Performance and Robustness Testing of Explicit-Rate ABR Flow Control Schemes

M. Zoranovic, C. Williamson
Department of Computer Science
University of Saskatchewan

Abstract

The objective of this paper is to compare and evaluate three candidate ABR flow control strategies (ERICA, ERICA+, and DEBRA) for ATM networks, using simulation. A set of benchmark network scenarios is proposed and used to illustrate various properties of each algorithm, such as efficiency, fairness, responsiveness, and scalability, as well as robustness in the presence of uncooperative sources. The simulation results show that ERICA+ and DEBRA perform similarly on most scenarios, and slightly better than ERICA, while DEBRA is more robust than ERICA and ERICA+. Overall, the study shows that DEBRA, a new explicit-rate flow control strategy, shows good potential for ABR traffic control in ATM networks.

Keywords*: Simulation, ABR flow control, ATM networks, performance evaluation*

1 Introduction

The Available Bit Rate (ABR) traffic class in Asynchronous Transfer Mode (ATM) networks has generated much discussion and debate within the ATM Forum and beyond [3, 7, 11, 13]. This traffic class is intended for data transmissions that are loss-sensitive, not as delay sensitive as voice and video, and can be transmitted at whatever rate is currently convenient for the network.

The goal in ABR service is to adjust source rates dynamically so as to maximise the utilisation of the available bandwidth for the ABR service class, without overloading the network. Furthermore, additional goals such as fairness of bandwidth allocation amongst competing ABR traffic flows must also be achieved.

ABR control schemes have evolved significantly over the past few years, from binary feed-back schemes, such as Explicit Forward Congestion Indication (EFCI) [11] to Relative-Rate (RR) schemes, such as EPRCA [14] and DMRCA[5], to Explicit-Rate (ER) schemes, such as ERICA and ERICA+ [8]. Explicit-Rate schemes use the ER field in Resource Management (RM) cells to specify source transmission rates. Sources adjust their transmission rate to the value indicated by the switches along the congested path.

Although many Explicit-Rate flow control schemes have been proposed [2, 4, 5, 6, 8, 9, 12, 14] there is not yet a commonly accepted set of network configurations to evaluate and compare these schemes. Many authors use one or two network scenarios to illustrate the behaviour of their schemes, with limited comparison to other ABR schemes. This makes direct comparison of competing ABR congestion control strategies difficult.

This paper makes three main contributions in this regard. First, it describes a collected set of network scenarios, drawn primarily from the research literature [4, 5, 9, 10, 18, 19] that can be used for benchmarking ABR algorithms. Second, it proposes additional scenarios for robustness testing of ABR algorithms in the presence of uncooperative ABR sources. Third, it proposes and evaluates a new ABR algorithm called DEBRA, and compares it to ERICA and ERICA+. The experiments are conducted using the ATM-TN (Asynchronous Transfer Mode Traffic and Network) simulator, developed as part of the TeleSim project [15, 16, 19].

The rest of the paper is organised as follows. Section 2 provides background information on ABR traffic control and the flow control schemes to be compared, namely ERICA, ERICA+, and DEBRA. Experimental design of the research is described in Section 3. Section 4 presents the simulation results for ERICA, ERICA+, and DEBRA on three network scenarios. The simulation results for robustness of the schemes are presented in Section 5. Finally, Section 6 presents conclusions.

2 ABR Traffic Control

2.1 ABR Flow Control Mechanism

In ATM networks, ABR traffic sources adjust their transmission rates dynamically between a pre-specified Minimum Cell Rate (MCR) and Peak Cell Rate (PCR), based on the amount of network bandwidth left unused by higher priority traffic classes. The rate adjustment is done using a closed-loop feedback mechanism, using RM cells. RM cells convey control information to ABR traffic flows about the state of the network, such as congestion state and bandwidth availability.

The ABR flow control mechanism is called closed-loop since it uses feedback information from the network to control the rate of each source. Forward Resource Management (FRM) cells are generated by sources and inserted into the outgoing data cell stream. On their way to the destination and back from the destination to the source, RM cells are processed by switches (Figure 1). When an RM cell arrives at the destination, the destination changes the direction bit (DIR) in the cell and returns it to the source. RM cells travelling from the destination to the source are called Backward Resource Management (BRM) cells. BRM cells bring updated network state information to the sources.

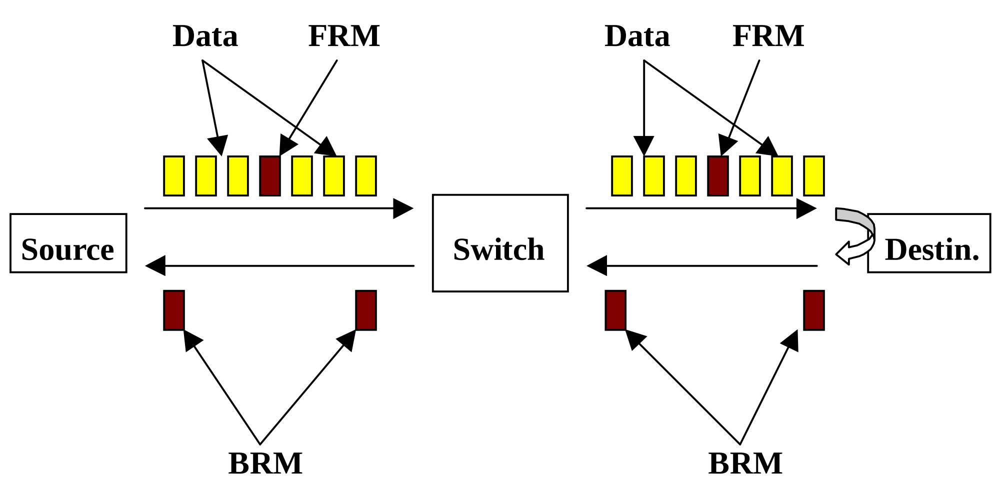


Figure 1: *ABR Traffic Management Model*

In Explicit-Rate based congestion control schemes, BRM cells tell the source exactly what transmission rate to use for outgoing traffic. This information is contained in the ER field of the RM cell. The rate at which a source is allowed to transmit cells is called the Allowed Cell Rate (ACR). The ACR is initially set to a default value called the Initial Cell Rate (ICR). It is always between the MCR and the PCR. The source puts its ACR value into the Current Cell Rate (CCR) field of outgoing RM cells, while the rate at which it wishes to transmit cells is put into the ER field. RM cells are generated by the source after every (Nrm –1) data cells are transmitted, where Nrm is a parameter to the ABR traffic control algorithm (the default value for Nrm is 32). When an RM cell arrives at the destination, if the destination is congested and cannot support the rate in the ER field, the destination reduces the ER to whatever rate it can support. The returning BRM cell will convey this information to the source [3, 7].

As an RM cell travels back through the network, each switch examines the cell to determine if it can support the ER rate for the requested connection. If the ER is too high for a switch, the switch reduces this value to a rate that it can support. Note that no switch is allowed to increase the ER, because doing so would violate the rate set by another switch (or destination), and likely cause transmitted cells to be lost at the bottleneck points.

When a source receives a BRM cell, it computes its allowed cell rate (ACR) using information from CCR and the ER field, and other information from the BRM cell [3]. The behavioural requirements of ABR traffic end-points (sources and destinations) are defined and explained in [3, 7]. Switch behaviours, on the other hand, are only outlined in [3]. Currently, switch behaviour is vendor specific and depends on the flow control algorithm implemented within the switch.

All ABR flow control algorithms use the same basic RM cell framework. The main characteristics of the three ABR algorithms evaluated in this paper, ERICA, ERICA+, and DEBRA, are described in the following sections.

2.2 The ERICA Algorithm

The ERICA (Explicit Rate Indication for Congestion Avoidance) algorithm [8, 9], proposed by Raj Jain *et al.*, tries to achieve a fair and efficient allocation of the available bandwidth to contending sources.

The basic idea in ERICA is to monitor, at each switch, the incoming cell rates of each ABR traffic source, and compare the aggregate ABR traffic demand to the desired target utilisation *U* for ABR traffic sources (typically *U*=0.95% of available ABR capacity in LAN environment, and *U*=0.90% of available ABR capacity in WAN environment). If the aggregate demand is less than the target load, then traffic source rates can be increased. If the aggregate demand exceeds the target load, then traffic source rates must be decreased. A parameter δ (e.g., $\delta = 0.2$) is used to determine what constitutes an aggregate demand that is "close" to the desired load, and a scaling factor of 1/*z*, where *z* is the ratio of actual load to desired load, is used to control the gradient of rate adjustments for the sources.

ERICA estimates source rates by counting incoming cells over an averaging interval. The duration of an averaging interval in ERICA is defined as every *t* seconds (e.g., *t* = 0.001 seconds), or every *count* cells (e.g., *count* = 50), whichever comes first.

Hence, the ERICA algorithm has four parameters in total: *Target Utilisation (U)*, δ, *t*, and *count*. The default values for these parameters are as indicated above.

2.3 The ERICA+ Algorithm

The ERICA+ algorithm, developed by Raj Jain *et al.* [8, 9], is a modified and improved ERICA algorithm with a few enhancements, such as target queueing delay rather than a target utilisation, and refined parameters for source rate adjustment for faster steady-state convergence.

The target queueing delay (*D*), determines the steady state buffer occupancy at the bottleneck link. With this approach, ERICA+ can achieve higher network utilisation than ERICA (i.e., 100% instead of 90% or 95%), while increasing the end-to-end delay only slightly (e.g., *D*=100 microseconds). In addition to this delay parameter, ERICA+ uses two hyperbolic functions (specified as parameters *a* and *b*, with typical values *a* = 1.15 and *b* = 1.05) to provide smoother rate adjustments (compared to ERICA's crude *1/z* scaling factor) around the desired equilibrium point, and a queue drain limit factor (*QDLF*) to bound the rate adjustment function (e.g., *QDLF*= 0.5) [8, 9].

Thus, ERICA+ has seven parameters in total. Three of these are inherited from ERICA (*δ, count* and *t*), and four are new ones (*D, a, b* and *QDLF).*

2.3 The DEBRA Algorithm

DEBRA (Dynamic Explicit Bid Rate Algorithm) is proposed by R. Gurski and C. Williamson [6]. This algorithm is based on a rate-based congestion control strategy called loss-load curves [17]. In the loss-load approach, switches compute and provide to traffic sources concise aggregate load information, allowing sources to compute precise transmission rates that provide the best trade-off between offered load and the level of packet loss experienced in the network.

The DEBRA algorithm works as follows. Switches compute the aggregate demand from the incoming bid rates. If this demand is less than or equal to the available ABR capacity, then each source simply receives its desired allocation. If the aggregate demand exceeds the available capacity, then partial allocations are made to each source using a rate allocation function (obtained from the loss-load curve using: $\tau = r * (1 - p)$, where τ is allocated bandwidth, r is the requested bandwidth, and p is the loss probability). The algorithm favours sources with lower bid rates as opposed to higher bid rates (greedy sources may receive little or none of their requested allocation), while still guaranteeing 100% utilisation of target ABR capacity. Another interesting feature of the algorithm is that switches can advertise the rate allocation function to the traffic sources. Sources use the advertised function to determine optimal bids (i.e., how to maximise their own individual bandwidth allocation in the presence of traffic bids from other sources). DEBRA has proven mathematical properties of bounded load, convergence, fairness and stability [17].

There are three parameters for DEBRA: *K*, *C* and *V*. *C* and *V* control the target utilisation (e.g., *C*=0.95) of the ABR capacity, and the fraction of ABR bandwidth that is actually advertised to traffic sources (e.g., V=1.0), respectively. The parameter K controls responsiveness, aggressiveness, and convergence time of the algorithm. Increasing K makes the algorithm less aggressive, but also reduces the responsiveness of the algorithm and slows the convergence [17].

3 Experimental Methodology

3.1 ATM-TN Simulator

The simulation experiments reported in this paper were conducted using the Asynchronous Transfer Mode Traffic and Network (ATM-TN) simulator [15, 16].

The ATM-TN simulator provides cell-level simulation of the ATM-TN traffic flows from traffic sources to traffic sinks, traversing one or more simulated ATM switches and links. Several different traffic source models and ATM switch models are supported in the simulator [1, 15, 16]. For this research, only the ABR persistent source model and the (per port) output buffered switch model were used. Also, the algorithms described in Section 2 are incorporated in the simulator, together with a set of benchmark scenarios described in next section.

3.2 Benchmark Scenarios

The network configurations used for evaluating ABR flow control schemes fall into two categories: performance tests and robustness tests. These configurations are summarised in Table 1 and Table 2.

The purpose of the first set of scenarios is to evaluate ABR algorithms under the assumption of cooperating source (i.e., sources respond correctly to the Explicit-Rate feedback in RM cells). Nine scenarios are used for these tests [19], though only three are presented in this paper.

The purpose of the second set of scenarios is to assess the performance of ABR algorithms when the cooperation assumption is relaxed. That is, the scenarios consider

Table 1: Benchmark Scenarios for Performance Testing

No.	Network Scenario	Type	Purpose
1.	One Source	LAN	Single traffic source, which should use all available ABR bandwidth.
2.	Two Sources	LAN	Test an algorithm for fairness and effective use of ABR bandwidth.
3.	Two Sources Staggered	LAN	Illustrates the responsiveness, fairness and efficiency of an algorithm when sources start and finish at different times.
4.	One-at-a-Time Arrivals	LAN	Test responsiveness, fairness and efficiency of an algorithm (see Section 4.1).
5.	Three-Switch Parking Lot	LAN	Test ABR algorithms for max-min fairness of bandwidth allocations amongst traffic sources with downstream bottlenecks.
6.	Five-Switch Parking Lot	MAN	The same as for scenario 5.
7.	Upstream Traffic	WAN	Assess the max-min fairness amongst traffic sources in the presence of upstream bottleneck on the network (see Section 4.2).
8.	Generic Fairness Configuration 1 (GFC1)	WAN	Test ABR algorithms for max-min fairness among traffic sources (see Section 4.3).
9.	Generic Fairness Configuration 3 (GFC3)	LAN MAN WAN	Assess the max-min fairness amongst traffic sources with different transmission capacities and propagation delays.

sources that intentionally overuse or underuse their share of the network (i.e., behave independently from the Explicit-Rate feedback provided to them), either with or without telling the switches of their actual rates (i.e., honest and dishonest traffic sources). Four configurations are considered in this category (see Table 2 and [19]), though only two are presented in this paper. All are based on Two Source LAN scenario [19].

Table 2: Benchmark Scenarios for Robustness Testing

No.	Network Scenario	Type	Purpose
1.	Dishonest sources	LAN	Robustness testing (see Section 5.1).
2.	Honest Sources-One High	LAN	Tests an algorithm for robustness against greedy source (see Section 5.2).
3.	Honest Sources-One Low	LAN	Assess the robustness of an algorithms when one of the sources decides to transmit at the rate lower than it fair share.
4.	Network Scenario with Extremely Long RTT	WAN	Tests the robustness of an algorithm in a presence of sources with extremely long RTT.

Following the methodology presented in [9], all network scenarios have the following settings, unless specified otherwise:

- network links are OC-3 (155.52 Mbps);
- each switch output port has a 1000-cell FIFO buffer for ABR traffic;
- traffic sources are greedy persistent sources, with infinite data to transmit;
- traffic sources have a PCR of 155 Mbps, an ICR of 25 Mbps, and an MCR of 0 Mbps;
- the Rate Increase Factor (RIF) parameter for sources is set to 1.0, allowing sources immediate use of the ER value indicated in returning RM cells;
- LAN configurations use 1 km links, with a propagation delay of 5 μsec;
- WAN configurations use 1000 km links, with a propagation delay of 5 msec.

3.3 Performance Metrics

Qualitatively, an ABR algorithm should provide full use of available ABR bandwidth, fairness among competing ABR sources, good steady-state and transient behaviour, and a low cell loss ratio. Scalability (e.g., with number of sources or with feedback delay) and robustness are also important.

Quantitatively, the performance of ABR algorithms is assessed using the following performance metrics:

- *Allowed Cell Rate (ACR*). The Allowed Cell Rate is used to show a source transmission rate as a function of time. It is expressed in Mbps, rather than cells per second, to facilitate direct comparison with link capacity used in each scenario.
- *Link Utilisation.* The link utilisation shows the percent of utilisation of a network link as a function of time;
- *Queue Length*. The queue length shows queue occupancy of the switch output buffer as a function of time. It is expressed in number of cells.
- *Throughput*. The throughput shows the cumulative count of the number of cells successfully delivered to an ABR destination as a function of time. It is expressed in cells. The slope of this function represents throughput in cells per second.

- *Cell Loss Ratio (CLR).* CLR is expressed as a ratio of the number of cells lost versus the number of cells transmitted.

In this paper, we focus primarily on two metrics: ACR and queue length. All five metrics are considered in [19].

4 Simulation Results: Performance Testing

This section presents simulation results for all three ABR algorithms on three network scenarios, namely One-at-a-Time Arrivals, Upstream Traffic and GFC1.

4.1 One-at-a-Time Arrivals Network Scenario

The first benchmark scenario is a LAN network configuration with 30 traffic sources (Figure 2) [18]. In this scenario, the individual traffic sources start up one at a time,

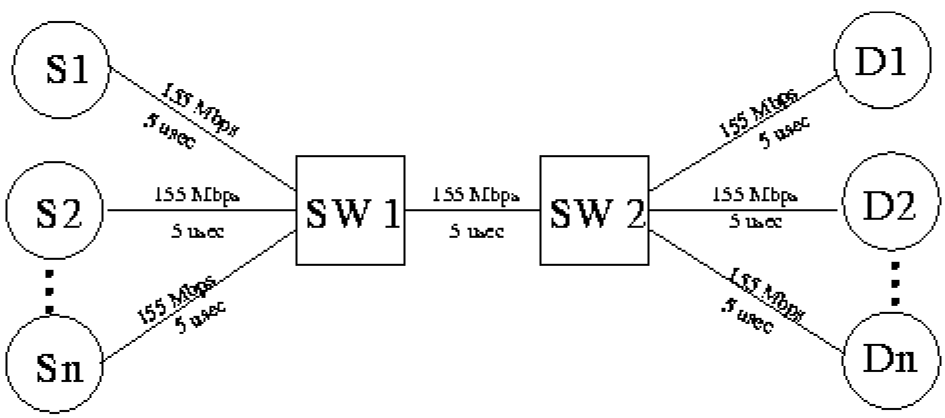


Figure 2: "One-at-a-Time Arrivals" Network Scenario

every 10 milliseconds, and each source has the same round trip time (approximately 30 microseconds). The Initial Cell Rate (ICR) for each source is 4.24 Mbps.

This scenario tests the responsiveness, fairness, and efficiency of ABR flow control schemes: the first source should reduce its transmission rate every time a new traffic source arrives, so that the ABR capacity is shared fairly among all traffic sources, with minimal queuing at the bottleneck switch.

The simulation results for ERICA on the one-at-a-time arrivals scenario are shown in Figure 3a and 3b. In Figure 3a, only the ACR's of S1 and S2 are shown, since the rest of the sources have similar behaviour. As one can see, source S1 initially uses the full bandwidth available, then reduces its rate to share the bandwidth fairly with the second source, and reduces its rate again to share the bandwidth equally amongst three sources, and so on. The resulting curve is a step function with progressive rate reductions until all 30 sources are transmitting in the network.

Figure 3b shows that each new arrival generates an impulse queue buildup, but the queue size quickly returns to its steady-state behaviour (0 to 2 cells) following this. After arrivals of all 30 ABR sources, ERICA goes into the steady state as expected.

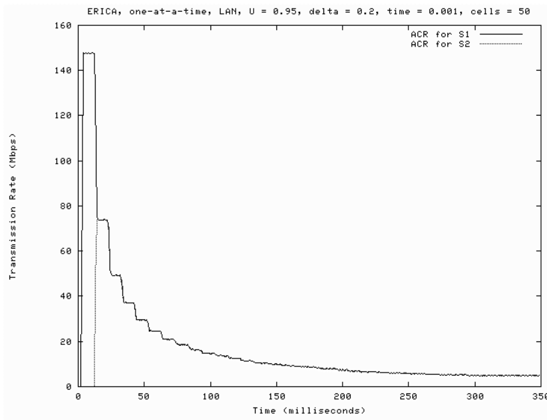
The simulation results for ERICA+ on this scenario are shown in Figure 3c and 3d. As with the ERICA results, the ERICA+ algorithm shows a clear progression of rate reductions for the first two sources (Figure 3c) as they react to each new ABR source arrival. Figure 3d shows that there is a short queue buildup related to the arrival of each new source, but the queue size converges to 70 cells (corresponding to the target queueing delay D=0.0002 sec) once all the sources are active.

The DEBRA results for this scenario (Figure 3e and 3f) are quite similar to those for ERICA and ERICA+. Figure 3e shows that source S1 initially uses the full bandwidth available, and then reduces its rate repeatedly until all 30 sources are active in the network. Figure 3f shows that each new arrival generates an impulse queue buildup, but the queue size quickly returns to its steady-state behaviour (0 to 2 cells) following this. It is noticeable that as the number of active sources increases, the magnitude of impulse decreases. This behaviour occurs because newly arriving sources always choose bid rates lower than those of the active sources [17].

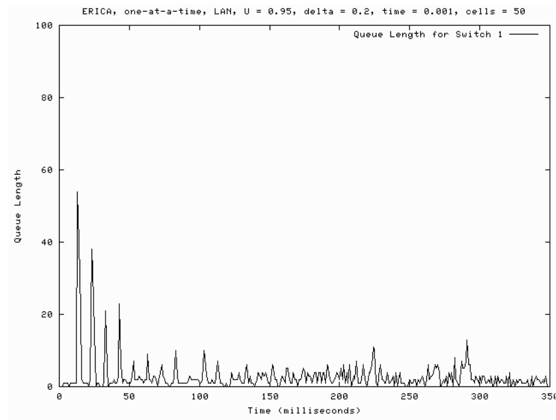
All three ABR algorithms perform well on this scenario.

4.2 Upstream Traffic Network Scenario

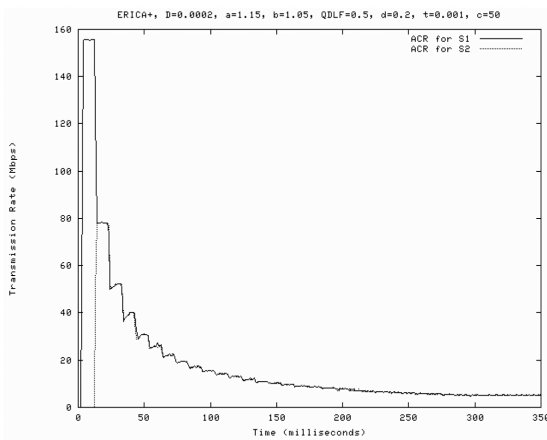
The next scenario (Figure 4) is a WAN configuration with an "upstream" bottleneck, as proposed in [9]. The purpose of this scenario is to assess the max-min fairness of



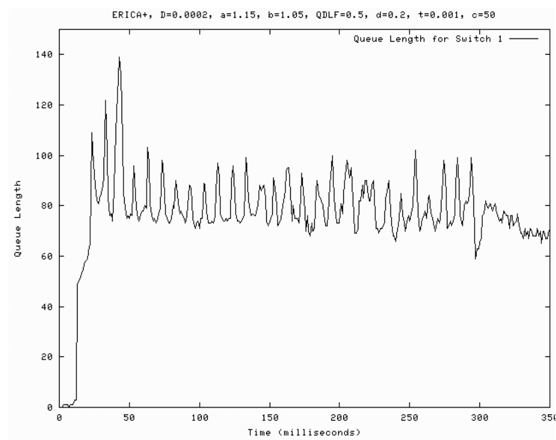
a) ERICA: ACR



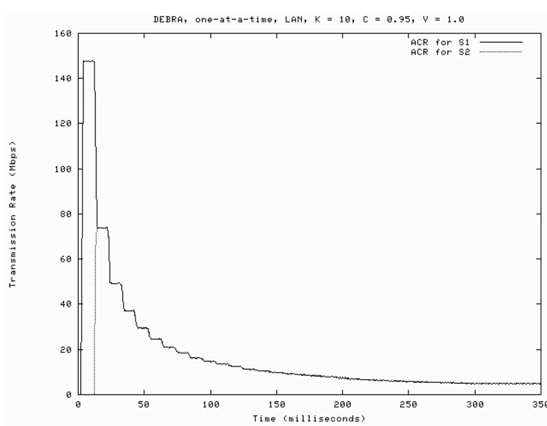
b) ERICA: Queue Length



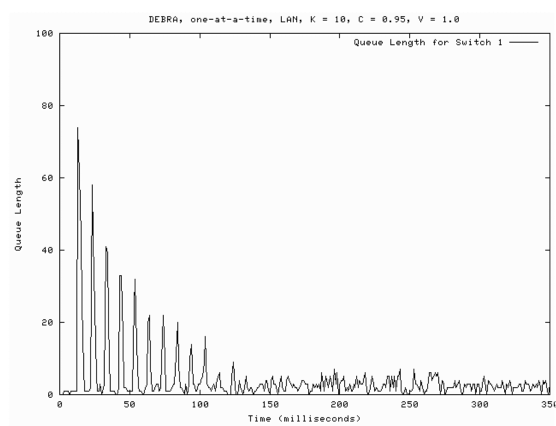
c) ERICA+: ACR



d) ERICA+: Queue Length



e) DEBRA: ACR



f) DEBRA: Queue Length

Figure 3: Simulation Results on One-at-a-Time Network Scenario

bandwidth allocations amongst the sources. The ERICA target utilisation parameter is set to U = 90%, the ERICA+ target delay parameter is set to D = 0.0005 seconds, and the count-based rate estimation interval is set to 100 cells [9]. In addition, the ICR for source 16 is set to 50 Mbps, the ICR for source 17 is set to 70 Mbps, while the ICR is set to 25 Mbps for all other sources. Since 15 sources (from 1 to 15) compete for the first inter-switch link, each of them should obtain about 10.4 Mbps. Source 16 and source 17 share the second inter-switch link with source 15, and each of them should obtain about 72.5 Mbps (since source 15 uses only 10.4 Mbps).

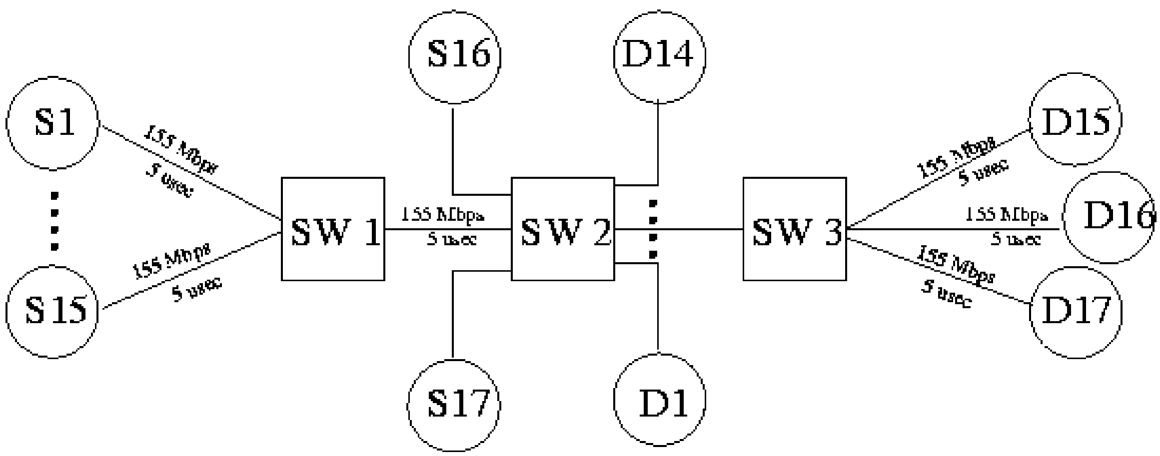
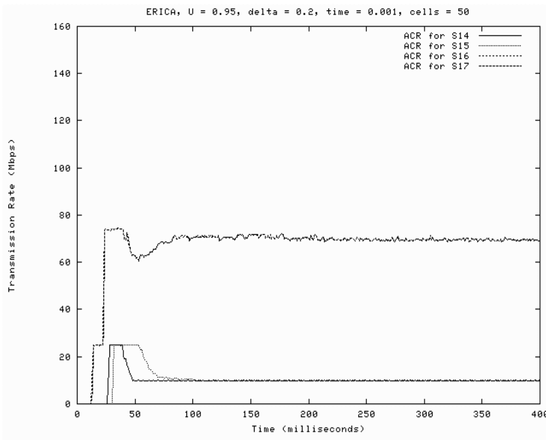


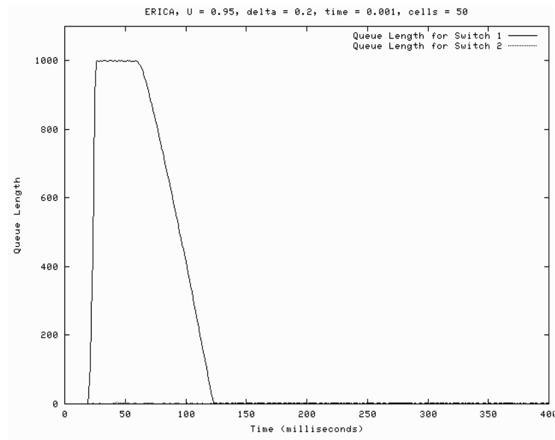
Figure 4: "Upstream Traffic" Network Scenario

The simulation results for ERICA on the WAN upstream traffic scenario are shown in Figure 5a and 5b. The simulation results in Figure 5a show that all the sources receive their max-min fair bandwidth allocations. Sources S1 to S15 each receive 9.8 Mbps while sources S16 and S17 oscillate slightly around a 68.9 Mbps ACR, despite their different initial cell rates. Queueing at the bottleneck link (Figure 5b) shows a large transient at connection startup, which actually fills and briefly overflows the available buffer space. This happens due to the use of SVC's in the ATM-TN simulator, the long round trip times (tens of milliseconds) for sources in this WAN configuration, and the RIF=1 assumption.

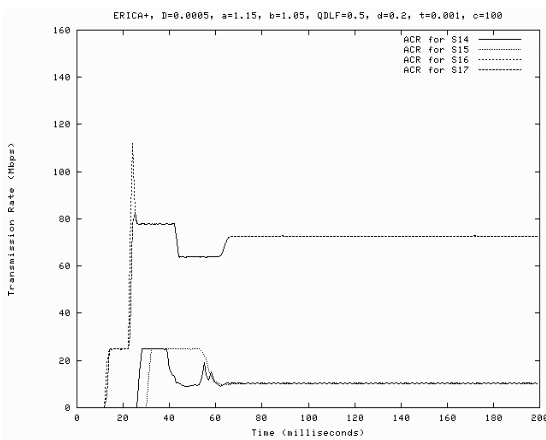
The simulation results for ERICA+ on this scenario are shown in Figure 5c and 5d. ERICA+ performs properly on this scenario. The simulation results in Figure 5c show



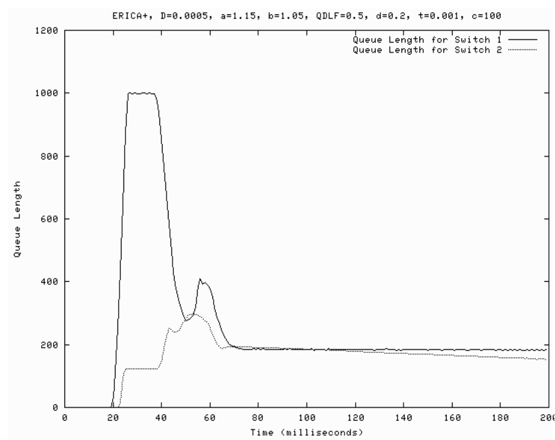
a) ERICA: ACR



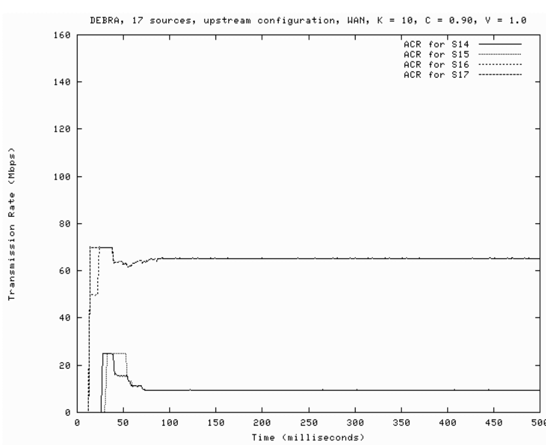
b) ERICA: Queue Length



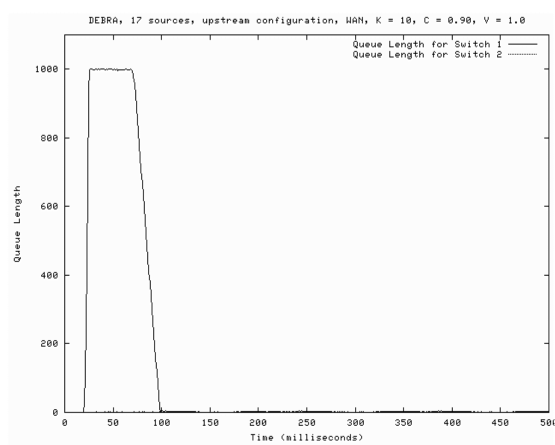
c) ERICA+: ACR



d) ERICA+: Queue Length



e) DEBRA: ACR



f) DEBRA: Queue Length

Figure 5: Simulation Results on Upstream Traffic Network Scenario

that all the sources receive their fair share bandwidth allocation: sources S1 to S15, each receive 10.37 Mbps, while sources S16 and S17, each receive 72.56 Mbps despite different ICRs. Queueing at the bottleneck link (Figure 5d) again shows a large transient at connection startup, overflowing the available buffer space. However, the queue size converges to a steady state of about 200 cells (about 500 microseconds, the target queuing delay in this scenario).

Figure 5e and 5f show the simulation results for DEBRA on the WAN upstream traffic scenario. Figure 5e shows that all the sources receive their fair share bandwidth allocation. Sources S1 to S15 each receive 9.33 Mbps while sources S16 and S17 each receives 65.32 Mbps, despite their different initial cell rates. Queueing at the bottleneck link fills and overflows the available buffer space at connection startup. This happens due to the aggressiveness of the DEBRA algorithm, and very large round trip times (tens of milliseconds) for sources in this WAN configuration.

All three algorithms perform similarly on this scenario. DEBRA performs as well as ERICA+, and slightly better than ERICA on this scenario, due to ERICA's slight oscillations in ACRs and link utilisations. The convergence times of the three algorithms are similar.

4.3 Generic Fairness Configuration 1 Network Scenario

The Generic Fairness Configuration 1 (GFC1) is a five-switch "parking-lot" WAN network configuration (Figure 6) used by ATM Forum [3]. The purpose of this scenario is to test ABR control schemes for the max-min fairness of bandwidth allocations among traffic sources with different bottleneck links and different round trip times. There are 23 traffic sources in this scenario: three A's, three B's, three C's, six D's, six E's and two F's. Each A and D source should receive about 5.55 Mbps, since three A sources share a 50 Mbps bottleneck link with six D sources. Each of the B and E sources should receive about 11.1 Mbps, since three B sources and six E sources share a 100 Mbps bottleneck link. Each C source should receive 33.3 Mbps, since 100 Mbps is left unused by the A and B sources. Similarly, the two F sources should receive 50 Mbps each.

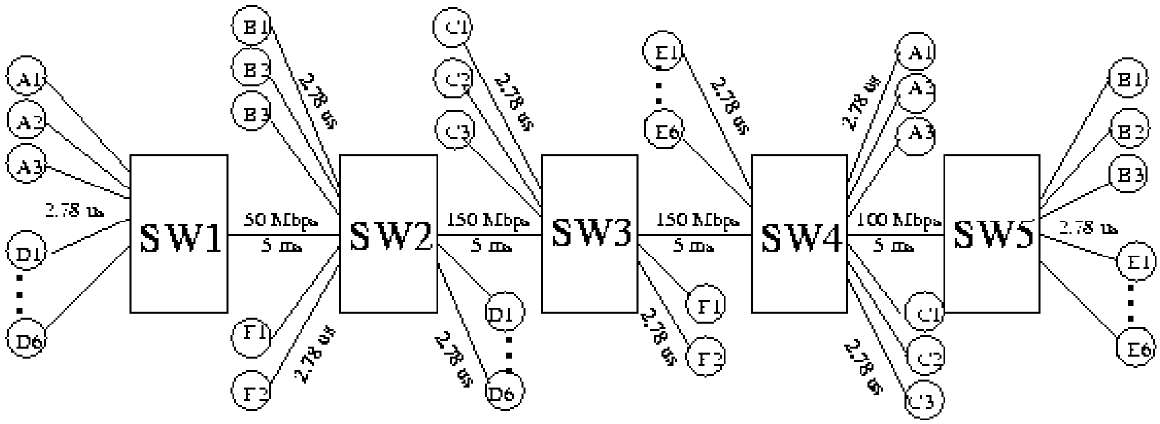
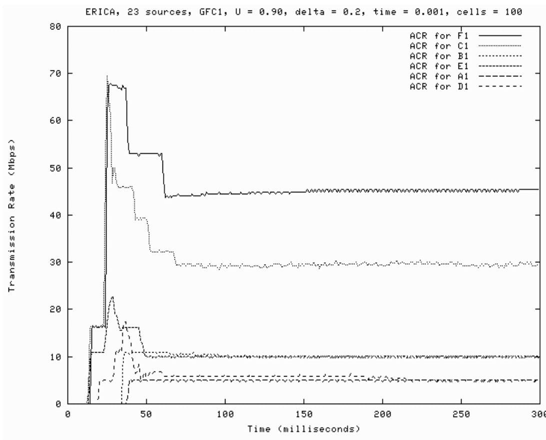


Figure 6: "Generic Fairness Configuration 1 " Network Scenario

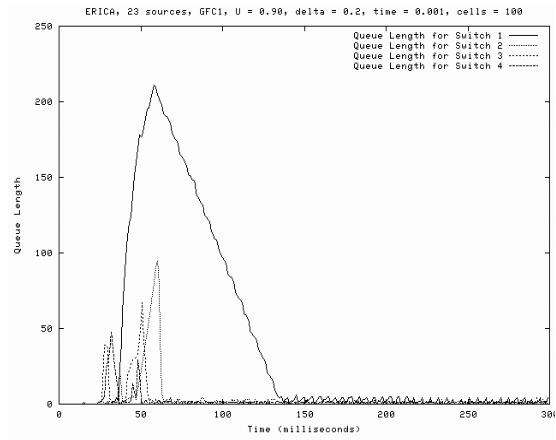
The simulation results for ERICA on Generic Fairness Configuration 1 (GFC1) are shown in Figure 7a and 7b. Figure 7a shows that ERICA achieves max-min fairness on this scenario (the A and D sources each receive 5 Mbps, the B and E sources each receive 10 Mbps, the C sources each receive 30 Mbps, and the F sources each receive 45 Mbps). Queueing behavior shows an initial queue buildup, but then queue size converges to the steady-state (0-4 cells).

The simulation results for ERICA+ on this scenario are shown in Figure 7c and 7d. Figure 7c shows that ERICA+ achieves max-min fairness on this scenario, although higher rate sources (C and F) oscillate around their fair share values during the steady-state. The A and D sources each receive 5.55 Mbps, the B and E sources each receive 11.1 Mbps, the C sources each receive about 33.4 Mbps, and the F sources each receive 50 Mbps. In this scenario, queue size converges to higher values as required for WAN configurations [16] (e.g., target queuing delay, D=0.002, corresponds to about 710 cells on the 150 Mbps link) (Figure 7d).

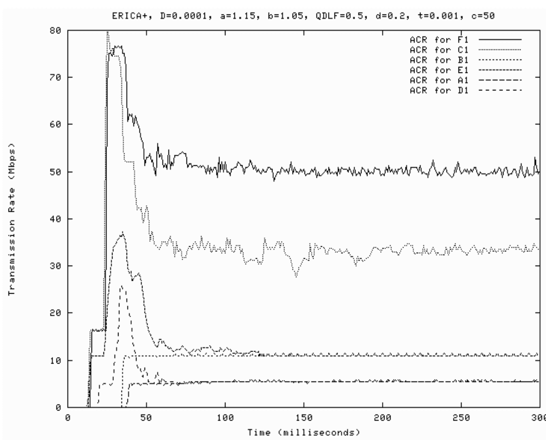
The simulation results for DEBRA on GFC1 are shown in Figure 7e and 7f. As Figure 7e shows, DEBRA achieves max-min fairness on this scenario (the A and D sources each receive 5.27 Mbps, the B and E sources each receive 10.55 Mbps, the C sources each receive 31.7 Mbps, and the F sources each receive 47.5 Mbps). Queueing behavior is



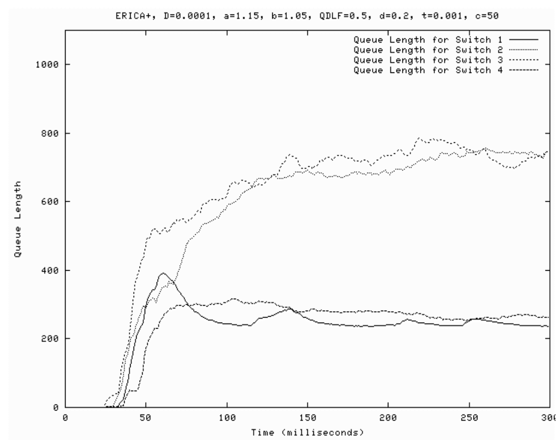
a) ERICA: ACR



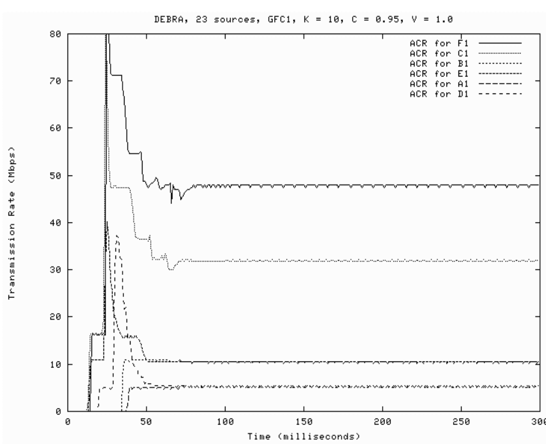
b) ERICA: Queue Length



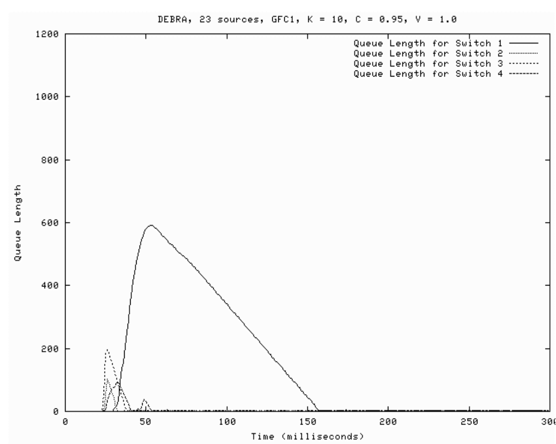
c) ERICA+: ACR



d) *ERICA+: Queue Length*



e) DEBRA: ACR



f)DEBRA: Queue Length

Figure 7: Simulation Results on GFC1 Network Scenario

shown in Figure 7f. Except the initial queue buildup at Switch 1, the queue size converges to the steady-state (0-2 cells) at all the switches.

Compared to the performance of ERICA and ERICA+ on the same scenario, DEBRA needs more time to converge to a steady state (150 milliseconds vs 70 milliseconds for ERICA and ERICA+). However, ERICA and ERICA+ do not perform as well as DEBRA during the steady-state period (i.e., both show some oscillation in ACR for the higher rate sources (C and F)).

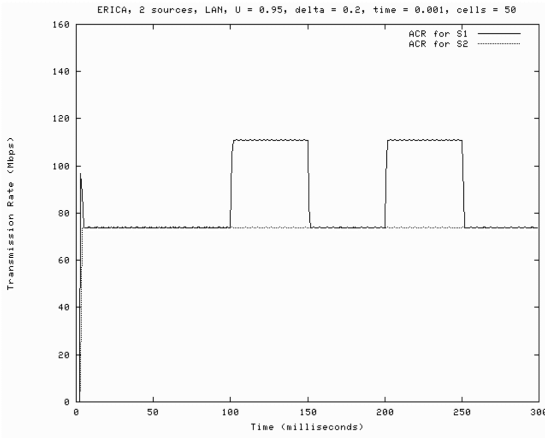
5 Simulation Results: Robustness Testing

The simulation results in two network scenarios, namely Dishonest Sources and Honest Sources–One High, are presented in this section.

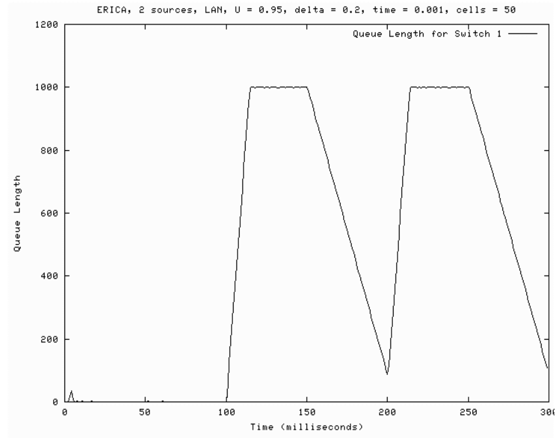
5.1 Dishonest Sources Network Scenario

The first robustness scenario considers a traffic source that lies about its rate of transmission. That is, this scenario assumes that one of the traffic sources is dishonest and greedy, meaning that the traffic source transmits at a rate higher than it is supposed to (higher than the fair share), but it does not inform the switches (i.e., the source is not inserting its actual CCR into outgoing RM cells). In particular, for the first 100 milliseconds both sources are behaving properly, and then, at the time 100 milliseconds, source S1 starts misbehaving by being greedy (transmitting at a rate 50% larger than its share) and dishonest for 50 milliseconds, then being "normal" for next 50 milliseconds, and continuing with this alternating pattern until the end of the simulation.

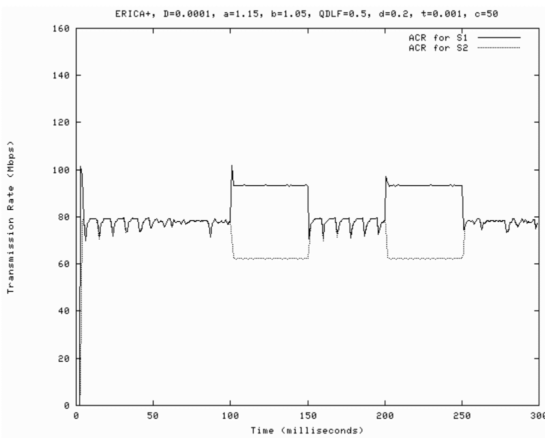
The simulation results on this scenario are shown in Figure 8. The results show that none of the ABR flow control schemes performs properly when there is a dishonest greedy source. That is, source S1 is able to gain advantage over source S2, simply by transmitting at a higher rate than claimed in its RM cells. In all three ABR flow control schemes, the number of cells received at D1 is higher than at D2 [19].



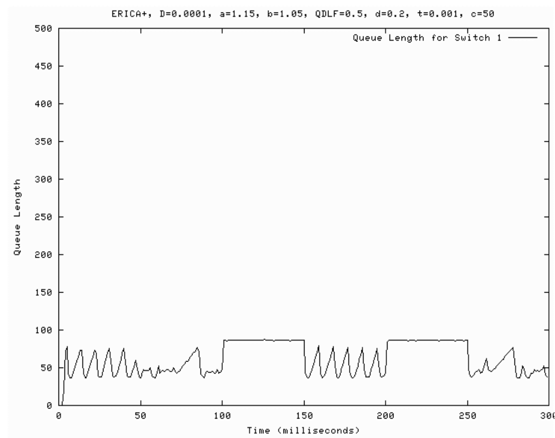
a) ERICA: ACR



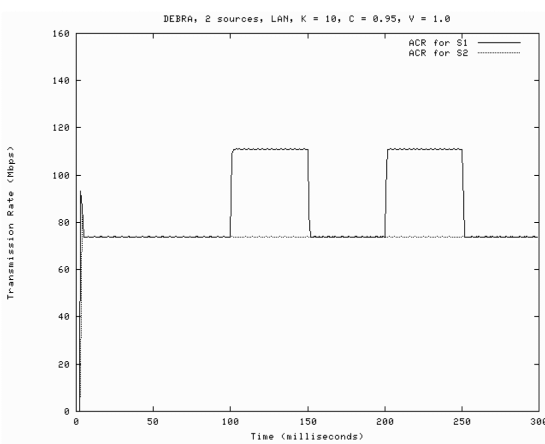
b) ERICA: Queue Length



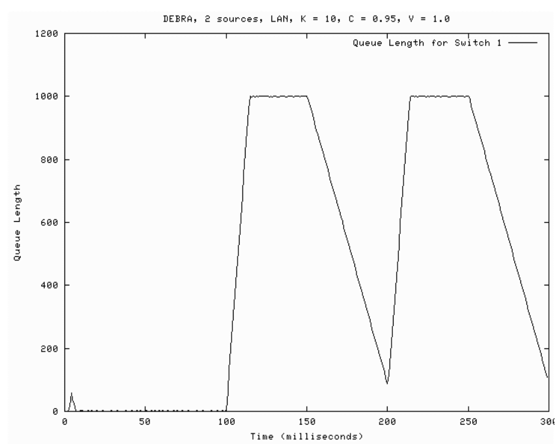
c) ERICA+: ACR



d) ERICA+: Queue Length



e) DEBRA: ACR



f) DEBRA: Queue Length

Figure 8: Simulation Results on Dishonest Sources Network Scenario

There are two distinctly different ways in which the dishonest greedy sources impact the network in the three ABR algorithms. In ERICA and DEBRA, the ACR of S2 (Figure 8a and 8e respectively) is unaffected by the behaviour of S1, but a substantial queue builds up (and overflows) while S1 is greedy (Figure 8b and 8f). This causes significant cell loss for both sources. The second type of effect occurs for the ERICA+ algorithm where the overall queue length is controlled (Figure 8d), but stability is achieved by reducing the ACR of source S2 (Figure 8c). In all three cases link utilisation reaches 100% , and source S1 achieves higher throughput than source S2 [19].

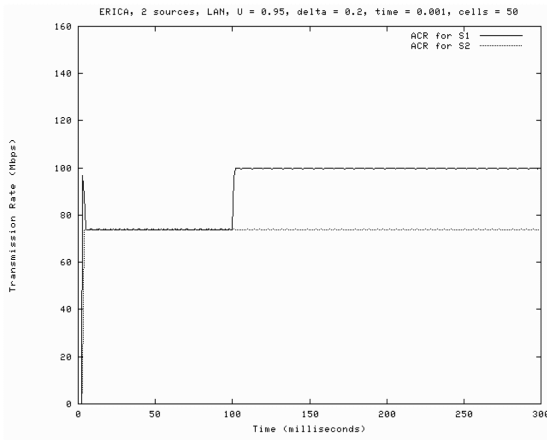
Since ERICA+ uses queue occupancy as a control mechanism, it is more focused on congestion avoidance than on fairness. It reacts to an excessive load from source S1 by decreasing the transmission rate of source S2, heading to unfairness (Figure 8c).

ERICA and DEBRA are less unfair on this scenario than ERICA+ (i.e., the source S2 transmission rate is unaffected, and the throughput advantage of source S1 is less than for ERICA+). However, ERICA and DEBRA produce queue overflow (Figure 8a and 8e), and CLR of about 18%, during the critical 50 millisecond period. This scenario highlights the adverse impact of dishonest sources on ABR traffic control.

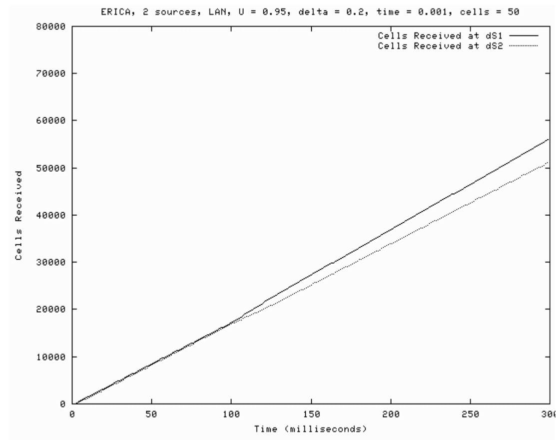
5.2 Honest Sources –One High Network Scenario

The second robustness test assumes that one of the traffic sources decides to transmit at a higher rate than its fair share, and informs the switches of this (i.e., the source inserts its actual CCR into RM cells). In this scenario, both sources behave properly for the first 100 milliseconds, and then, at time 100 milliseconds, source S1 starts transmission at the 100 Mbps, which is higher than its fair share. Source S1 continues using this rate until the end of the simulation.

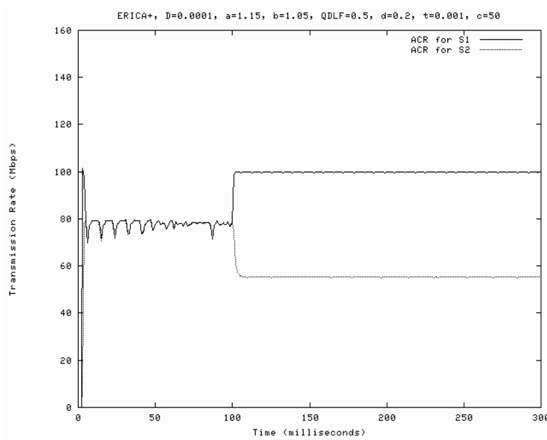
The simulation results on this scenario for all three flow control schemes are shown in Figure 9 (ACR performance and cells received). ERICA behaves the same on this scenario as it does on the network scenario with dishonest sources. Even though the algorithm is aware of the misbehaving source S1, ERICA does not prevent the source from obtaining extra bandwidth. Figure 9a shows that source S1 uses 100 Mbps



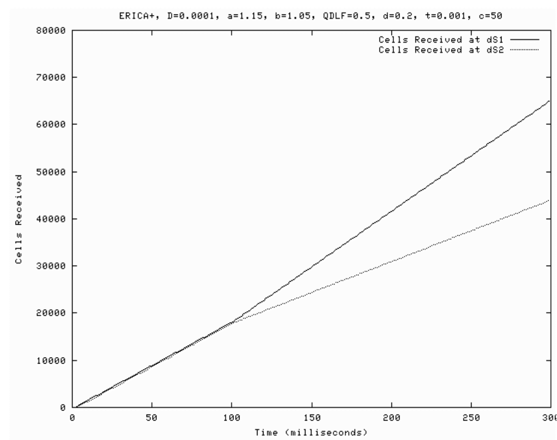
a) ERICA: ACR



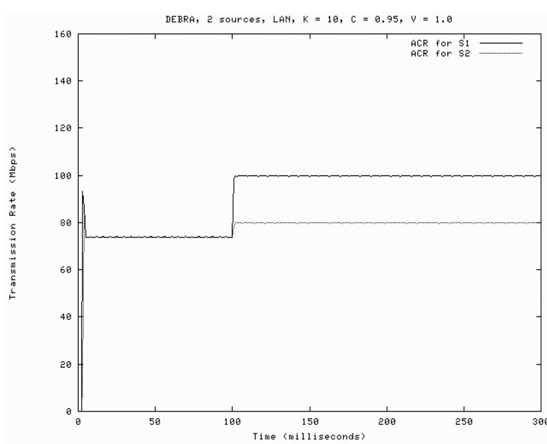
b) ERICA: Cells Received



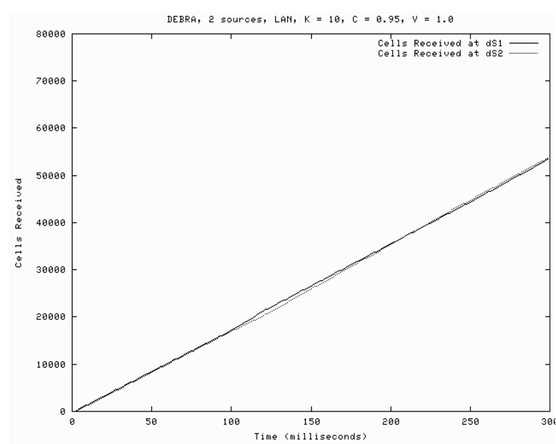
c) ERICA+: ACR



d) ERICA+: Cells Received



e) DEBRA: ACR



f) DEBRA: Cells Received

Figure 9: Simulation Results on Honest Sources-One High Network Scenario

transmission rate, from time 100 milliseconds until the end of the simulation, while source S2 transmits at the same rate (73.8 Mbps) throughout the simulation. The algorithm experiences queue overflow from 100 milliseconds until the end of the simulation (not shown). The CLR for S1 is slightly under 0.08, while CLR for source S2 is slightly over 0.07 (relative to their ACR). However, the number of received cells at the destinations (Figure 9b) is higher for D1 than for D2, showing that fairness is not completely achieved. The reason for this is that source S1 uses an additional 5% of link bandwidth (i.e., the difference between target link utilisation (95%) and physical link bandwidth (100%)).

ERICA+ on this scenario performs identically to its performance on the network scenario with dishonest sources. It does not experience any queue overflow nor cell loss, while achieving target 100% utilisation. However, the algorithm exhibits noticeable unfairness. In order to avoid overflow and congestion, the algorithm decreases the transmission rate of source S2 (Figure 9c), which causes a large difference in number of delivered cells at the destinations (Figure 9d).

DEBRA, unlike the previous algorithms, performs as expected on this network scenario. Although the transmission rate of source S1 (after increasing its rate) stays at 100 Mbps throughout the simulation (Figure 9e), DEBRA preserves fairness by delivering the same number of cells at destinations D1 and D2 (Figure 9f). The algorithm achieves fairness on this scenario by using two features to react to the rate increase of source S1: first, the transmission rate of source S2 increases to respond to the increase by source S1 (Figure 9e), and second, the algorithm discards only cells transmitted from the source S1, achieving CLR of 0.15. The algorithm thus punishes greedy sources by discarding their cells selectively, and protects those who cooperate, by allowing them to increase their rate up to the fair share. This property is inherited from the loss-load curve congestion control mechanism [17].

6 Conclusions

In order to evaluate and compare ABR congestion control strategies, one must illustrate and evaluate various properties of each algorithm, such as efficiency, fairness, responsiveness, scalability, and robustness. Scenarios should consider LAN and WAN topologies, as well as cooperative and uncooperative ABR sources. The benchmarks described in this paper provide a wide variety of network configurations to evaluate ABR flow control algorithms.

The simulation results in this paper show that none of the ABR flow control schemes evaluated is perfect. Generally, ERICA+ performed better than ERICA on the basic set of network configurations and on the set of network configurations for testing robustness. It did not experience as much instability during steady-state and as many buffer overflows as ERICA did.

The results also show that DEBRA, a new explicit-rate ABR flow control scheme, is very competitive. It performed as well as ERICA+ on the basic set of network scenarios, and better than ERICA+ on the set of network configurations for testing robustness of ABR control schemes. Further work is warranted to evaluate its potential for ABR traffic control.

References

[1] M. Arlitt, Y. Chen, R. Gurski, and C. Williamson, "Traffic Modelling in the ATM-TN TeleSim Project: Design, Implementation, and Performance Evaluation", *Proceedings of the 1995 Summer Computer Simulation Conference (SCSC'95)*, Ottawa, Ontario, pp. 847-851, July 1995.

[2] A. Arulambalam, X.Chen, and N. Ansari, "An Intelligent Explicit Rate Control Algorithm for ABR Service in ATM Networks", *Proceedings of ICC '97*, Montreal, Canada, pp. 200-204, June 1997.

[3] The ATM Forum Technical Committee, "Traffic Management Specification Version 4.0", *af-tm-0056.000*, April 1996. URL: http://ftp.atmforum.com/pub/approved-specs/aftm-0056.000.ps.

[4] F. Chiussi, and Y. Wang: "An ABR Rate-Based Congestion Control Algorithm for ATM Switches with Per-VC Queuing", *Proceedings of IEEE GLOBECOM'97*, Phoenix, AZ, pp. 771-778, November 1997.

[5] F. Chiussi, Y.Xia and V. Kumar, "Dynamic Max Rate Control Algorithm for ABR Service in ATM Networks", *Proceedings of GLOBECOM '96*, London, UK, pp. 2108-2117, November 1996.

[6] R. Gurski and C. Williamson, "DEBRA: Dynamic Explicit Bid Rate Algorithm for ATM ABR Traffic", *TeleSim Project Technical Report*, University of Calgary, November 1996.

[7] R. Jain, S. Kalyanaraman, S. Fahmy, R. Goyal, and S. Kim, "Source Behavior for ATM ABR Traffic Management: An Explanation", *IEEE Communications Magazine*, Vol. 34, No. 11, pp. 50-57, November 1996.

[8] 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", *Paper submitted to the IEEE/ACM Transactions on Networking*, January 1997.

[9] R. Jain, S. Fahmy, S. Kalyanaraman, and R. Goyal, "The ERICA Switch Algorithm for ABR Traffic Management in ATM Networks, Part II: Requirements and Performance Evaluation", *Paper submitted to the IEEE/ACM Transactions on Networking*, January 1997.

[10] H. Kanakia, S. Keshav, and P. Mishra, "A Comparison of Reactive Host-Based Flow Control Schemes in Reservationless Networks", *Technical Memorandum*, AT&T Bell Laboratories, July 1992. URL: http://simon.cs.cornell.edu/Info/People/skeshav/92/3-12.ps.

[11] A. Kolarov and G. Ramamurthy, "Comparison of explicit rate and explicit forward congestion indication flow control schemes for ABR service in wide area networks", *Performance Evaluation- An International Journal, No. 31*, pp. 89 –106, October 1997.

[12] Y. Moret and S. Fdida, "ERAQLES: An Efficient Explicit Rate Algorithm for ABR", *Proceedings of IEEE GLOBECOM'97,* Phoenix, AZ, pp. 801-805, November 1997.

[13] T. Ott and N. Aggarwal, "TCP over ATM: ABR or UBR?", *Proceedings of the 1997 ACM SIGMETRICS Conference*, Seattle, WA, pp. 52-63, June 1997.

[14] L. Roberts, "Enhanced PRCA (Proportional Rate Control Algorithm)", *ATM Forum Contribution 94-0735*, August 1994.

[15] TeleSim Project, Available at URL: http://www.wnet/ca/telesim/

[16] B. Unger, F. Gomes, X. Zhonge, P. Gburzynski, T. Ono-Tesfaye, S. Ramaswamy, C. Williamson, and A. Covington, "A High Fidelity ATM Traffic and Network Simulator", *Proceedings of the 1995 Winter Simulation Conference*, Arlington, VA, pp. 996-1003, December 1995.

[17] C. Williamson, "Dynamic Bandwidth Allocation Using Loss-Load Curves", *IEEE/ACM Transactions on Networking*, vol.4, No. 6, pp. 829-839, December 1996.

[18] C. Williamson, "A Performance Study of ABR Traffic Control Strategies Using the ATM-TN Simulator", *TeleSim Project Report*, Department of Computer Science, University of Saskatchewan, April,1998.

[19] M. Zoranovic, "Performance Evaluation of Explicit-Rate ABR Flow Control Schemes", *Master of Science Thesis,* Department of Computer Science, University of Saskatchewan, Saskatoon, April 1999.