CHAPTER 4

SURVEY OF ATM SWITCH CONGESTION CONTROL SCHEME PROPOSALS

In this chapter, we shall look at a number of ATM rate-based feedback congestion control scheme proposals. Prior to the development of rate-based mechanisms, there was a prolonged debate in the ATM forum between the hop-by-hop credit (or window) based framework and the end-to-end rate-based framework [69]. We will briefly summarize this debate and concentrate on the survey of rate-based switch algorithms. For each algorithm, we will identify the key contributions and present its drawbacks. This will lay a foundation for comparison with the OSU, ERICA and ERICA+ schemes developed in this dissertation. It should be noted that several of these schemes were designed after the ERICA scheme was developed, and therefore, may exhibit some overlap of concepts. This chapter uses the terminology developed in section 3.

4.1 Credit-Based Framework

The credit-based framework was proposed by Professor H. T. Kung, it was supported by Digital, BNR, FORE, Ascom-Timeplex, SMC, Brooktree, and Mitsubishi [86, 26]. The approach bears some similarity in concept to the sliding window-based

protocols used in data link control protocols. The framework consists of a per-link, per-VC window flow control. Each link consists of a sender node (which can be a source end system or a switch) and a receiver node (which can be a switch or a destination end system). Each node maintains a separate queue for each VC. The receiver monitors queue lengths of each VC and determines the number of cells that the sender can transmit on that VC. This number is called "credit." The sender transmits only as many cells as allowed by the credit.

If there is only one active VC, the credit must be large enough to allow the whole link to be full at all times. In other words:

$$\text{Credit} \geq \text{Link Cell Rate} \times \text{Link Round Trip Propagation Delay}$$

The link cell rate can be computed by dividing the link bandwidth in Mbps by the cell size in bits.

The scheme as described so far is called "Flow Controlled Virtual Circuit (FCVC)" scheme. There are two problems with this initial static version. First, if credits are lost, the sender will not be aware of it. Second, each VC needs to reserve the entire round trip worth of buffers even though the link is shared by many VCs. These problems were solved by introducing a credit resynchronization algorithm and an adaptive version of the scheme.

The credit resynchronization algorithm consists of both sender and receiver maintaining counts of cells sent and received for each VC and periodically exchanging these counts. The difference between the cells sent by the sender and those received by the receiver represents the number of cells lost on the link. The receiver reissues that many additional credits for that VC.

The adaptive FCVC algorithm [85] consists of giving each VC only a fraction of the round trip delay worth of buffer allocation. The fraction depends upon the rate at which the VC uses the credit. For highly active VCs, the fraction is larger while for less active VCs, the fraction is smaller. Inactive VCs get a small fixed credit. If a VC doesn't use its credits, its observed usage rate over a period is low and it gets smaller buffer allocation (and hence credits) in the next cycle. The adaptive FCVC reduces the buffer requirements considerably but also introduces a ramp-up time. If a VC becomes active, it may take some time before it can use the full capacity of the link even if there are no other users.

4.2 Rate-Based Approach

This approach, which was eventually adopted as the standard was proposed originally by Mike Hluchyj and was extensively modified later by representatives from twenty two different companies [29].

Original proposal consisted of a rate-based version of the DECbit scheme [63], which consists of end-to-end control using a single-bit feedback from the network. Initially, sources send data at an negotiated "Initial Cell rate." The data cells contain a bit called the EFCI bit in the header. The switches monitor their queue lengths and, if congested, set the EFCI bit in the cell headers. The destination monitors these indications for a periodic interval and sends an RM cell back to the source. The sources use an additive increase and multiplicative decrease algorithm to adjust their rates.

This is an example of a "bit-based" or "binary" feedback scheme. As we shall see later in this section, it is possible to give explicit rate feedback in the rate-based

framework. The complete framework has been treated in the chapter introducing ATM traffic management 2.

4.3 Binary Feedback Schemes

4.3.1 Key Techniques

Binary feedback schemes essentially use a single bit feedback. The initial binary feedback algorithm used a "negative polarity of feedback" in the sense that RM cells are sent only to decrease the source rate, and no RM cells are required to increase the rate. A "positive polarity," on the other hand, would require sending RM cells for increase but not on decrease. If RM cells are sent for both increase and decrease, the algorithm would be called "bipolar."

The problem with negative polarity is that if the RM cells are lost due to heavy congestion in the reverse path, the sources will keep increasing their load on the forward path and eventually overload it.

This problem was fixed in the next version by using positive polarity. The sources set EFCI on every cell except the nth cell. The destination will send an "increase" RM cell to source if they receive any cells with the EFCI off. The sources keep decreasing their rate until they receive a positive feedback. Since the sources decrease their rate proportional to the current rate, this scheme was called "proportional rate control algorithm (PRCA)."

PRCA was found to have a fairness problem. Given the same level of congestion at all switches, the VCs traveling more hops have a higher probability of having EFCI set than those traveling smaller number of hops. If p is the probability of EFCI being set on one hop, then the probability of it being set for an n-hop VC is $1-(1-p)^n$ or

np. Thus, long path VCs have fewer opportunities to increase and are beaten down more often than short path VCs. This was called the "beat-down problem [12]."

One solution to the beat down problem is the "selective feedback" [98] or intelligent marking [9] in which a congested switch takes into account the current rate of the VC in addition to its congestion level in deciding whether to set the EFCI in the cell. The switch computes a "fair share" and if congested it sets EFCI bits in cells belonging to only those VCs whose rates are above this fair share. The VCs with rates below fair share are not affected.

The RM cell also contains two bits called the "Congestion Indication" bit and the "No Increase" bit. Schemes which mark these bits are not necessarily classified as "binary schemes" since they may feedback more information than just one bit. Several ER-based schemes like CAPC2 and DMRCA (discussed later in this chapter) also use the CI and NI bit setting options.

4.3.2 Discussion

The attractive features about the binary schemes are:

- Simple to implement. Requires only a bit in the header. Feedback calculation is also typically simple. The multiple round trip times required for convergence is not a problem for local area networks because the round trip delay for these networks is in the order of microseconds.

- Cost effective option to introduce ABR in LANs.

- Typical bit-based schemes look only at the queue length as a metric for congestion. Though we point out later that the queue length is not an accurate

metric of congestion, using a single metric localizes the errors possible. Further, since the end-result of all errors is additional queues, taking the queue length as a metric is a safe method to avoid divergence, even when the demand and capacity are variable.

The drawbacks of this approach are:

- The bit-based feedback schemes were initially designed for low-speed networks. Since a bit gives only two pieces of information ("up" or "down"), the system may take several round trips to converge to stable values. That is, the transient convergence period is long. During this period, the network might either be underutilized, or queues might build up, which is a definite concern in high-speed networks.

- The bit-based feedback was originally designed for window-flow control where the maximum queue is simply the sum of all source windows. In rate control, if the sum of the source rates is larger than the capacity, the queues could grow to infinity unless the rates are changed [67, 65]. The transient convergence period determines what this worst case queue will be. The queue can be large (proportional to the steady state queue plus a term proportional to the transient convergence time) when a new source starts up after the system is in the steady state.

- The bit-based feedback was originally designed for connectionless networks where it is possible that packets from a source to a destination may take multiple paths. Hence, it is not a good idea to give authoritative feedback information based on partial knowledge. On the other hand, switches in connection-oriented

networks like ATM can have complete knowledge about a flow based upon measurement, and can give authoritative feedback.

- The steady state itself exhibits oscillatory behavior in terms of queue length and rate allocations. The reason for the behavior is that the control is based upon the queue length, a highly variable quantity in rate-based control. Further, when the queue length is zero, or beyond a threshold, the schemes essentially "guess" the allocations due to the unreliability of the metric.

- The technique requires several parameters to force convergence. The system is also quite sensitive to the parameter settings.

- The buffer requirement at switches is large, and increases linearly with the number of connections.

- The "beat-down" fairness problem needs to be solved in every implementation. This adds to the complexity at switches.

A theme we gather from the above observations is that the bit-based feedback and the technique of using queue length as a congestion indicator is a legacy from the window-based control schemes for low-speed, connectionless networks. The adaptation to rate-based, high-speed, connection-oriented networks like ATM has some advantages in terms of simplicity, and hence cost-effectiveness. But, the performance in the Wide Area scenario leaves a lot to be desired. This led to the introduction of explicit rate feedback schemes.

4.4 Explicit Rate Feedback Schemes

In July 1994, Charny, Clark and Jain [20] argued that the binary feedback was too slow for rate-based control in high-speed networks and that an explicit rate indication would not only be faster but would offer more flexibility to switch designers.

In addition to providing a solution to the problems of the bit-based feedback schemes described in the previous section, explicit rate schemes are attractive for other reasons. First, policing is straight forward. The entry switches can monitor the returning RM cells and use the rate directly in their policing algorithm. Second with fast convergence time, the system come to the optimal operating point quickly. Initial rate has less impact. Third, the schemes are robust against errors in or loss of RM cells. The next RM cell carrying "correct" feedback will bring the system to the correct operating point in a single step.

Further, one of the reasons for choosing the rate-based framework was that ABR could be used for applications other that just data applications - to provide a cost-effective alternative to applications that traditionally use higher priority classes. Typical applications are compressed video, which could tolerate variable quality. The explicit rate schemes could reduce the variation in the rates seen at the end-systems, higher throughput and a controlled delay through the network. Further, the video applications could directly use the rate values to tune their parameters as opposed to the credit value, which cannot be directly used without knowledge of the round trip delay.

In the following sections we survey several rate-based explicit feedback schemes. In each section, we will have a brief discussion of the key techniques used by the

scheme followed by a discussion of the contributions and drawbacks of the proposed scheme.

4.5 MIT Scheme

The explicit rate approach was substantiated with a scheme designed by Anna Charny during her master thesis work at the Massachusetts Institute of Technology (MIT) [20, 19].

4.5.1 Key Techniques

The MIT scheme consists of each source sending an control (or RM) cell every n th data cell. The RM cell contains the VC's current cell rate (CCR) and a "desired rate." The switches monitor all VC's rates and compute a "fair share." Any VC's whose desired rate is less than the fair share is granted the desired rate. If a VC's desired rate is more than the fair share, the desired rate field is reduced to the fair share and a "reduced bit" is set in the RM cell. The destination returns the RM cell back to the source, which then adjusts its rate to that indicated in the RM cell. If the reduced bit is clear, the source could demand a higher desired rate in the next RM cell. If the bit is set, the source uses the current rate as the desired rate in the next RM cell.

The switches maintain a list of all of its VCs and their last seen desired rates. All VCs whose desired rate is higher than the switch's fair share are considered "overloading VCs." Similarly, VCs with desired rate below the fair share are called "underloading VCs." The underloading VCs are bottlenecked at some other switch and, therefore, cannot use additional capacity at this switch even if available.

The capacity unused by the underloading VCs is divided equally among the overloading VCs. Thus, the fair share of the VCs is calculated as follows:

$$\text{Fair Share} = \frac{\text{Capacity} - \sum \text{Bandwidth of underloading VCs}}{\text{total number of VCs} - \text{Number of underloading VCs}}$$

It is possible that that after this calculation some VCs that were previously underloading with respect to the old fair share can become overloading with respect to the new fair share. In this case these VCs are re-marked as overloading and the fair share is recalculated.

Charny [19] has shown that two iterations are sufficient for this procedure to converge. Charny also showed that the MIT scheme achieves max-min optimality in $4k$ round trips, where k is the number of bottlenecks.

4.5.2 Discussion

The contributions of the MIT scheme were as follows:

- Help define the framework for explicit rate feedback mechanisms in the ATM ABR specifications
- Provided a reference iterative algorithm
- Max-min fairness is achieved because the underloading VCs see the same advertised rate
- The switch algorithm is essentially a rate calculation algorithm which is not concerned with the enforcement of the rates. The enforcement of rates may be carried out either at the edge of the network or at every network switch though queuing and scheduling policies. This algorithm gives the network designer the

flexibility of decoupling the enforcement and feedback calculation. This aspect has since become a standard feature in all schemes developed.

- The algorithm quickly adapts to dynamic changes in the network provided the declared values of the parameters "desired rate" etc are accurate. The algorithm is shown to be "self-stabilizing" in the sense that it recovers from any past errors, changes in the set of network users, individual session demands and session routes.

- The algorithm provides fast convergence to max-min rates (within $4k$ round trips, where k is the number of bottlenecks).

- Charny also shows that the algorithm is "well-behaved" in transience, i.e., given an upper bound on the round-trip delay, the actual transmission rates can be kept *feasible* throughout the transient stages of the algorithm operation while still providing reasonable throughput to all users. A feasible set of rate allocations ensures that a rate allocation is such that no link capacity is exceeded. The arguments assumed synchronization among sources, or a special source policy which forces synchronization in the asynchronous case.

The drawbacks of the scheme were:

- The computation of the fairshare requires order n operations, where n is the number of VCs. The space requirements of the scheme are also order n.

- The feedback procedure is unipolar, i.e., switches only reduce the rates of sources. As a result, the sources require an extra round trip for increase. This feature is addressed in the Precise Fair Share Computation option of the OSU

scheme which provides a bipolar feedback (i.e., switch can increase as well as decrease the desired rates).

- If the sources do not use their allocations, or temporarily go idle, there is no mechanism prescribed to detect this condition. Since the scheme relies on the declared values and does not measure the source rates, nor the offered load, it is possible that the offered load is very different from the sum of the desired rate values, leading to underutilization.

- There is no policy prescribed to drain queues built up during transient periods, or errors in feedback.

- The scheme as described is not compatible with the current ATM Forum standard, and requires minor realignment to be compatible.

This proposal was well received, and considered a baseline for other schemes to be compared with. The key exception was that the computation of fair share requires order n operations, where n is the number of VCs. The space requirements of the scheme are also order n. This set off a search for schemes which were O(1) both in time and space complexity. This led to the EPRCA, the OSU scheme proposals in September 1994, and later the CAPC2 proposal in late 1994. We continue our survey looking at these schemes. The OSU scheme and ERICA schemes will be treated in separate chapters of this dissertation.

4.6 EPRCA and APRC

The merger of PRCA with explicit rate scheme lead to the "Enhanced PRCA (EPRCA)" scheme at the end of July 1994 ATM Forum meeting [106].

4.6.1 Key Techniques

In EPRCA, the sources send data cells with EFCI set to 0. After every n data cells, they send an RM cell. The RM cells contain desired explicit rate (ER), current cell rate (CCR), and a congestion indication (CI) bit. The sources initialize the ER field to their peak cell rate (PCR) and set the CI bit to zero.

The destinations monitor the EFCI bits in data cells. If the last seen data cell had EFCI bit set, they mark the CI bit in the RM cell.

In addition to setting the explicit rate, the switches can also set the CI bit in the returning RM cells if their queue length is more than a certain threshold. Some versions of the EPRCA algorithm do not set the EFCI bits, and mark the CI and ER fields alone.

The scheme uses two threshold values QT and DQT on the queue length to detect congestion. When the queue length is below QT, all connections are allowed to increase their rate.

When the queue length exceeds QT, the switch is considered congested and performs *intelligent marking*. By intelligent marking, we mean that the switch selectively asks certain sources to increase their rates and certain sources to reduce their rates. In order to do this, the switch maintains the Mean ACR ($MACR$), and selectively reduces the rate of all connections with ACR larger than $MACR$. The switch may reduce the rates by setting the CI bit and/or by setting the ER field of an RM cell when CCR value exceeds $MACR \times DPF$ (DPF is the *Down Pressure Factor*). The DPF is introduced to include those VCs whose rate is very close to $MACR$. Typically DPF is 7/8. The CI bit setting forces the sources to decrease their rate as described in the source end system rules (see chapter 2.

If the port remains congested and the queue length exceeds DQT threshold, the switch is considered heavily congested and all connections have their rate reduced.

To avoid the O(N) computation of the advertised rate, the fairshare is approximated by $MACR$ using the running exponential weighted average, computed every time the switch receives an RM cells,as :

$MACR = MACR(1 - AV) + CCR * AV$

where AV is an averaging factor, typically equal to 1/16, allowing the implementation using addition and shift operations.

4.6.2 Discussion

The contributions of the EPRCA scheme are:

- Introduced a class of algorithms which operate with O(1) space and O(1) time requirements.
- EPRCA allows both binary-feedback switches and the explicit feedback switches on the path, since it bridged the gulf between PRCA and explicit rate schemes. This feature has been incorporated in the ATM Traffic Management standard.
- Uses the mean ACR as the threshold and allocates this rate to all unconstrained VCs. This technique converges to fair allocations when the mean ACR is a good estimate of the "fair share," i.e., the max-min advertised rate (computed by the MIT scheme).

The drawbacks of the EPRCA scheme are:

- If the mean ACR is not a good estimate of the "fair share," then the scheme can result in considerable unfairness [22].

- The exponential averaging of the rates may become biased towards the higher rates. For example, consider two sources running at 1000 Mbps and 1 Mbps. In any given interval, the first source will send 1000 times more control cells than the second source and so the exponentially weighted average is very likely to be 1000 Mbps regardless of the value of the weight used for computing the average.

  The problem is that the exponential averaging technique (which is similar to the arithmetic mean) is not the right way to average a set of ratios (like ACRs = number of cells/time) where the denominators are not equal [66]. We address this averaging issue in the design of ERICA later in this dissertation.

- The scheme uses queue length thresholds for congestion detection. As a result, it effectively "guesses" the rate allocations when the queue value is zero, or above the high threshold. We will argue later in this dissertation that the queue length does not provide full information about the congestion at the switch, and hence is not reliable as the primary metric for rate-based congestion control.

- The scheme uses a number of parameters whose values are typically set conservatively. This technique trades off transient response time (time required to reach the steady state after a change in network conditions). This means that the utilization of the bottlenecks will be lower on the average compared to aggressive allocation schemes. Further, when the network is constantly in the state when demand and capacity are variable (no steady state), the performance

of the scheme is unclear, and is expected to be lower because of the conservative parameter settings.

Researchers at University of California at Irvine (UCI) suggested a solution to the problems of EPRCA through a scheme they developed called "Adaptive Proportional Rate Control" [78]. Essentially, they suggested that the queue growth rate be used as the load indicator instead of the queue length. The change in the queue length is noted down after processing, say, K cells. The overload is indicated if the queue length increases.

However, this approach still suffers from the defect that the metric gives no information when the queue lengths are close to zero (underutilization). Basically, the problem is that the queue length information needs to be combined with the ABR capacity and ABR utilization to get a full picture of the congestion situation at the switch.

4.7 CAPC2

In October 1994, Barnhart from Hughes Systems proposed a scheme called "Congestion Avoidance using Proportional Control (CAPC)[10]."

4.7.1 Key Techniques

This scheme used some of the concepts developed in the OSU scheme and used a phase-locked loop style filter in the algorithm. In this scheme, as in OSU scheme (described later in this dissertation), the switches set a target utilization parameter slightly below 1. This is the ABR capacity utilization the scheme aims to achieve. As in this OSU scheme, the switches measure the input rate and load factor z (which

is the ratio of the input rate to the product of the ABR capacity and the target utilization). The load factor z is used as the primary congestion detection metric as opposed to using the queue length for that purpose. The scheme calculates a single "fairshare" using the load factor as follows.

During underload ($z < 1$), fair share is increased as follows:

$$\text{Fair share} = \text{Fair share} \times \text{Min}(ERU, 1 + (1 - z) * Rup)$$

Here, Rup is a slope parameter in the range 0.025 to 0.1. ERU is the maximum increase allowed and was set to 1.5.

During overload ($z > 1$), fair share is decreased as follows:

$$\text{Fair share} = \text{Fair share} \times \text{Max}(ERF, 1 - (z - 1) * Rdn)$$

Here, Rdn is a slope parameter in the range 0.2 to 0.8 and ERF is the minimum decrease required and was set to 0.5.

The fair share is the maximum rate that the switch will grant to any VC.

This method of using (1- z) (or a term proportional to unused capacity) for feedback calculation is also used by the Phantom [3] described later in this survey.

In addition to the load factor, the scheme also uses a queue threshold. Whenever the queue length is over this threshold, a congestion indication (CI) bit is set in all RM cells. This prevents all sources from increasing their rate and allows the queues to drain out.

4.7.2 Discussion

The CAPC scheme and its successor CAPC2 (which addressed some initialization issues) was proposed in late 1994, before many of the scheme proposals surveyed in this chapter. The contributions of the CAPC scheme include:

- An oscillation-free steady state performance. The frequency of oscillations is a function of $1 - z$, where z is the load factor. In steady state, $z = 1$, the frequency is zero, that is, the period of oscillations is infinite.

- Simple to implement.

- Uses the load factor as the primary metric, and does not use the CCR field.

- The single "fairshare" threshold is similar to the EPRCA concept. This allows the scheme to have an O(1) space complexity and easily converge to fairness under conditions of constant demand and capacity.

The drawbacks of the scheme include:

- The convergence time of the scheme is longer since it uses parameters whose values are chosen conservatively.

- Since the algorithm uses a binary indication bit in very congested states, it is prone to unfair behaviors [3].

4.8 Phantom

This scheme was developed by Afek, Masour and Ostfeld at the Tel-Aviv University [3]. An important design goal in this work is to develop a *constant space* congestion avoidance algorithm, while achieving max-min fairness, and good transient response.

4.8.1 Key Techniques

The key idea is to bound the rate of sessions that share a link by the amount of unused bandwidth on that link. The scheme uses the concept of a *Phantom* session

which shares the link equally as all other connections. The link allocates rates fairly among all sources including the phantom.

Specifically, the variable Δ is defined to be the unused link capacity, i.e.,

Link Capacity $-$

Σ(Rates of sessions that use the link).

It is measured as:

$(Numberofcellstransmittableonlink - Numberofcellsinput)/\tau$

where τ is a fixed time interval.

Observe that Δ can be greater than zero, when the actual queue at the link is non-zero.

The rate of sessions that are above Δ are reduced towards Δ and the rate of sessions that are below Δ may be increased. The mechanism reaches a steady state only when the unused capacity (Δ) is equal to the maximum rate of any session that crosses the link and all the sessions that are constrained by the link are at this rate. So, Δ is the "fairshare" value at each link.

For example [3], if three sessions share a 100 Mbps link, then in the steady state, each session receives 25 Mbps and the link utilization is 75 % ($\Delta = 25Mbps$). However, if two of the three sessions are restricted elsewhere to 10 Mbps each, the third sessions gets 40 Mbps ($\Delta = 40Mbps$).

The scheme addresses five important implementation aspects:

1. **Measuring Δ:** Naive measurement of Δ can be very noisy. The scheme uses exponential averaging to smooth out variance in Δ and accumulates it in a variable called "Maximum Allowed Cell Rate (MACR)":

$$MACR = max(MACR \cdot (1 - \alpha) + \Delta \cdot \alpha, MACR \cdot dec\_factor)$$

The lower bound is required to filter out variations caused by sudden capacity changes.

2. **Sensitivity to queue length:** The scheme recognizes the need to compensate for errors, and transient queues by looking at the absolute value of the queue length. The averaging parameter α is replaced by two parameters α_{inc}, when $\Delta > MACR$, and α_{dec}, when $\Delta \leq MACR$. Both these parameters vary depending upon the queue length.

3. **Utilization:** The utilization may be improved by restricting the bandwidth of connections by $utilization\_factor$ times $MACR$, instead of Δ.

4. **Variance consideration:** The problem with the utilization factor is that $MACR$ may exhibit large oscillations. The scheme therefore smoothes out the factors α_{inc} and α_{dec} based upon the variance in Δ. The algorithm used is similar to the TCP RTT estimation smoothing algorithm.

5. **Reducing maximum queue length:** The scheme also sets the NI bit based on another variable called $Fast\_MACR$ which tracks the variation of the capacity more closely.

4.8.2 Discussion

The contributions of the Phantom scheme are as follows:

- The idea of a phantom connection, combined with the utilization factor can bring the allocations close to max-min. The basic algorithm allocates rates

proportional to the the unused ABR capacity. The inclusion of the utilization factor brings the efficiency closer to the maximum possible.

- The algorithm developed is O(1) in both space and time requirements.
- In the basic scheme, the residual unused capacity to accommodate new sessions without queue buildup.
- Fairness is maintained because, over a period, all sources see the same "advertised rate" (MACR).
- The scheme explicitly addresses the issues in measurement, variance reduction and error compensation. The variance suppression is a necessity for the scheme since the phantom bandwidth, Δ is highly variant.
- The exponential averaging of measurements is valid (unlike the EPRCA algorithm where the technique is dubious) because they are made over fixed intervals. We note again that the arithmetic mean (or exponential averaging) is not the correct method for averaging ratios where the denominator is not constant [66].
- The scheme considers the issues of high bottleneck utilization combined with a systematic method to cope with queuing delays (due to transient queues).
- Fast implementations can be derived by replacing multiply operations by bit-shifting.

The drawbacks of the scheme are:

- The use of the utilization factor introduces higher degree of variation in load, and the possibility of sharp queue spikes. This necessitates complex variance reduction and queue control techniques within the algorithm, and introduces several extra parameters. The scheme may also require the sources to negotiate a lower value of the "Rate Increase Factor (RIF)" parameter to moderate the network-directed rate increases.

- The queue thresholding procedures may require a new set of parameter recommendations for Wide Area Networks. It is not clear whether the scheme will work in WANs without complex parameter changes.

4.9 UCSC Scheme

This scheme was proposed by researchers Kalampoukas and Varma at the University of California, Santa Cruz (UCSC), and K.K. Ramakrishnan of AT&T Research [79].

4.9.1 Key Techniques

The scheme cleverly approximates the MIT scheme with O(1) bookkeeping, and hence brings the computational complexity from O(N) to O(1). Intuitively, the scheme spreads the MIT O(N) iteration over successive RM cells. As a result, the convergence time and buffer requirements are traded off with computational complexity. The space requirements compared to the MIT scheme remain O(N) since the scheme maintains some per-VC state information. In special cases, optimization may be achieved by using shift operations instead of multiply/divide operations.

The key technique in the scheme is the following. On the lines of the MIT scheme, the scheme assumes that the source demands a certain rate, and the switch tries to satisfy that demand. In the scheme, VC_i "requests" bandwidth equal to $min(ER_i, CCR_i)$. We can consider this as the "demand" of VC_i. The same quantity can also be considered as the bandwidth "usage" of the VC. The scheme computes a "maximum bandwidth" value A_{max} depending upon the VC's current state. A_{max} is the *fairshare* which is given to the source as feedback. Next, we describe the states of the VCs and show how they are used to compute the bandwidth allocations.

Each VC_i can be in one of the following two states:

1. **Bottlenecked:** the switch cannot allocate the requested bandwidth to VC_i on the outgoing link, $A_{max} < min(ER_i, CCR_i)$. The set of bottlenecked connections is B. Intuitively, the bottlenecked connections are those that can use a higher rate allocation at the switch.

2. **Satisfied:** the switch can satisfy the request, $A_{max} \geq min(ER_i, CCR_i)$. The set of bottlenecked connections is S. Intuitively, the bottlenecked connections are those which cannot use even the current maximum bandwidth allocation A_{max}. In some sense, they are currently "saturated."

Typically, a given VC_i will be in different states (bottlenecked and satisfied) at different switches. Observe that connections can move from one state to another depending upon their demand and the available bandwidth. *Free bandwidth* is defined as the amount of bandwidth available as a result of the satisfied connections not claiming their equal share, B_{eq}. The computation of the maximum bandwidth allocation for a connection is done as follows.

First, the state changes of the connection are detected and variables updated:

- If $A_{max} < min(ER_i, CCR_i)$, VC_i is marked as "bottlenecked." Further, in this case, if VC_i was "satisfied" prior to the update, the free bandwidth, B_f is updated, and the number of bottlenecked connections, N_{bot} is incremented.

  Observe that the VC's allocation A_i is not updated since it is not used in the computation of A_{max} as long as it is bottlenecked.

- If $A_{max} \geq min(ER_i, CCR_i)$, VC_i is marked as "satisfied;" its allocation A_i is set to $min(ER_i, CCR_i)$; the free bandwidth, B_f is updated.

  Further, if VC_i was "bottlenecked" prior to the update, the number of bottlenecked connections, N_{bot}, is decremented.

The next step is the computation of the bandwidth allocation. If a connection, $VC_i \in B$, i.e., is currently *bottlenecked*, its maximum allocation (or fairshare, A_{max}) is calculated as:

$$A_{max} = B_{eq} + \frac{B_f}{N_{bot}}$$

On the other hand, if $VC_i \in S$, i.e., is currently satisfied, it is *treated as bottlenecked* and the maximum allocation (or fairshare, A_{max}) is calculated as:

$$A_{max} = B_{eq} + \frac{B_f + A_i - B_{eq}}{N_{bot} + 1}$$

In the preceding equation, observe that the bandwidth allocation of VC_i over and above the equal share $A_i - B_{eq}$ is also considered as part of the "free bandwidth". The use of $N_{bot} + 1$, in the denominator of the fraction shows that the source is considered a bottlenecked connection in the calculation. The purpose of this step is to ensure

the bandwidth allocations to *satisfied* connections as always less than or equal to the allocations to *bottlenecked* connections. The algorithm thus "claims" back any extra bandwidth previously allocated to the connection.

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

$$ER_i = min(ER_i, A_{max})$$

4.9.2 Discussion

The authors classify their work as a "state-maintaining" algorithm since they maintain state information on a per-connection basis. They observe that "stateless" algorithms which do not maintain per-connection state may allocate rates such that there may be significant discrepancies between the sum of the ER values signaled to ABR connections and available link bandwidth.

While this observation is valid in general, an optimistic over-allocation can help increase network utilization, especially in cases when the ABR demand and capacity is variable. The arguable risk is that of queuing delays.

The contributions of the UCSC scheme are the following:

- O(1) emulation of MIT scheme concept
- Focus on scalability. If the VCs set up are always active, then the scheme has O(1) computational complexity with respect to the number of VCs.
- In the steady state, $min(ER_i, CCR_i$ gives the path bottleneck rate. This is because ER_i gives the downstream bottleneck rate, while CCR_i gives the upstream bottleneck rate. This observation is valid when the ER marking is done in the backward RM cells.

- The scheme requires no parameter settings.
- Performance analysis with fixed and variable ABR capacity.

The drawbacks of the scheme are:

- The scheme does not measure the load (aggregate input rate) at the switch. As a result, if a source is sending at a rate below its CCR, then the bottleneck will be underutilized.
- The scheme also does not observe the queuing delay at the switch. Errors in estimation of ABR capacity result in errors in feedback and eventually result in queues. Hence, there is a possibility of infinite queues if the queuing delay is not considered as a metric. However, such a mechanism may easily be developed on similar lines as the ERICA+ proposal studied later in the dissertation.
- The scheme assumes that the sum of the number of bottlenecked and satisfied connections is equal to the number of connections setup. The scheme does not measure the number of active connections. As a result, if a connection is setup, but remains idle for a while, the allocations to other connections remain low and may result in underutilization.
- The convergence time is slower since the scheme attempts never to over-allocate (conservative). This non-optimistic strategy may result in link underutilization of the sources are not always active, or cannot utilize their ER allocations [2].

4.10 DMRCA scheme

The *Dynamic Max Rate Control Algorithm (DMRCA)* scheme [22] was developed by Chiussi, Xia and Kumar at Lucent Technologies, in an attempt to improve the EPRCA scheme.

4.10.1 Key Techniques

DMRCA uses a rate marking threshold similar in concept to the MACR of EPRCA. However, the DMRCA threshold is a function of the *degree of congestion* at the switch and the *maximum rate of all active connections*. This rate threshold is used to estimate the maximum fairshare of any active connection on the link.

The authors observe that the EPRCA depends upon the *mean cell rate of all connections* which it uses as a rate marking threshold. If this mean is close to the fairshare of available bandwidth on the link, then EPRCA performs well. But, if the approximation does not hold, then EPRCA introduces considerable unfairness. For example, if some connections are bottlenecked in other switches, they may cause underestimation of the fairshare. Another case is when rates oscillate due to transient behaviors and/or interactions with multiple switches, leading to incorrect estimates of the actual rate of the connection.

The authors propose to use the maximum rate of all the active connections instead of the mean rate used by EPRCA. They observe that the maximum rate of all connections quickly rises to be above the desired "fairshare" (the maximum rate allocation for unconstrained connections at this switch). Further, this value can be made to converge to fairshare in the steady state.

However, certain problems need to be solved before this idea can be used effectively. First, *the maximum VC rate oscillates* excessively leading to transient instabilities in the behavior. This problem can be tackled by smoothing the maximum rate, filtering out the short-term variations. Second, in some situations, *the maximum rate does not converge rapidly* to the fairshare, again compromising fair behavior. The authors address this by using a reduction factor which is a function of the *degree of congestion* in the switch.

DMRCA uses two thresholds QT and DQT on the queue length for congestion detection. The switch also monitors the maximum rate MAX of all connections arriving at the switch, as well as the VC number of the corresponding connection, MAX_VC.

The algorithm smoothes excessive oscillations in MAX using exponential averaging to calculate an *adjusted maximum rate*, as:

$A\_MAX = (1 - Alpha) \times A\_MAX + Alpha \times MAX$

The averaging factor, $Alpha$ is typically 1/16. The implementation is as follows:

if($RMCell->CCR \geq Beta \times MAX$) {

$A\_MAX = (1 - Alpha) \times A\_MAX + Alpha \times RMCell->CCR$

}

This implementation avoids the need for measurement of MAX over a measurement interval. MAX increases when some VC other than MAX_VC observes that its rate is larger than MAX. MAX decreases when MAX_VC updates MAX based

on its CCR. Further MAX times out if it is not updated for a while (as in the case of bursty sources where some sources can become idle and start up with old allocations).

When the queue length exceeds the threshold QT, the switch considers itself congested and performs intelligent marking. The threshold used to perform intelligent marking is:

$Marking\ Threshold = A\_MAX \times Fn(QueueLength)$

where *Function(Queue Length)* is a discrete non-increasing function of the queue length.

The work also addresses how to tackle the case of connections with $MCR > 0$. For example, the fairness criterion "MCR plus Equal Share" is implemented by subtracting the MCR of the corresponding connection for the CCR of each RM cell and using the result as the algorithm. MCR is added back in order to set the ER field in RM cells. The fairness criterion "Maximum of MCR or Max-Min Share" is implemented by simply ignoring the forward RM cells whose CCR is equal to their MCR.

4.10.2 Discussion

The contributions of the scheme are:

- An enhanced EPRCA-like approach with better fairness and control of rate oscillations.
- Low implementation complexity. A chip implementation of the algorithm is available (the "Atlanta" chip of Lucent Technologies [110].
- The use of a single advertised rate threshold value for all VCs results in nearly equal allocations to unconstrained VCs, even in the presence of asynchrony.

- Use of exponentially averaged "Maximum VC rate" instead of "Mean VC rate," combined with an aggressive queue thresholding policy improves efficiency over EPRCA. The scheme is optimistic in the sense that even if there is a single unconstrained connection and the switch is not fully loaded, the allocated rates increase leading to high utilization.

The drawbacks of the scheme are as follows:

- The scheme measures neither the aggregate load (demand), the aggregate ABR capacity, nor the number of active sources at any point of time. This leads to inaccurate control when the input load does not equal the sum of the declared rates.

- The scheme depends heavily on the queue thresholding, and parameterized control to achieve efficiency. In other words, it does not explicitly try to match ABR demand with the ABR capacity, but indirectly controls it looking at the queue length. As we shall describe later in this thesis, the queue length alone is not a good metric for detecting congestion, and this approach may lead to oscillations especially when the ABR demand and capacity are both highly variable.

- The scheme uses CI bit setting in cases where ER setting becomes unreliable. This approach may result in unfairness especially when the load is variable.

- Another effect of parametric control is longer transient convergence times.

- The queue thresholding procedure requires a number of parameters to be set. These parameters are sensitive to the round-trip time and feedback delay. In

other words, a different set of parameters are required if round trip times change by an order of magnitude, with the link capacities being constant.

- The performance of the scheme in the presence of variable ABR demand and capacity is unclear. Also, the side effects (if any) of the resetting the MAX variable will become more clearer under such conditions.

- Arithmetic mean (or exponential averaging) is not the correct method for averaging ratios where the denominator is not constant [66]. Further, the running average assumes that the successive values averaged are close to each other. The technique cannot effectively average (or track) sequence of values which are uncorrelated.

4.11 FMMRA Scheme

The "Fast Max-Min Rate Allocation (FMMRA)" scheme [6] was developed by researchers Arulambalam, Chen, Ansari at NJIT and Bell Labs.

4.11.1 Key Techniques

The algorithm combines ideas from the ERICA scheme (described in this dissertation) and the UCSC scheme described in section 4.9. It is based on the measurement of available capacity and the exact calculation of fair rates, while not being sensitive to inaccuracies in CCR values.

It uses the concept of an advertised rate, γ, a rate which is given to unconstrained connections. The advertised rate is updated upon receipt of a BRM cell of a session, using its previous value, the change in the bottleneck bandwidth of the session and the change in the bottleneck status of the session. A connection which cannot use

the advertised rate is marked as a bottlenecked connection and its bandwidth usage is recorded. The ER field in the RM cell is read and marked in both directions to speed up the rate allocation process.

The scheme uses the load factor (similar to ERICA) and ER to compute an exponential running average of the maximum value of ER, ER_max:

$ER_{max} = (1 - \alpha)ER_{max} + \alpha max(ER, \frac{ER_{max}}{LoadFactor})$

This computation is done in the backward direction and is expected to reflect the advertised rate after considering the load. Based on the level of congestion, which is determined as a function of the queue length and the load factor, the ER field in the RM cell (both forward and backward) is updated according to:

$ER = min(ER, max(\gamma, (1 - \beta)ER_{max}))$

where β is a single bit value indicating that the connection is bottlenecked elsewhere.

The work also mentions approaches to update the ER field in order to control the queue growth. Specifically, if the queue length reaches a low threshold QT, and $LoadFactor > 1$, only the advertised rate is used in marking the ER field, i.e.,

$ER = min(ER, \gamma)$

The algorithm also has a mode for "severe congestion" $(Q > DQT)$ where ER_{max} is set to the advertised rate. This implies that even if some connections are idle, the non-idle connections are not given any extra bandwidth, allowing queues to drain.

4.11.2 Discussion

The contributions of the scheme are:

- A combination of several ideas in a scheme which achieves the essential goals of fast convergence to fair shares and control of queues.

- An O(1) approximation of the MIT scheme idea, combined with the tracking of load through the exponential averaging of the ER_{max} variable.

Some of the drawbacks of the scheme are:

- The calculation of feedback at the receipt of both the forward and backward RM cells increases the computation burden on the switch.

- The setting of ER in both directions may inhibit rate increase for one round trip time (when the backward direction feedback using the latest information cannot increase the rate because the forward direction had commanded a rate decrease).

- The use of exponential averaging of rates is not entirely correct because a) the rates are ratios and averaging of ratios should be done carefully [66], b) the successive values of rates used in the averaging may not be correlated. In general, Exponential averaging does not produce good results if the values averaged do not exhibit correlation.

4.12 HKUST Scheme

4.12.1 Key Techniques

This scheme was developed by researchers Tsang and Wong at the Hong Kong University of Science and Technology (HKUST). The scheme is a modification of the MIT scheme, which retains the O(N) computational complexity and marks the

ER in both the forward and backward directions. It assumes that the destination resets the ER field to the peak cell rate (PCR), which is not mandated by the traffic management standard. Since it derives from the properties of the MIT scheme, it is fair. The setting of feedback in both the forward and backward directions improves the response of the scheme compared to the MIT scheme. Another interesting aspect is that due to the bidirectional ER setting, and the resetting at the destination, the minimum of ER fields in the forward and backward directions gives the current bottleneck rate for that VC.

4.12.2 Discussion

Though given the above interesting aspects, the scheme has several drawbacks:

- It retains the O(N) complexity of the MIT scheme. Further, doing the ER calculation at the receipt of both the forward and backward RM cells increases the computation burden on the switch.
- The scheme does no load measurement, and as a result may not work if the sources are bottlenecked at rates below their allocations.
- The scheme does not measure the number of active VCs, and uses the (static) total number of VCs for the computation.
- The scheme is incompatible with the ATM Forum's Traffic Management 4.0 specification since it requires the ER to be reset by the destination.
- It is not clear how the scheme accounts for variable capacity, especially the handling of queues which build up during transient phases.

4.13 SP-EPRCA scheme

The SP-EPRCA scheme [16] was developed by Cavendish, Mascolo and Gerla at the University of California at Los Angeles.

4.13.1 Key Techniques

The key idea in SP-EPRCA is the use of a *proportional controller* with a *Smith Predictor (SP)* to compensate for the delay in the ABR feedback loop. Effectively, the dynamic control system with a delay in the feedback loop is converted into a simple first order dynamic system with a delay in cascade. Since, theoretically the delay is brought out of the feedback loop, it does not affect stability and the system should not have oscillations in the steady state.

The scheme aims to keep the queue occupancy under some desired value while achieving a fair distribution of rates. In the steady state, the scheme aims for the following relation between the rate stationary rate u_s, and the stationary queue length x_s of a VC:

$$u_s = \frac{X^o - x_s}{1/K + RTD}$$

K is the gain factor, a parameter of the Smith Predictor, X^o is the target queue length, and RTD is the round trip delay.

The scheme functions as follows. The switches send the available buffer space for cell storage for that particular connection back to the source (in one version of the scheme, the target queue length, X^o, can be fed back instead of using individual buffer allocations). Each source implements a Smith Predictor which requires the knowledge of the round trip delay and an estimate of the varying delay in the network.

The gain factor (K), a parameter of the smith predictor determines the rate of convergence to a steady state. There is also a tradeoff between the buffer space needed, maximum achievable throughput, and the maximum RTD estimation error supported. The queue implementation (FIFO or per-VC queuing) also has a significant impact on convergence. In the default case, the scheme requires a separate Smith Predictor for each VC. The conversion to the single predictor, and the implementation of the FIFO service at the switches requires additional complexity at the switches.

The challenge faced by the scheme designers was to estimate the network delays accurately. Errors in delay would cause the system to be of a higher order. Due to these constraints, the default implementation of the scheme requires per-VC queuing at the switches, and the rate computation to be done at the source end system. Another reason for this was that the round trip times of VCs (required for the smith predictors) can be estimated better at the sources rather than at all switches. Since the ATM Forum standard [35] does not specify rate computation at the source end system or provide hooks for measuring the round trip time at the source end system, the scheme is incompatible with the standards. Note also that the ATM Forum standard expects the switch to compute rates and feedback the rates and not the queue length.

One contribution of the scheme is in its mechanisms for estimating the round trip delays. The scheme uses two mechanisms for dealing with delays, acting in different time scales: **a)** a long time scale delay, keeping track of the variation of the round trip delay due to queuing at intermediate switches and **b)** a short time scale delay, which is called "virtual feedback." The latter mechanisms measures the variability of the

RTD and shuts off the source (for stability) until the RTD come back to reasonable levels.

4.13.2 Discussion

The contributions of the scheme are:

- Use of a control-theoretic approach to the ATM congestion control problem.
- Use of a Smith Predictor to remove the effect of delay from the control loop leading to a simple controller design.
- Techniques for estimating round trip delays and maintaining scheme stability.
- Proof of steady state and stability analysis of the controlled system
- Queues can be controlled to provide zero-loss.

The drawbacks of the scheme are as follows:

- The scheme is incompatible with the current ATM Forum standards, and cannot inter-operate with other schemes implemented in different switches.
- The default version requires the implementation of per-VC queuing at the switches and a separate smith predictor at every source - involving high implementation complexity.
- The transient performance of the scheme is dependent on the accuracy of RTD estimation and the gain factor, K. The latter parameter needs to be reduced to compensate for oscillatory behavior, which in turn affects the convergence time.
- The performance of the scheme in the presence of variable ABR demand and capacity is unclear.

4.14 Summary of Switch Congestion Control Schemes

We have observed in our preceding survey that different schemes have addressed different subsets of switch scheme goals listed in chapter 3. In this section, we summarize these goals and approaches and identify areas not addressed by these proposals.

If we sort the schemes by time, we find that early schemes addressed the basic problem of achieving max-min fairness with minimal complexity. We can see a transition from using purely bit-based ideas for ER-feedback to using purely ER-based ideas for the same purpose.

Early schemes used a number of concepts which have been based on the legacy of bit-based feedback design, which may not be best when explicit-rate feedback capability is available. For example, control of queuing delay is done typically through an threshold-based or hysteresis-based approach which is a legacy from bit-based feedback design. This approach does not work when there is high variance in queue fluctuations due to traffic variation. As discussed in a later chapter, using queue thresholds alone to detect congestion is a flawed technique especially when rate-based control is used.

Later schemes addressed the speed of convergence and the implementation complexity of the scheme, and faced a tradeoff between the two. The issue of measurement raised in this dissertation has been recognized in several contemporary schemes. Some of the schemes described in this section have been developed at the same time, or after the development of the OSU, ERICA and ERICA+ schemes. As a result, they share several features with the schemes we have described in this dissertation.

4.14.1 Common Drawbacks

Though the evolution of switch schemes has yielded increasingly efficient and fair algorithms, with reduction in implementation complexity, and a broader scope, many of these proposals suffer from a common set of drawbacks as listed in this section.

In general schemes, with a few notable exceptions, ***have not been comprehensive in design***, i.e., they either do not address all the goals of a switch scheme and/or make too many assumptions about measurement related aspects of the scheme.

For example, many schemes ***do not address the issue of how to measure the ABR demand***. The lack of information about the demand may lead to under-allocation of rates. Several schemes (like those which use the concept of Mean ACR (MACR)) approximate the average demand per connection. However, if the total demand (aggregate input rate) is not measured, the scheme could be consistently making estimation errors.

Other schemes ***do not monitor the activity of sources***, and may overlook a source becoming temporarily idle. If the idle source is considered while determining allocations for all other sources, the allocations for the other sources may be reduced.

In brief, measurement is necessary to track the current network state used by the scheme. Ideally, a scheme should measure every component of the network state it uses for its calculations.

Another issue is ***how to measure the scheme metrics when there is high variation in the traffic demand and available capacity***. Several metrics need to be observed over intervals of time and averaged over many such intervals to smooth out the effects of such variation. The length of the interval is a key factor in a tradeoff between quick response and accurate response. Implementation issues include

specifying where and when exactly the measurements should be made, and feedback should be given. Several schemes (with the notable exceptions of the Phantom and, to some extent, the DMRCA scheme) do not attempt to address these concerns.

Several switch algorithms ***require the source to restrict its rise by limiting the Rate Increase Factor (RIF) parameter*** to avoid oscillations. But, this affects the transient performance of the scheme. Other switch algorithms ***require setting multiple parameters, and may sometimes be sensitive to parameters***.

Another issue with respect to parameters is ***control-feedback correlation***. Switch algorithms use several control parameters (available capacity, source's rate, the aggregate input rate, the number of active sources etc) to calculate the feedback quantities. Typically, control parameters values are measured asynchronously with respect to when feedback is given. One important responsibility of the switch algorithm is to ensure that the feedback is correlated with the control. Lack of such correlation will lead to perpetual oscillations at best, and queue divergence and collapse at worst. Most schemes do not specify in detail how the correlation is maintained (especially when there is high variation in the network traffic).

Many schemes change from one policy to another for small changes in system state. This introduces ***discontinuities*** in the feedback rate calculation function. If the system state is oscillating around the places where discontinuity is introduced, the scheme would exhibit undesirable oscillations. However, the presence of discontinuities in the feedback function alone does not mean that the scheme is bad. If the number of discontinuities are many (like the use of several queuing thresholds) the scope for undesirable oscillations increases.

The ATM Traffic Management standard also mention ***the Use-it or Lose-it problem*** where sources may retain allocations and use it later when the allocations are invalid. The standard provides minimal support from the source end systems. The switch needs to be able to tolerate transient queuing, and recover quickly from such uncontrollable circumstances.

In this dissertation, we address all these issues and present the design, performance analysis of the switch scheme, and several other aspects of ABR traffic management.

BIBLIOGRAPHY

[1] Santosh P. Abraham and Anurag Kumar. Max-Min Fair Rate Control of ABR Connections with Nonzero MCRs. *IISc Technical Report*, 1997.

[2] Yehuda Afek, Yishay Mansour, and Zvi Ostfeld. Convergence Complexity of Optimistic Rate Based Flow Control Algorithms. In *28th Annual Symposium on Theory of Computing (STOC)*, pages 89–98, 1996.

[3] Yehuda Afek, Yishay Mansour, and Zvi Ostfeld. Phantom: A Simple and Effective Flow Control Scheme. In *Proceedings of the ACM SIGCOMM*, pages 169–182, August 1996.

[4] Anthony Alles. ATM Internetworking. White paper, Cisco Systems, http://www.cisco.com, May 1995.

[5] G.J. Armitage and K.M. Adams. ATM Adaptation Layer Packet Reassembly during Cell Loss. *IEEE Network Magazine*, September 1993.

[6] Ambalavanar Arulambalam, Xiaoqiang Chen, and Nirwan Ansari. Allocating Fair Rates for Available Bit Rate Service in ATM Networks. *IEEE Communications Magazine*, 34(11):92–100, November 1996.

[7] A.W.Barnhart. Changes Required to the Specification of Source Behavior. ATM Forum 95-0193, February 1995.

[8] A.W.Barnhart. Evaluation and Proposed Solutions for Source Behavior # 5. ATM Forum 95-1614, December 1995.

[9] A. W. Barnhart. Use of the Extended PRCA with Various Switch Mechanisms. ATM Forum 94-0898, 1994.

[10] A. W. Barnhart. Example Switch Algorithm for TM Spec. ATM Forum 95-0195, February 1995.

[11] J. Bennett, K. Fendick, K.K. Ramakrishnan, and F. Bonomi. RPC Behavior as it Relates to Source Behavior 5. ATM Forum 95-0568R1, May 1995.

[12] J. Bennett and G. Tom Des Jardins. Comments on the July PRCA Rate Control Baseline. ATM Forum 94-0682, July 1994.

[13] J. Beran, R. Sherman, M. Taqqu, and W. Willinger. Long-Range Dependence in Variable-Bit-Rate Video Traffic. *IEEE Transactions on Communications*, 43(2/3/4), February/March/April 1995.

[14] U. Black. *ATM: Foundation for Broadband Networks*. Prentice Hall, New York, 1995.

[15] P. E. Boyer and D. P. Tranchier. A reservation principle with applications to the atm traffic control. *Computer Networks and ISDN Systems*, 1992.

[16] D. Cavendish, S. Mascolo, and M. Gerla. SP-EPRCA: an ATM Rate Based Congestion Control Scheme basedon a Smith Predictor. Technical report, UCLA, 1997.

[17] Y. Chang, N. Golmie, L. Benmohamed, and D. Siu. Simulation study of the new rate-based eprca traffic management mechanism. *ATM Forum 94-0809*, 1994.

[18] A. Charny, G. Leeb, and M. Clarke. Some Observations on Source Behavior 5 of the Traffic Management Specification. ATM Forum 95-0976R1, August 1995.

[19] Anna Charny. An Algorithm for Rate Allocation in a Cell-Switching Network with Feedback. Master's thesis, Massachusetts Institute of Technology, May 1994.

[20] Anna Charny, David D. Clark, and Raj Jain. Congestion control with explicit rate indication. In *Proceedings of the IEEE International Communications Conference (ICC)*, June 1995.

[21] D. Chiu and R. Jain. Analysis of the Increase/Decrease Algorithms for Congestion Avoidance in Computer Networks. *Journal of Computer Networks and ISDN Systems*, 1989.

[22] Fabio M. Chiussi, Ye Xia, and Vijay P. Kumar. Dynamic max rate control algorithm for available bit rate service in atm networks. In *Proceedings of the IEEE GLOBECOM*, volume 3, pages 2108–2117, November 1996.

[23] D.P.Heyman and T.V. Lakshman. What are the implications of Long-Range Dependence for VBR-Video Traffic Engineering ? *ACM/IEEE Transactions on Networking*, 4(3):101–113, June 1996.

[24] Harry J.R. Dutton and Peter Lenhard. *Asynchronous Transfer Mode (ATM) Technical Overview*. Prentice Hall, New York, 2nd edition, 1995.

[25] H. Eriksson. MBONE: the multicast backbone. *Communications of the ACM*, 37(8):54–60, August 1994.

[26] J. Scott et al. Link by Link, Per VC Credit Based Flow Control. ATM Forum 94-0168, 1994.

[27] L. Roberts et al. New pseudocode for explicit rate plus efci support. *ATM Forum 94-0974*, 1994.

[28] M. Hluchyj et al. Closed-loop rate-based traffic management. *ATM Forum 94-0438R2*, 1994.

[29] M. Hluchyj et al. Closed-Loop Rate-Based Traffic Management. ATM Forum 94-0211R3, April 1994.

[30] S. Fahmy, R. Jain, S. Kalyanaraman, R. Goyal, and F. Lu. On source rules for abr service on atm networks with satellite links. In *Proceedings of First International Workshop on Satellite-based Information Services (WOSBIS)*, November 1996.

[31] Chien Fang and Arthur Lin. A Simulation Study of ABR Robustness with Binary-Mode Switches: Part II. ATM Forum 95-1328R1, October 1995.

[32] Chien Fang and Arthur Lin. On TCP Performance of UBR with EPD and UBR-EPD with a Fair Buffer Allocation Scheme. ATM Forum 95-1645, December 1995.

[33] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, and T. Berners-Lee. Hypertext Transfer Protocol – HTTP/1.1. Request For Comments, RFC 2068, January 1997.

[34] ATM Forum. http://www.atmforum.com.

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

[36] M. Garrett and W. Willinger. Analysis, modeling, and generation of self-similar vbr video traffic. In *Proceedings of the ACM SIGCOMM*, August 1994.

[37] Matthew S. Goldman. Variable Bit Rate MPEG-2 over ATM: Definitions and Recommendations. ATM Forum 96-1433, October 1996.

[38] Rohit Goyal, Raj Jain, Shiv Kalyanaraman, Sonia Fahmy, and Seong-Cheol Kim. Performance of TCP over UBR+. ATM Forum 96-1269, October 1996.

[39] Rohit Goyal, Raj Jain, Shiv Kalyanaraman, Sonia Fahmy, Bobby Vandalore, Xiangrong Cai, and Seong-Cheol Kim. Selective Acknowledgements and UBR+ Drop Policies to Improve TCP/UBR Performance over Terrestrial and Satellite Networks. ATM Forum 97-0423, April 1997.

[40] Rohit Goyal, Raj Jain, Shiv Kalyanaraman, Sonia Fahmy, Bobby Vandalore, Xiangrong Cai, and Seong-Cheol Kim. Selective Acknowledgements and UBR+ Drop Policies to Improve TCP/UBR Performance over Terrestrial and Satellite Networks. ATM Forum 97-0423, April 1997.

[41] M. Grossglauser, S.Keshav, and D.Tse. RCBR: a simple and efficient service for multiple time-scale traffic. In *Proceedings of the ACM SIGCOMM*, August 1995.

[42] S. Hrastar, H. Uzunalioglu, and W. Yen. Synchronization and de-jitter of mpeg-2 transport streams encapsulated in aal5/atm. In *Proceedings of the IEEE International Communications Conference (ICC)*, volume 3, pages 1411–1415, June 1996.

[43] D. Hughes and P. Daley. More abr simulation results. *ATM Forum 94-0777*, 1994.

[44] D. Hunt, Shirish Sathaye, and K. Brinkerhoff. The realities of flow control for abr service. *ATM Forum 94-0871*, 1994.

[45] Van Jacobson. Congestion avoidance and control. In *Proceedings of the ACM SIGCOMM*, pages 314–329, August 1988.

[46] J. Jaffe. Bottleneck Flow Control. *IEEE Transactions on Communications*, COM-29(7):954–962, 1980.

[47] R. Jain. A delay-based approach for congestion avoidance in interconnected heterogeneous computer networks. *Computer Communications Review*, 19.

[48] R. Jain. A timeout-based congestion control scheme for window flow-controlled networks. *IEEE Journal on Selected Areas in Communications*, 1986.

[49] R. Jain. A comparison of hashing schemes for address lookup in computer networks. *IEEE Transactions on Communications*, 1992.

[50] R. Jain. The eprca+ scheme. *ATM Forum 94-0988*, 1994.

[51] R. Jain. The osu scheme for congestion avoidance using explicit rate indication. *ATM Forum 94-0883*, 1994.

[52] R. Jain. Atm networking: Issues and challenges ahead. *Engineers Conference, InterOp+Network World*, 1995.

[53] R. Jain. Congestion control and traffic management in atm networks: Recent advances and a survey. *Computer Networks and ISDN Systems*, 1995.

[54] R. Jain, D. Chiu, and W. Hawe. A quantitative measure of fairness and discrimination for resource allocation in shared systems. *DEC TR-301*, 1984.

[55] R. Jain, D. Chiu, and W. Hawe. A quantitative measure of fairness and discrimination for resource allocation in shared systems. *DEC TR-301*, 1984.

[56] R. Jain, S. Kalyanaraman, R. Goyal, S. Fahmy, and F. Lu. A Fix for Source End System Rule 5. ATM Forum 95-1660, December 1995.

[57] R. Jain, S. Kalyanaraman, R. Goyal, S. Fahmy, and F. Lu. Erica+: Extensions to the erica switch algorithm. *ATM Forum 95-1145R1*, 1995.

[58] R. Jain, S. Kalyanaraman, and R. Viswanathan. Method and apparatus for congestion management in computer networks using explicit rate indication. *U. S. Patent application filed (S/N 307, 375),*, 1994.

[59] R. Jain, S. Kalyanaraman, and R. Viswanathan. The transient performance: Eprca vs eprca++. *ATM Forum 94-1173*, 1994.

[60] R. Jain, S. Kalyanaraman, R. Viswanathan, and R. Goyal. A sample switch algorithm. *ATM Forum 95-0178R1*, 1995.

[61] R. Jain, K. Ramakrishnan, and D Chiu. Congestion avoidance scheme for computer networks. *U.S. Patent #5377322,*, 1994.

[62] R. Jain and K. K. Ramakrishnan. Congestion avoidance in computer networks with a connectionless network layer: Concepts, goals, and methodology. *Proc. IEEE Computer Networking Symposium*, 1988.

[63] R. Jain, K. K. Ramakrishnan, and D. M. Chiu. Congestion Avoidance in Computer Networks with a Connectionless Network Layer. Technical Report DEC-TR-506, Digital Equipment Corporation, August 1987.

[64] R. Jain and S. Routhier. Packet Trains - Measurement and a new model for computer network trafic. *IEEE Journal of Selected Areas in Communications,*, 1986.

[65] Raj Jain. Congestion Control in Computer Networks: Issues and Trends. *IEEE Network Magazine*, pages 24–30, May 1990.

[66] Raj Jain. *The Art of Computer Systems Performance Analysis*. John Wiley & Sons, 1991.

[67] Raj Jain. Myths about Congestion Management in High-speed Networks. *Internetworking: Research and Experience*, 3:101–113, 1992.

[68] Raj Jain. ABR Service on ATM Networks: What is it? Network World, 1995.

[69] Raj Jain. Congestion Control and Traffic Management in ATM Networks: Recent advances and a survey. *Computer Networks and ISDN Systems Journal*, October 1996.

[70] Raj Jain, Sonia Fahmy, Shivkumar Kalyanaraman, Rohit Goyal, and Fang Lu. More Straw-Vote Comments: TBE vs Queue sizes. ATM Forum 95-1661, December 1995.

[71] Raj Jain, Shiv Kalyanaraman, Rohit GOyal, and Sonia Fahmy. Source Behavior for ATM ABR Traffic Management: An Explanation. *IEEE Communications Magazine*, 34(11), November 1996.

[72] Raj Jain, Shivkumar Kalyanaraman, Sonia Fahmy, and Fang Lu. Bursty ABR Sources. ATM Forum 95-1345, October 1995.

[73] Raj Jain, Shivkumar Kalyanaraman, Sonia Fahmy, and Fang Lu. Out-of-Rate RM Cell Issues and Effect of Trm, TOF, and TCR. ATM Forum 95-973R1, August 1995.

[74] Raj Jain, Shivkumar Kalyanaraman, Sonia Fahmy, and Fang Lu. Straw-Vote comments on TM 4.0 R8. ATM Forum 95-1343, October 1995.

[75] Raj Jain, Shivkumar Kalyanaraman, Rohit Goyal, Ram Viswanathan, and Sonia Fahmy. Erica: Explicit rate indication for congestion avoidance in atm networks. U.S. Patent Application (S/N 08/683,871), July 1996.

[76] Raj Jain, Shivkumar Kalyanaraman, and Ram Viswanathan. The osu scheme for congestion avoidance in atm networks: Lessons learnt and extensions. *Performance Evaluation Journal*, October 1997. to appear.

[77] Raj Jain and Shivkumar Kalyanaraman Ram Viswanathan. 'method and apparatus for congestion management in computer networks using explicit rate indication. U. S. Patent application (S/N 307,375), SepJuly 1994.

[78] H. Tzeng K. Siu. Intelligent congestion control for abr service in atm networks. *Computer Communication Review*, 24(5):81–106, October 1995.

[79] Lampros Kalampoukas, Anujan Varma, and K.K. Ramakrishnan. An efficient rate allocation algorithm for atm networks providing max-min fairness. In *6th IFIP International Conference on High Performance Networking (HPN)*, September 1995.

[80] Shivkumar Kalyanaraman, Raj Jain, Sonia Fahmy, Rohit Goyal, and Jianping Jiang. Performance of TCP over ABR on ATM backbone and with various VBR traffic patterns. In *Proceedings of the IEEE International Communications Conference (ICC)*, June 1997.

[81] Shivkumar Kalyanaraman, Raj Jain, Rohit Goyal, and Sonia Fahmy. A Survey of the Use-It-Or-Lose-It Policies for the ABR Service in ATM Networks. Technical Report OSU-CISRC-1/97-TR02, Dept of CIS, The Ohio State University, 1997.

[82] D. Kataria. Comments on rate-based proposal. *ATM Forum 94-0384*, 1994.

[83] J.B. Kenney. Problems and Suggested Solutions in Core Behavior. ATM Forum 95-0564R1, May 1995.

[84] Bo-Kyoung Kim, Byung G. Kim, and Ilyoung Chong. Dynamic Averaging Interval Algorithm for ERICA ABR Control Scheme. ATM Forum 96-0062, February 1996.

[85] H. T. Kung. Adaptive Credit Allocation for Flow-Controlled VCs. ATM Forum 94-0282, 1994.

[86] H. T. Kung. Flow Controlled Virtual Connections Proposal for ATM Traffic Management. ATM Forum 94-0632R2, September 1994.

[87] T.V. Lakshman, P.P. Mishra, and K.K. Ramakrishnan. Transporting compressed video over atm networks with explicit rate feedback control. In *Proceedings of the IEEE INFOCOM*, April 1997.

[88] L.G.Roberts. Operation of Source Behavior # 5. ATM Forum 95-1641, December 1995.

[89] Hongqing Li, Kai-Yeung Siu, Hong-Ti Tzeng, Chinatsu Ikeda, and Hiroshi Suzuki. Tcp over abr and ubr services in atm. In *Proceedings of IPCCC'96*, March 1996.

[90] S. Liu, M. Procanik, T. Chen, V.K. Samalam, and J. Ormond. An analysis of source rule # 5. ATM Forum 95-1545, December 1995.

[91] B. Lyles and A. Lin. Definition and preliminary simulation f a rate-based congestion control mechanism with explicit feedback of bottleneck rates. *ATM Forum 94-0708*, 1994.

[92] P. Newman. Traffic Management for ATM Local Area Networks. *IEEE Communications Magazine*, 1994.

[93] P. Newman and G. Marshall. Becn congestion control. *ATM Forum 94-789R1*, 1993.

[94] P. Newman and G. Marshall. Update on becn congestion control. *ATM Forum 94-855R1*, 1993.

[95] Craig Partridge. *Gigabit Networking*. Addison-Wesley, Reading, MA, 1993.

[96] Vern Paxson. Fast Approximation of Self-Similar Network Traffic. Technical Report LBL-36750, Lawrence Berkeley Labs, April 1995.

[97] K. Ramakrishnan and R. Jain. A binary feedback scheme for congestion avoidance in computer networks with connectionless network layer. *ACM Transactions on Computers*, 1990.

[98] K. K. Ramakrishnan, D. M. Chiu, and R. Jain. Congestion Avoidance in Computer Networks with a Connectionless Network Layer. Part IV: A Selective Binary Feedback Scheme for General Topologies. Technical report, Digital Equipment Corporation, 1987.

[99] K. K. Ramakrishnan and P. Newman. Credit where credit is due. *ATM Forum 94-0916*, 1994.

[100] K. K. Ramakrishnan and "Issues with Backward Explicit Congestion Notification based Congestion Control. Issues with backward explicit congestion notification based congestion control. *ATM Forum 94-0231*, 1993.

[101] K. K. Ramakrishnan and J. Zavgren. Preliminary simulation results of hop-by-hop/vc flow control and early packet discard. *ATM Forum 94-0231*, 1994.

[102] K.K. Ramakrishnan, P. P. Mishra, and K. W. Fendick. Examination of Alternative Mechanisms for Use-it-or-Lose-it. ATM Forum 95-1599, December 1995.

[103] L. Roberts. The benefits of rate-based flow control for abr service. *ATM Forum 94-0796*, 1994.

[104] L. Roberts. Enhanced prca (proportional rate-control algorithm). *ATM Forum 94-0735R1*, 1994.

[105] L. Roberts. Rate-based algorithm for point to multipoint abr service. *ATM Forum 94-0772R1*, 1994.

[106] Larry Roberts. Enhanced PRCA (Proportional Rate-Control Algorithm). ATM Forum 94-0735R1, August 1994.

[107] A. Romanov. A performance enhancement for packetized abr and vbr+ data. *ATM Forum 94-0295*, 1994.

[108] Allyn Romanov and Sally Floyd. Dynamics of TCP Traffic over ATM Networks. *IEEE Journal on Selected Areas in Communications*, May 1995.

[109] W. Stallings. Isdn and broadband isdn with frame relay and atm. *ATM Forum 94-0888*, 1995.

[110] Lucent Technologies. Atlanta chip set, microelectronics group news announcement, http://www.lucent.com/micro/news/032497.html.

[111] Christos Tryfonas. MPEG-2 Transport over ATM Networks. Master's thesis, University of California at Santa Cruz, September 1996.

[112] H. Tzeng and K. Siu. A class of proportional rate control schemes and simulation results. *ATM Forum 94-0888*, 1994.

[113] H. Tzeng and K. Siu. Enhanced credit-based congestion notification (eccn) flow control for atm networks. *ATM Forum 94-0450*, 1994.

[114] International Telecommunications Union. http://www.itu.ch.

[115] Bobby Vandalore, Shiv Kalyanaraman, Raj Jain, Rohit Goyal, Sonia Fahmy, Xiangrong Cai, and Seong-Cheol Kim. Performance of Bursty World Wide Web (WWW) Sources over ABR. ATM Forum 97-0425, April 1997.

[116] Bobby Vandalore, Shiv Kalyanaraman, Raj Jain, Rohit Goyal, Sonia Fahmy, and Pradeep Samudra. Worst case TCP behavior over ABR and buffer requirements. ATM Forum 97-0617, July 1997.

[117] L. Wojnaroski. Baseline text for traffic management sub-working group. *ATM Forum 94-0394R4*, 1994.

[118] Gary R. Wright and W. Richard Stevens. *TCP/IP Illustrated, Volume 2.* Addison-Wesley, Reading, MA, 1995.

[119] Lixia Zhang, Scott Shenker, and D.D.Clark. Observations on the dynamics of a congestion control algorithm: The effects of two-way traffic. In *Proceedings of the ACM SIGCOMM*, August 1991.