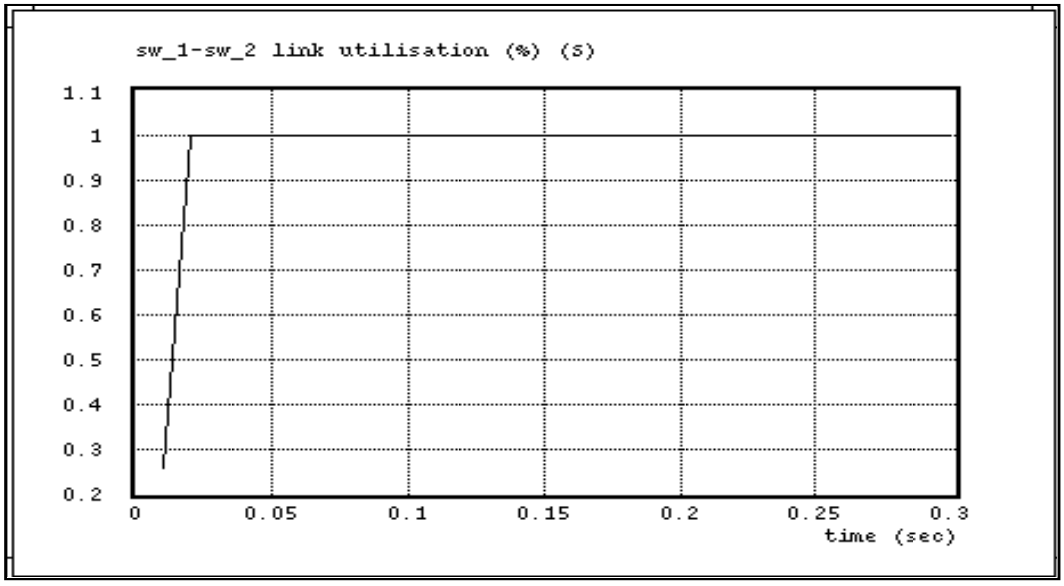


Figure 13: Two Node Configuration Case 3: Link Utilization plots

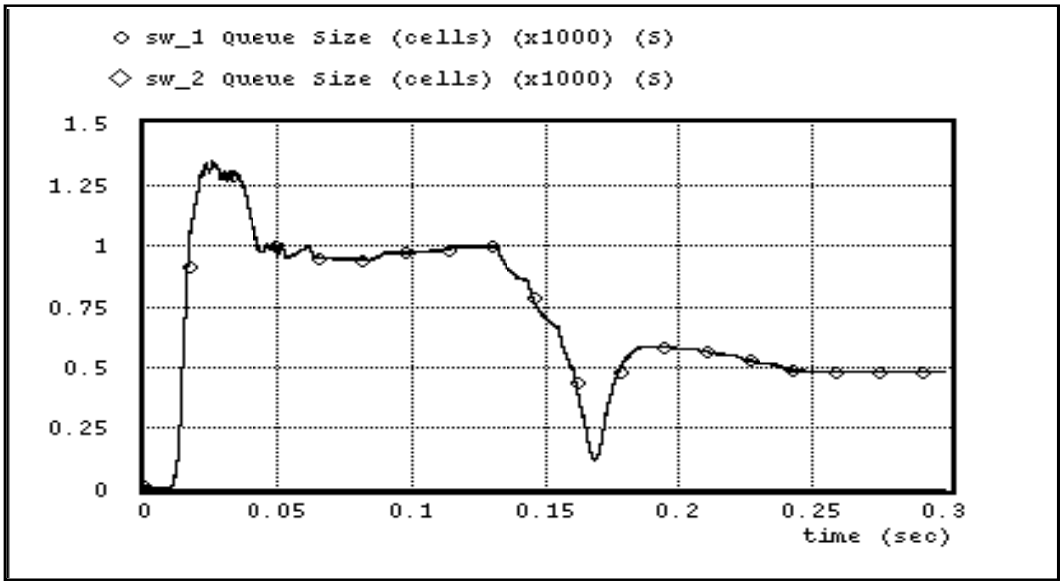


Figure 14: Two Node Configuration Case 3: Queue Size plots

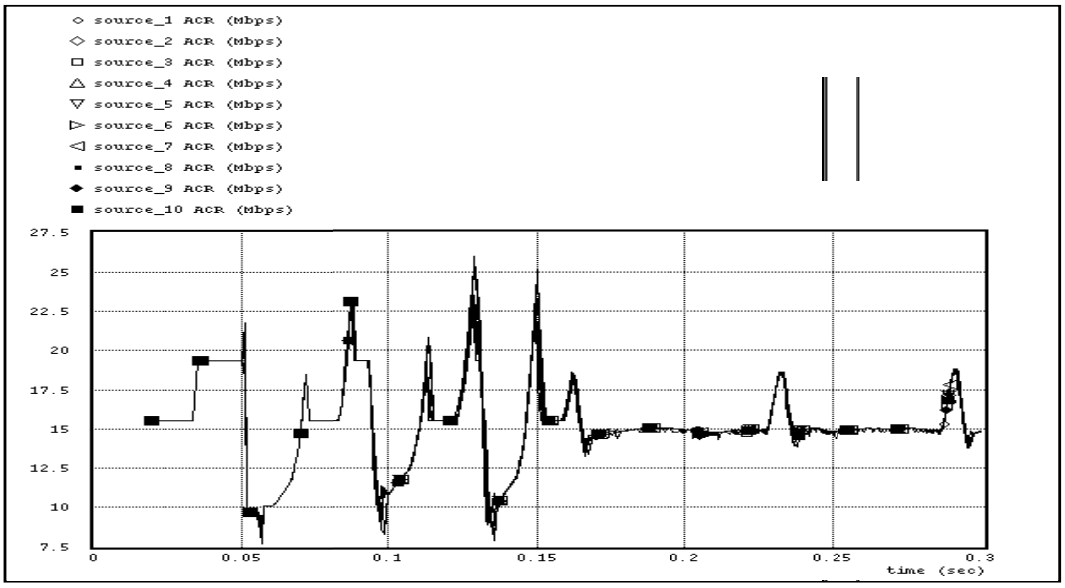


Figure 15: Two Node Configuration Case 4: Source ACR plot

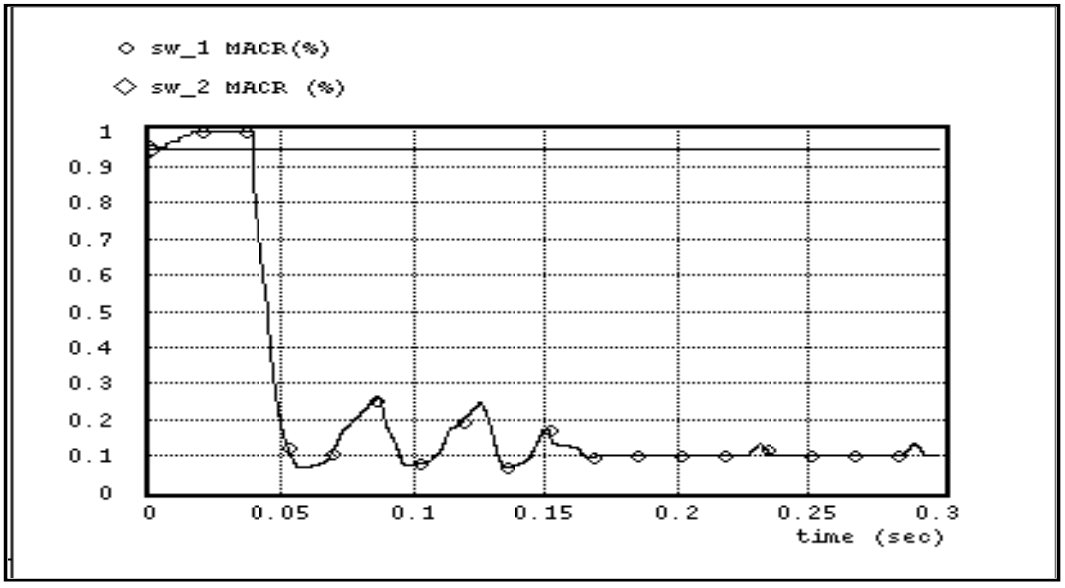


Figure 16: Two Node Configuration Case 4: Switch MACR plots

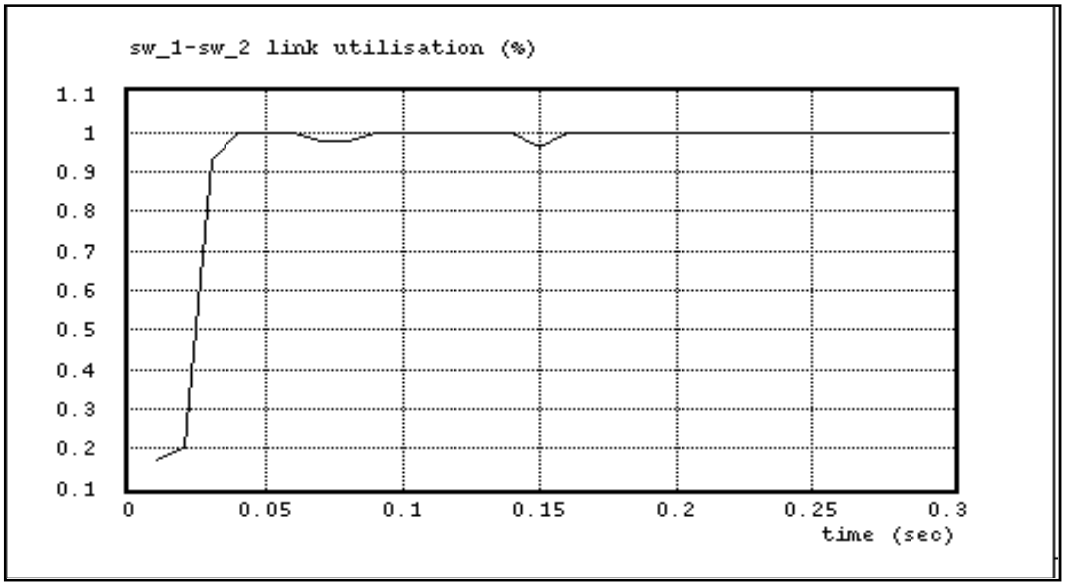


Figure 17: Two Node Configuration Case 4: Link Utilization plots

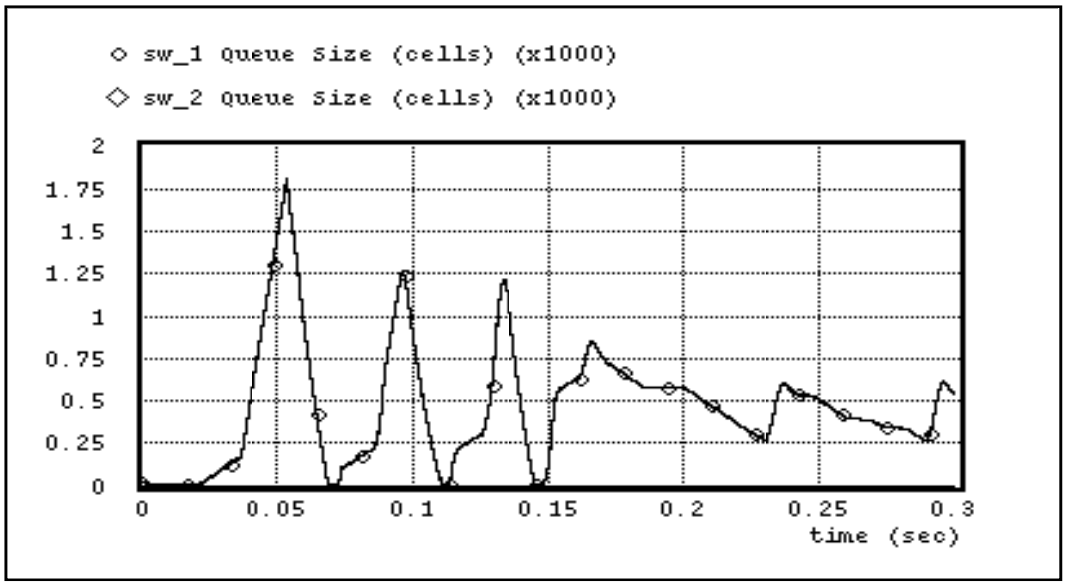


Figure 18: Two Node Configuration Case 4: Queue Size plots

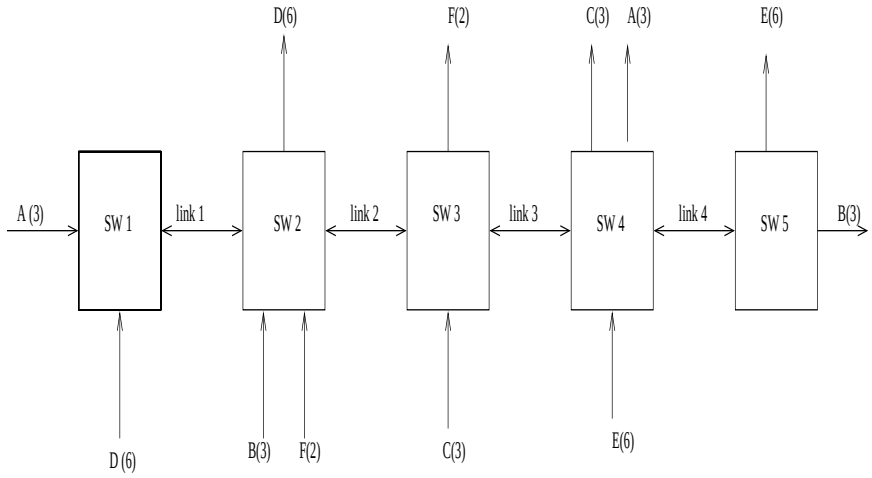


Figure 19: Generic Fairness Configuration

6.2 Generic Fairness Configuration (Multiple Bottleneck Topology)

This configuration shown in figure 19 is useful in testing the performance of the algorithm in the presence of multiple bottlenecks, some being partial and some complete bottlenecks. Links 1, 2 and 3 are partial bottlenecks. Link 4 is a complete bottleneck. Link 1 is the bottleneck link for only group D sources. Link 2 for only group F, Link 3 for only group A and C. Link 4 is the bottleneck link for groups B and E.

The max-min fair shares for this configuration are shown in table 6. The simulation time used in all the cases below is 0.5 seconds.

Table 6: Max Min Fair Shares

Group	Fair Share
A	4.5 Mbps
B	11.1 Mbps
C	38.8 Mbps
D	4.5 Mbps
E	11.1 Mbps
F	51.5 Mbps

6.2.1 Case 1 : WAN Scenario. Inter-Switch Distance = 800 km . Source-Switch Distance = 3.2km i.e RTT = 8.064ms, All sources start at t=0

Figures 20 through 23 represent the plots of the various performance metrics for this case. The ACR and the MACR plots show that the values attained over the steady state agree with the values calculated using the max-min criteria. All the links are utilized to 100%. The queue sizes over the steady state converges to a value well within in the normal region.

6.2.2 Case 2 : WAN Scenario. Inter-Switch Distance = Group A to switch distance = 800 km . Other Distance = 3.2km , All sources start at t=0

Figures 24 through 27 represent the plots for this case. The plots demonstrate a similar behavior as in the previous case except that the queue sizes exhibit a more oscillatory behavior. This could be attributed to the fact that we have used longer feedback delay in this case. Note that the ACR plot of Group A sources is more oscillatory in nature compared to the previous case. This again is attributed to the longer feedback delay between group A sources and the switch.

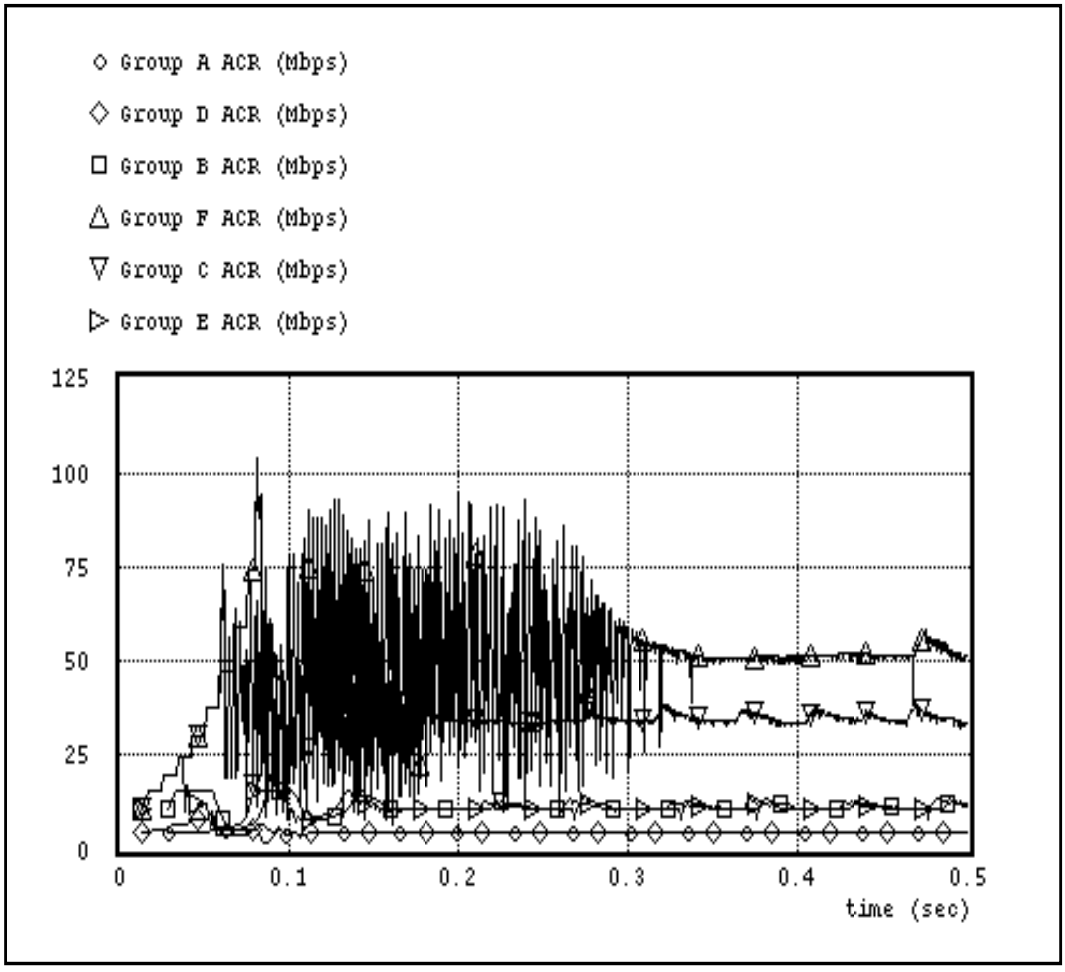


Figure 20: Generic Fairness Config. Case 1: Source ACR plots

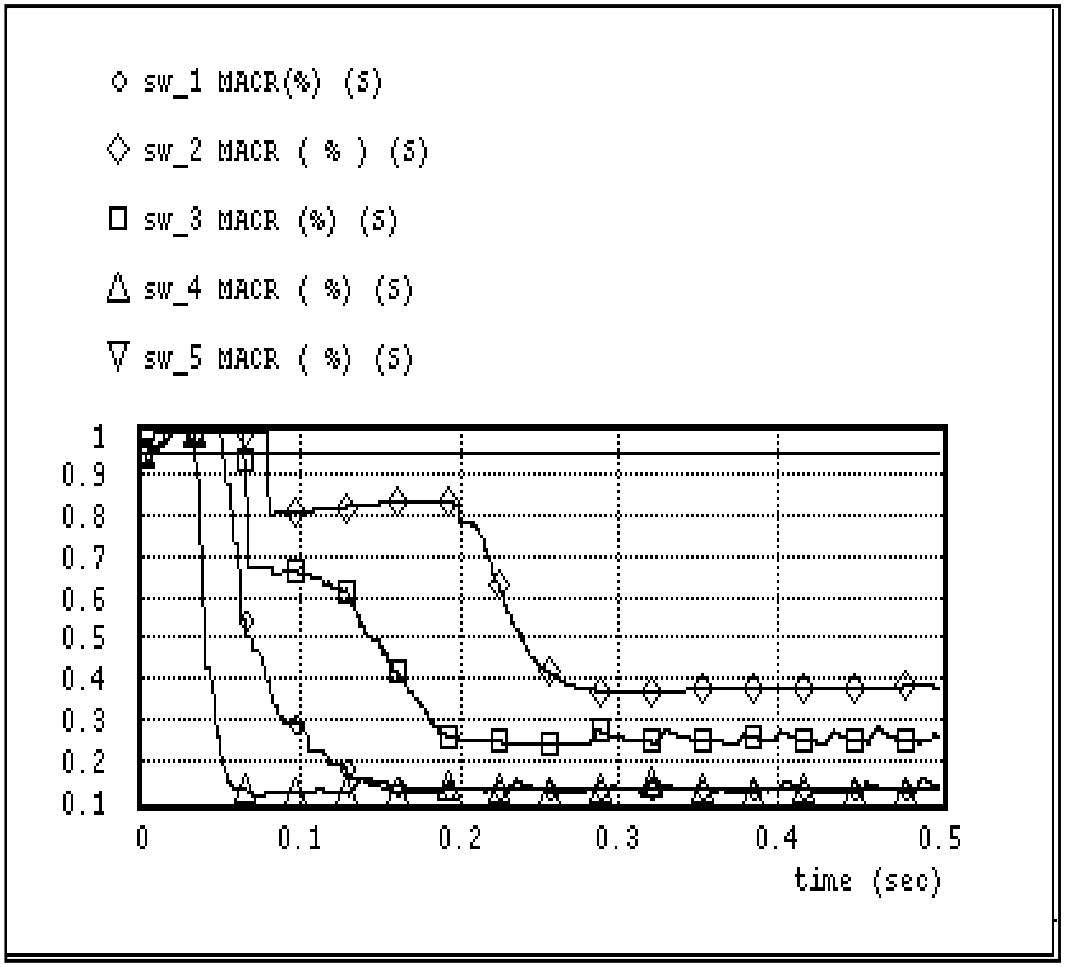


Figure 21: Generic Fairness Config. Case 1: Switch MACR plots

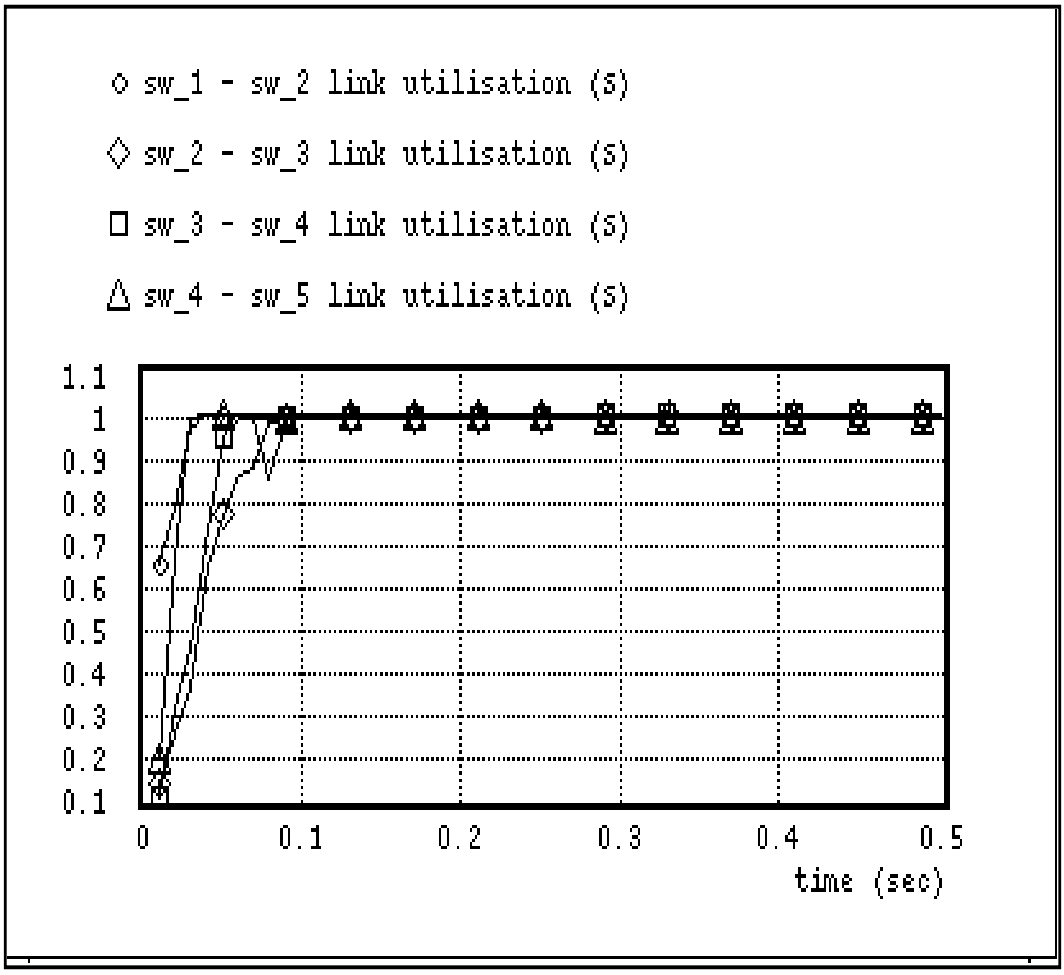


Figure 22: Generic Fairness Config. Case 1: Link Utilization plots.

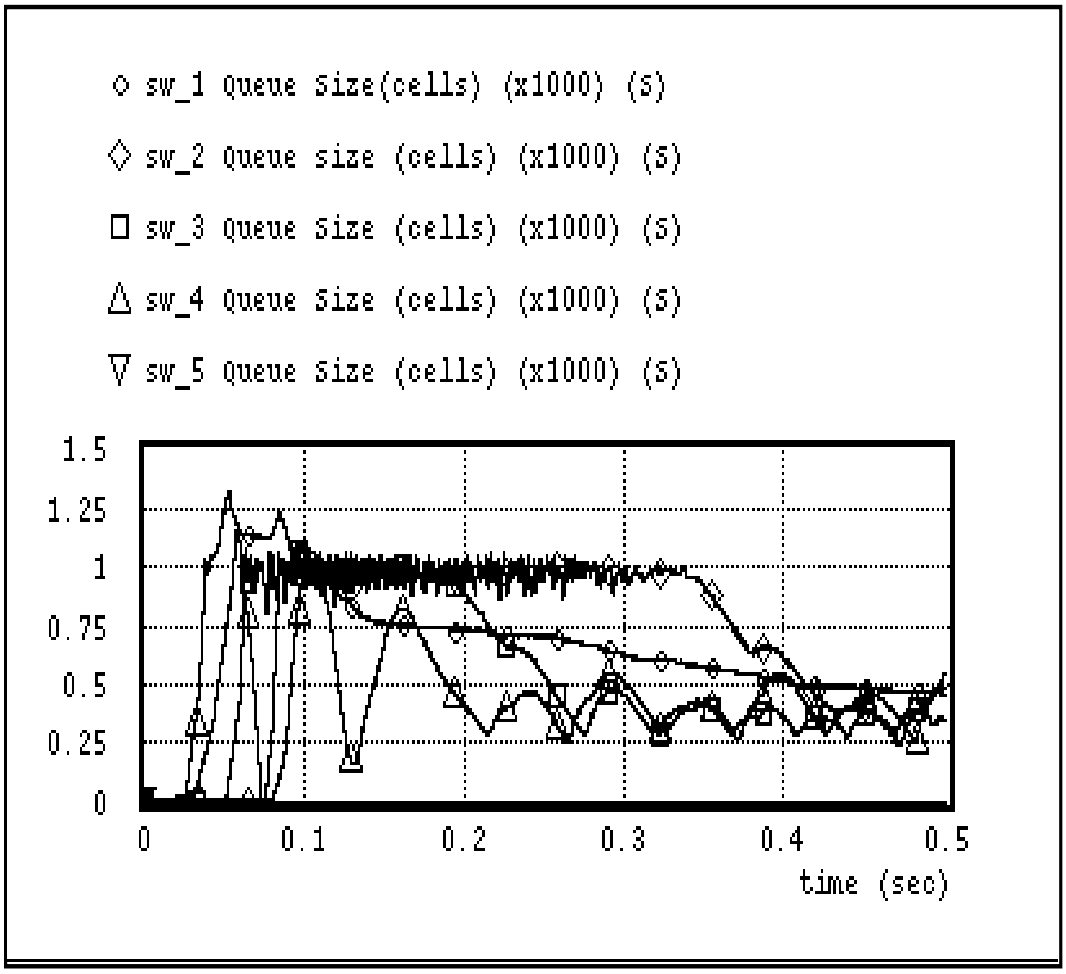


Figure 23: Generic Fairness Config. Case 1: Queue Size plots

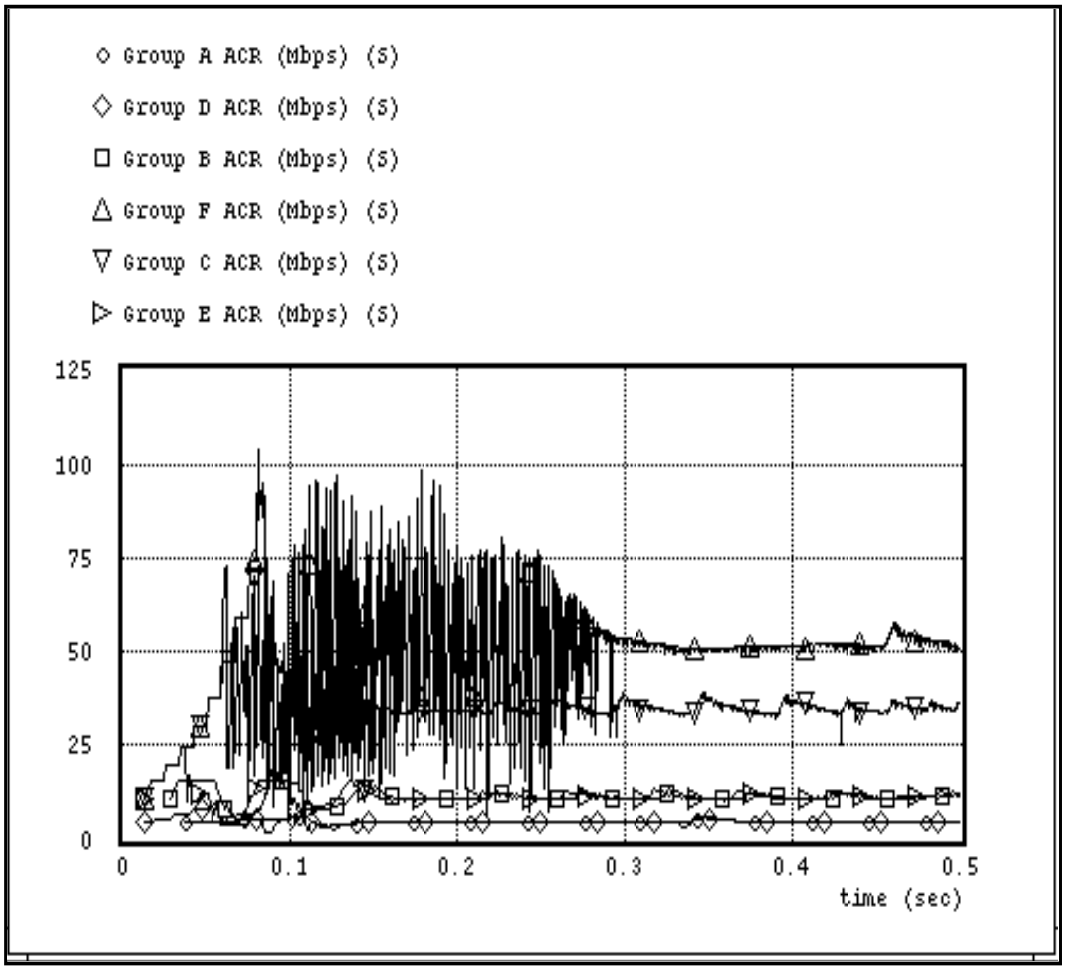


Figure 24: Generic Fairness Config. Case 2: Source ACR plot

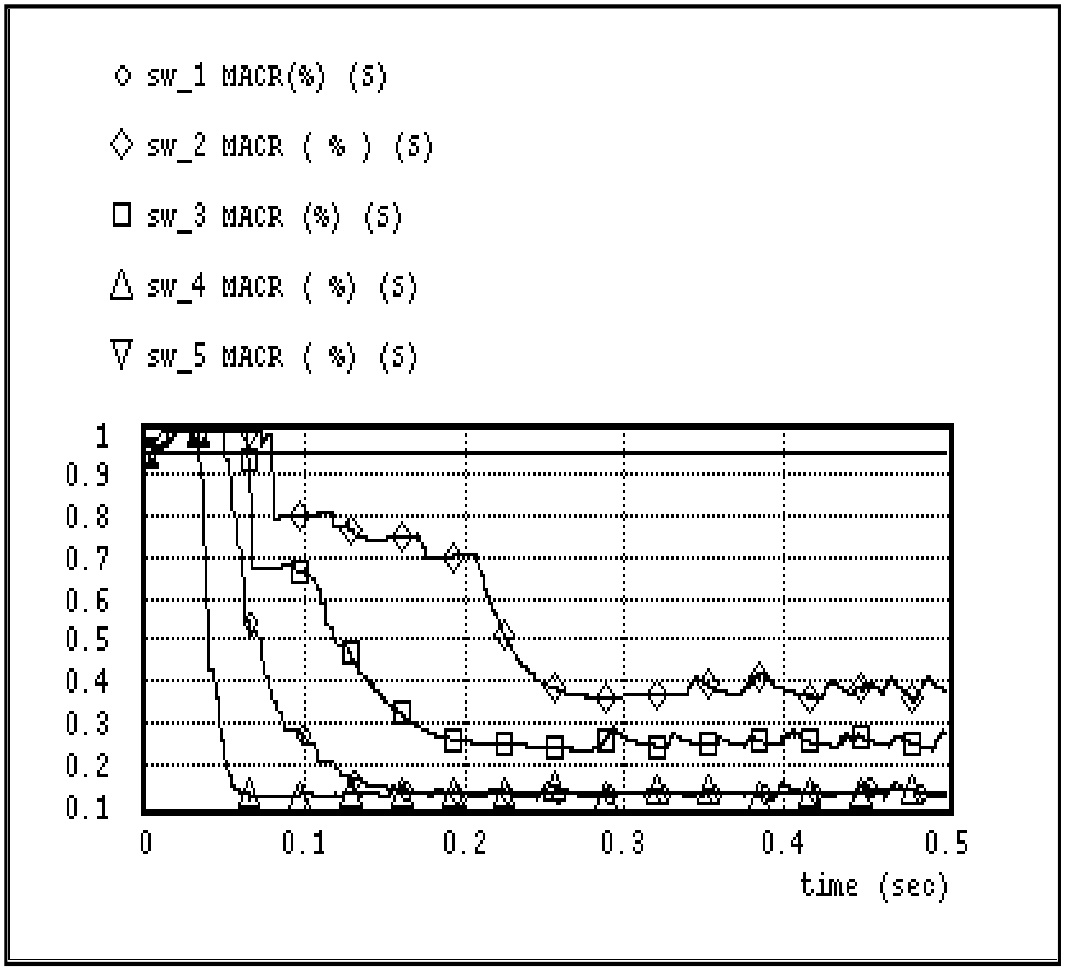


Figure 25: Generic Fairness Config. Case 2: Switch MACR plots

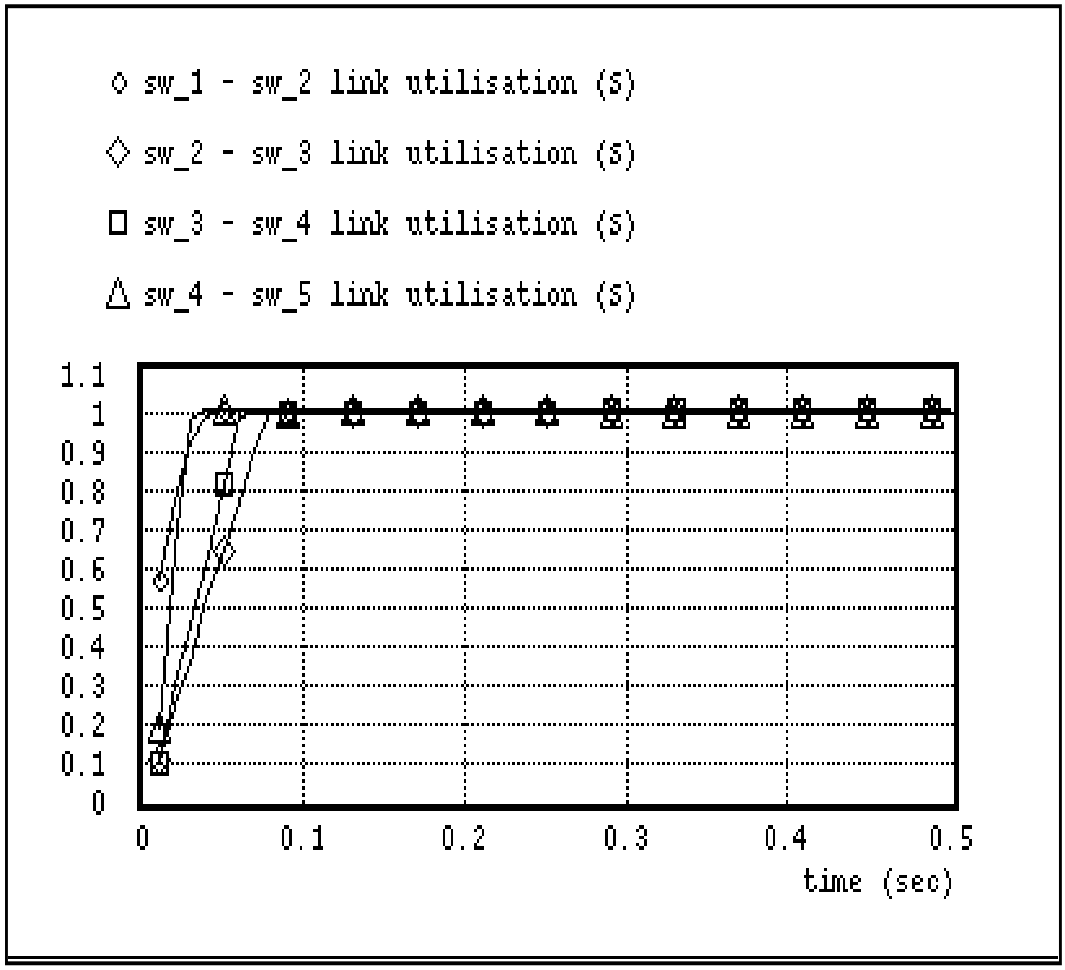


Figure 26: Generic Fairness Config. Case 2: Link Utilization plots

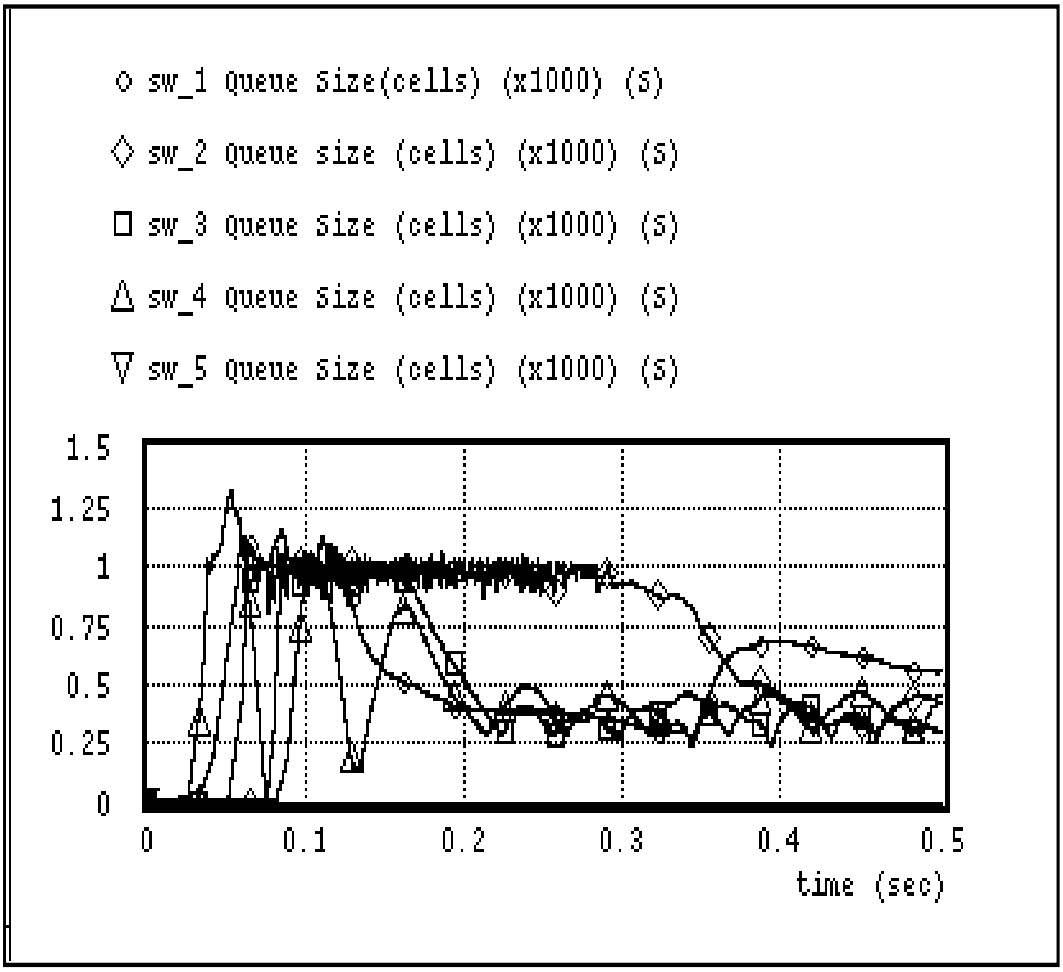


Figure 27: Generic Fairness Config. Case 2: Queue Size plots

6.3 Simple Core-Edge Topology

An Edge-Core Network topology contains two types of switches. Core switches that make up the interior of the network and edge switches that surround the core. VCs entering the network would first encounter an edge switch, then one or more core switches, then another edge switch. The core and edge switches can be produced by different vendors and thus run different ABR congestion control algorithms.

A network configured according to the Edge-Core architecture possesses a number of constructs in which ABR congestion control algorithms can produce suboptimal results. There will be multiple bottlenecks, some of which will be complete bottlenecks, while others will be partial. There will be VCs with different feedback delays and VCs that pass through switches running different ABR flow control algorithms. All of these attributes will be present in any realistic, heterogeneous network and are not just unique to the Edge-Core architecture.

In order to study the performance of the Series-D ER algorithm in the Core-Edge architecture, we used the topology shown in figure 28, that is representative of the larger architecture. Switches 2, 3 and 5 are core switches while switches 1, 4, 6 & 7 are edge switches.

In this topology with greedy sources and no background VBR traffic, the steady state throughput that should be received by each of the VCs is static. The values are given in table 7.

Several attributes worth mentioning of this topology are

1. The link between switch 1 and switch 2 (link 1) is a partial bottleneck. It is the bottleneck for VCs 1 and 2, but it is not a bottleneck for VC 3 and 4. The switch controlling it is an Edge switch and has a shorter feedback delay to the VCs that pass through it.

Table 7: Max-Min Fair Throughputs with no VBR traffic

VC	throughput(Mbps)
1 and 2	45
3 and 4	30
5, 6 and 7	30

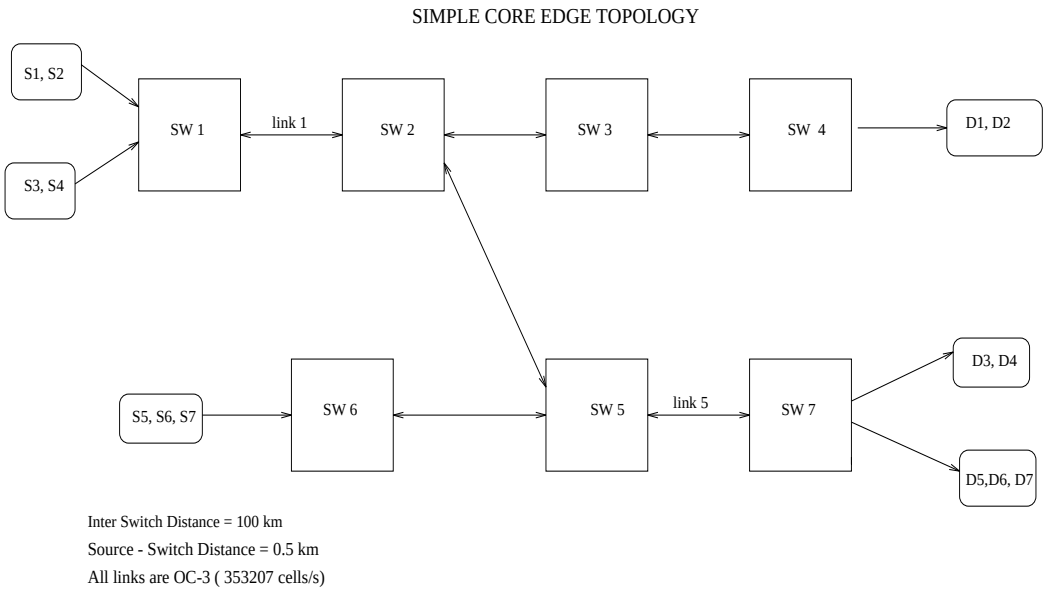


Figure 28: Simple Core/Edge Topology

2. The link between switches 5 and 7 (link 5) is a complete bottleneck, because it is a bottleneck for all the VCs that pass through it(VCs 3-7). The switch controlling it is a Core switch.

3. VCs 3 and 4 pass through two bottlenecks. Link 5 is their true bottleneck.

The simulation time used in this case is 0.5 seconds.

6.3.1 Case 1: Inter Switch distance = 100 km. Source-Switch distance = 0.5 km i.e, RTT = 1.1 ms. All links are OC-3.

Figures 29 through 32 show the ACR, MACR, Queue Size and the link utilization plots for this case. From the ACR plots (Figure 29) it can be seen that VCs 1 and 2 attain a steady state ACR value of 45Mbps and VCs 3-7 attain a value of 30Mbps. This is in agreement with the values calculated using the Max-Min fairness criteria (Table 7).

The MACR plots of all the switches other than Switch 1 and Switch 5 exhibit an almost constant value of 1. This is in consensus with the expected behavior as only switches 1 and 5 a re bottleneck switches and it is only the MACR values at these switches that affect the ACRs of the bottle-necked VCs corresponding to them. Since we have switch 1 as the bottleneck switch for VCs 1 and 2, we should be expecting the switch 1 MACR to reach a value of $45/150 = 0.3$ i.e, 30% of the link speed and switch 2 MACR value to be $30/150 = 0.2$ i.e, 20% of the link speed. The simulation results show exactly the same values.

Link Utilization plots of the various links are shown in figure 31. The links between switches 1 and 2 and switches 5 and 7 is expected to be 100%. Links between switches 2 & 3, 3 & 4, 6 & 5 should attain a value of 60% and the link between switch 2 and 5 is expected to be utilized to 40%. The simulation results indicate a near ideal behavior.

The Queue size plots are shown in Figure 32. Only the queue sizes of switch 1 and switch 2 are of interest. Over the steady state the queue sizes of both of these switches converge to a value of approximately 300 cells which is our normal region. So our intended goal for this configuration is also met.

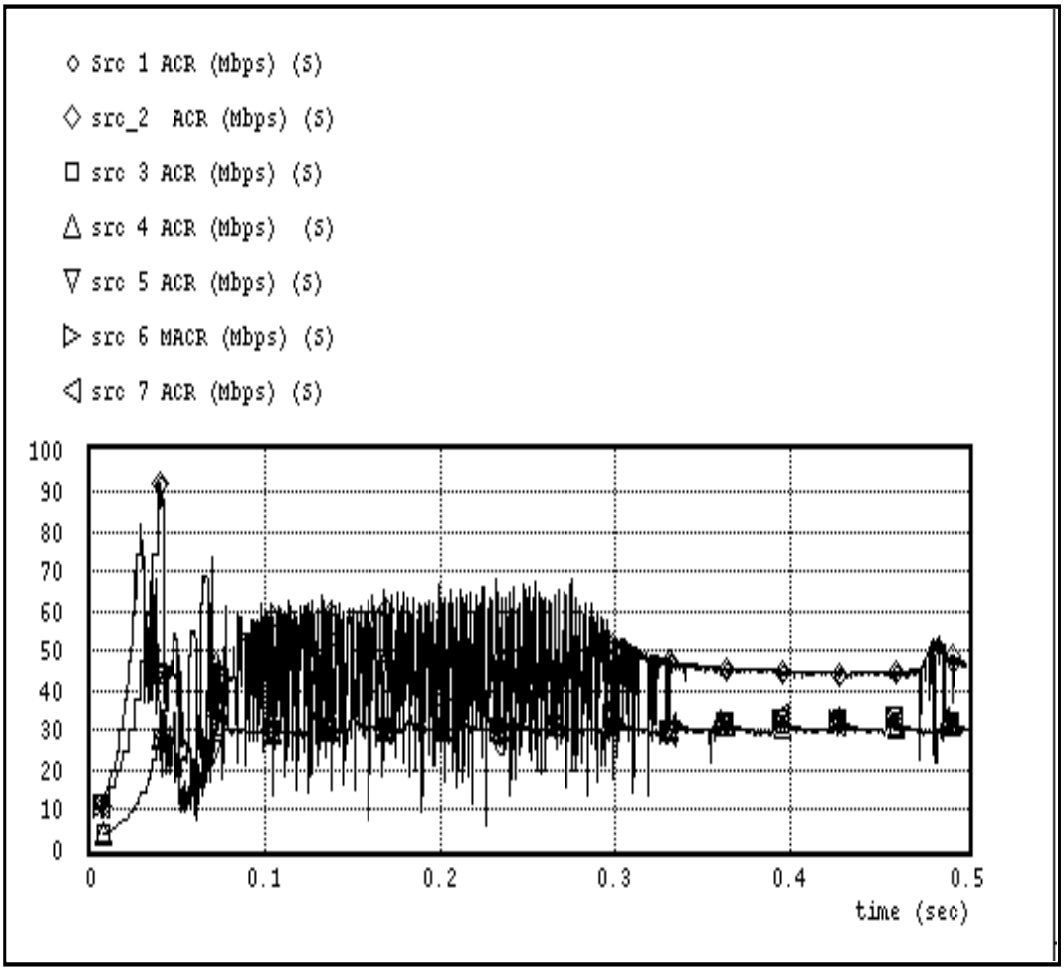


Figure 29: Simple Core/Edge Topology: Source ACR plots

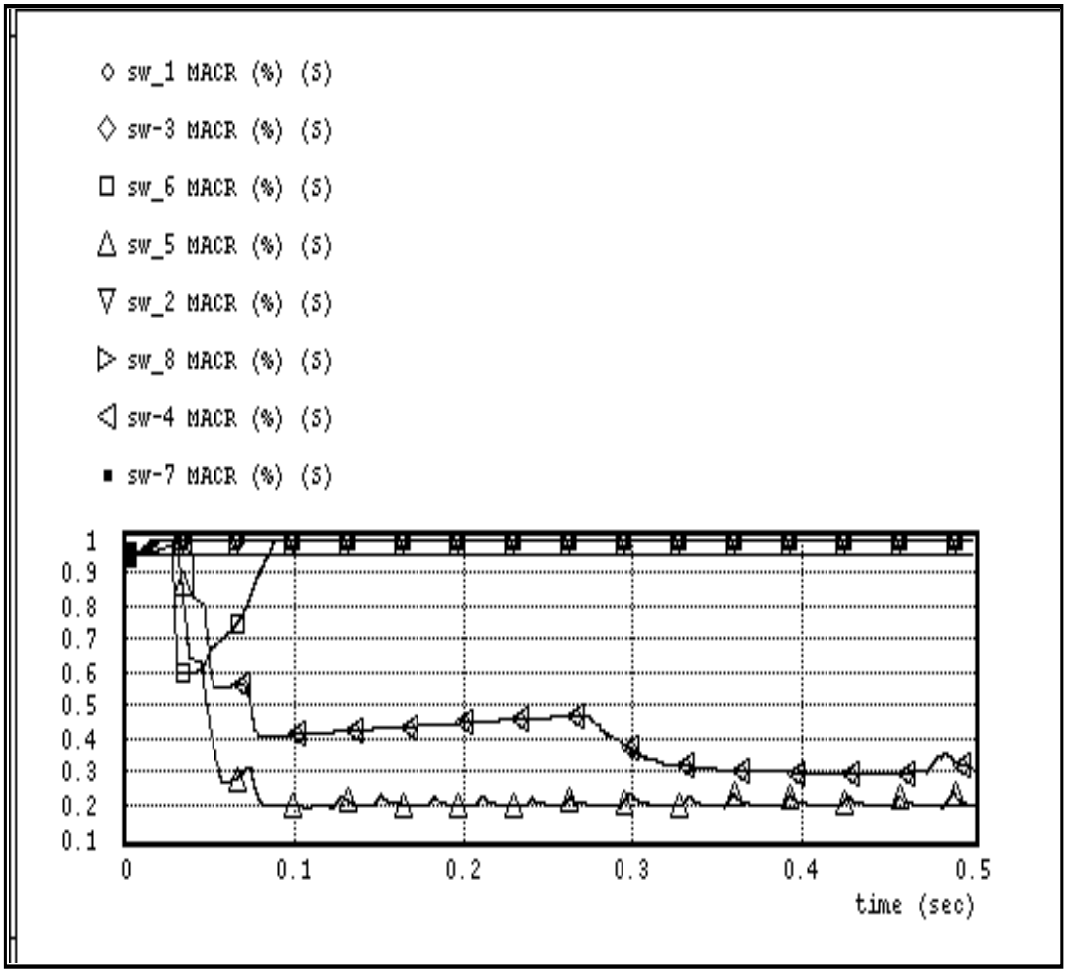


Figure 30: Simple Core/Edge Topology: Switch MACR plots

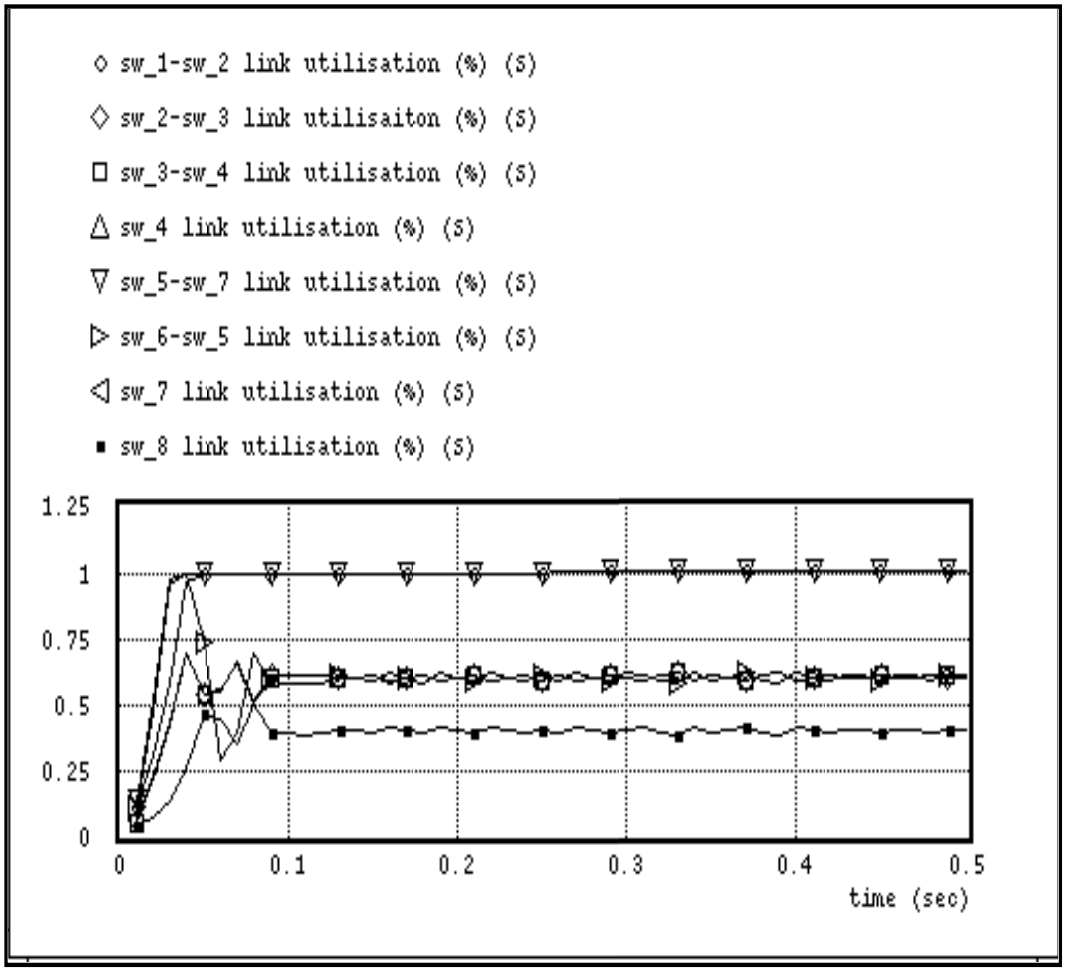


Figure 31: Simple Core/Edge Topology: Link Utilization plots

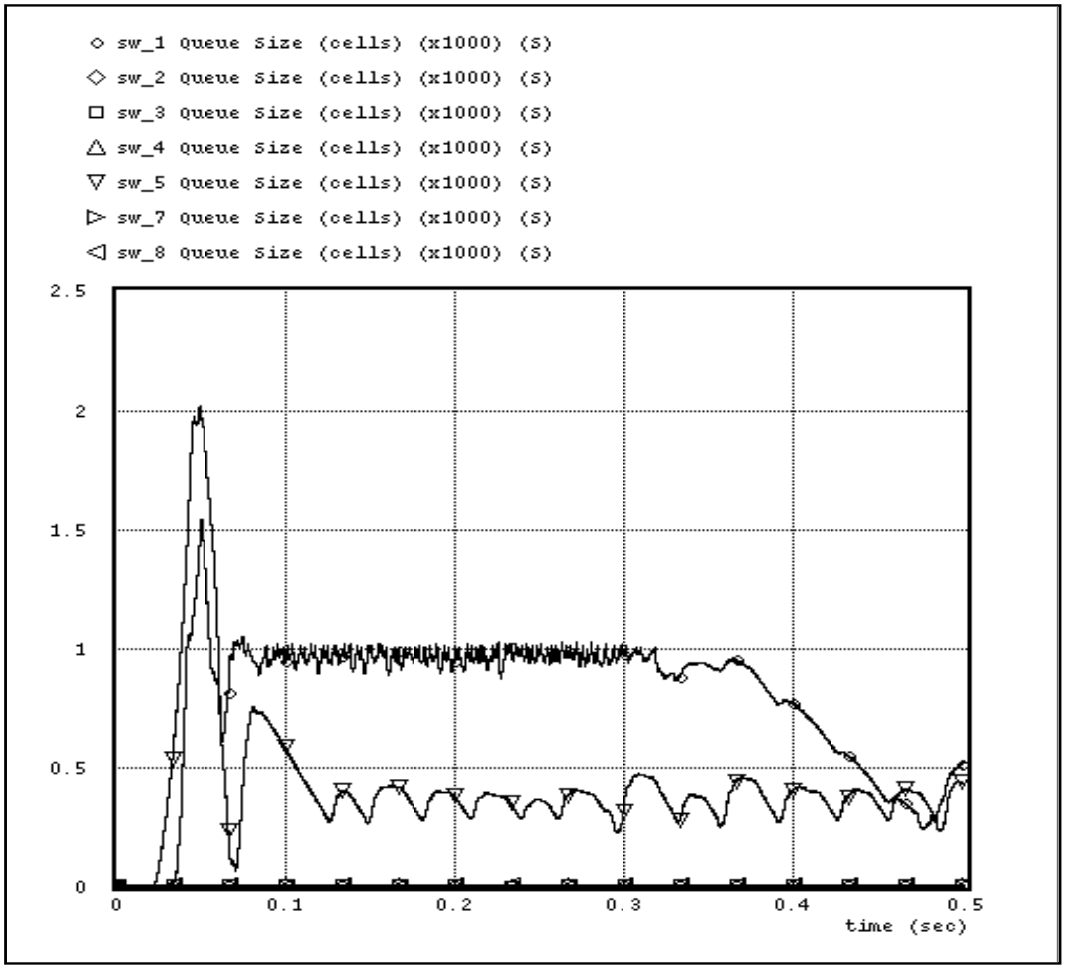


Figure 32: Simple Core/Edge Topology: Queue Size plots

7 Conclusions and Future work

The performance of the series-D ER ABR algorithm has been evaluated in the presence of pure ABR traffic. The simulation results show that the algorithms performs well in terms of fairness, throughput and buffer sizes for all the cases. The next phase of the study will consider the following issues.

1. Performance of the algorithm in the presence of high priority traffic i.e, VBR traffic.

2. Investigation of the extent to which the switch architecture will affect the algorithm's performance.

3. Determining the interactions of TCP over ABR i.e testing the algorithms performance with TCP sources.

4. Testing the algorithm's performance in heterogeneous network topologies i.e, in conjunction with other ABR algorithms.

8 References

[1] Pseudo code from FORE.

[2] OPNET Modeler Modeling Volume 1 & 2.

[3] ATM Forum Traffic Management version 4.0

[4] Bertsekas, D., and R. Gallager. 1993. Data Networks, Prentice Hall.