ELSEVIER

Performance Evaluation 31 (1997) 51–65

PERFORMANCE EVALUATION
An International Journal

ATM rate-based congestion control using a Smith Predictor

Saverio Mascolo [a,*], Dirceu Cavendish [b], Mario Gerla [b]

[a] Dipartimento di Elettrotecnica ed Elettronica, Politecnico di Bari, 70125 Bari, Italy

[b] Department of Computer Science, University of California, Los Angeles, CA 90024, USA

Abstract

This paper presents a feedback control algorithm for ATM congestion control in which ABR source rates are throttled according to VC queue levels at intermediate nodes along the path. The goal is to "fill in" the residual bandwidth, without exceeding a specified queue threshold. In order to obtain this, we propose a simple and classical proportional controller, plus a Smith Predictor to overcome instabilities due to large propagation delays. As a result, each queue behaves as a simple first-order dynamic system with a delay in cascade. The delay is out of the feedback loop, and therefore does not affect stabilitity. Moreover, since the system dynamic is a first-order one, it is not only stable but it does not even have damped oscillations. We show that this rate-based control scheme can actually be interpreted as a type of an end to end credit scheme. Finally, we propose an effective EPRCA implementation in which each source computes its input rate based on the maximum queue level along the path. Theoretical and experimental results show the fairness of the proposed control scheme, its efficiency under the constraints of the EPRCA implementation, as well as its cell loss free property. © 1997 Elsevier Science B.V.

Keywords: ATM congestion control; End to end flow control; Long propagation delay

1. Introduction

In an ATM network, in order to avoid congestion it is expected that the ABR traffic input rate be regulated so that all entering cells can be completely delivered using the existing network resources (i.e. queues, processing power and link transmission capacity). A classical control approach to deal with this problem consists in monitoring the level of the unused resources and in feeding back the measured levels to a controller which adjusts the ABR traffic input rates so that congestion is avoided.

The input rate control approach is known in literature as rate-based [4] in contraposition to the credit approach [7] which, instead of the rate, aims at regulating the number of incoming cells. Many rate-based algorithms can be found in literature. However, none of these is completely satisfactory either for its complexity or for lack of stability properties, as is well reported in the excellent paper by Benmohamed and Meerkov [3]. In fact, due to transmission and propagation delay, most algorithms exhibit persistent oscillations. Furthermore, they have not been analyzed from the stability point of view, and so cannot guarantee

* Corresponding author. E-mail: mascolo@poliba.it.

PII S0166-5316(97)00008-4

the boundedness of the queues. Considering, for example, the well-known additive increase/multiplicative decrease PRCA [2], it is neither possible to state its stability nor to guarantee cell loss avoidance. To our best knowledge, the paper by Benmohamed and Meerkov [3] is the first attempt to develop an analytic method for the design of congestion controllers which ensure good dynamic performance along with fairness in bandwidth allocation. However, the control law proposed in that paper requires a complex adjustment of control parameters in order to maintain stability and damp oscillations. Moreover, these parameters must be dynamically tuned to the specific input traffic and network condition. Finally, it is difficult to prove global stability, due to the complexity of the control strategy.

This paper presents a simple and effective rate-based congestion control algorithm capable of "filling in" quickly the unused bandwidth with ABR traffic. The main appeal of the proposed congestion control algorithm consists in the use of a simple, first-order dynamic model (for the queue levels) in cascade with a delay. This yields the following properties: (a) the queue occupancy never exceeds maximum queue capacity (i.e. no cell loss); (b) the queue occupancy dynamic is always stable for any positive proportional gain K, thus relaxing the need to dynamically adjust this parameter in order to stabilize queues or damp oscillations.

The paper is structured as follows: In Section 2, the network model, the queue model and the input rate control law are presented along with an analytical study of the resulting dynamics; in Section 3 a discretized version of the algorithm is developed and an interpretation in terms of credit based end to end flow control is given. In Section 4 some further modifications are introduced to take into account the constraints of the ATM Forum specifications [8] and a steady state input rates analysis is outlined. In Section 5 a comparison with the PRCA algorithm is carried on, in Section 6 simulations results are reported, while conclusions are addressed in the last section.

2. The model

In this section first the network and the queue model are presented. Next the input rates control law is proposed along with an analysis of the transient and steady state behaviour of the continuous controlled system.

2.1. Network model

We mainly follow the notation reported in [3]. The network consists of $N = \{1, \ldots, n\}$ nodes and $L = \{1, \ldots, l\}$ links. Each link i is characterized by: transmission capacity $c_i = 1/t_i$ (cells/s); propagation delay td_i; processing capacity $1/tpr_i$ (cell/s), where tpr_i is the time the switch i needs to take a packet from the input and place it on the output queue. We assume that the processing capacity of each node is larger than the total transmission capacity of its incoming links so that congestion is caused by transmission capacity only. The network traffic is contributed by source/destination pairs (S, D), where $S, D \in N$. To each (S, D) connection is associated a virtual circuit (VC) mapped on the path $p(S, D)$. Each source is characterized by its maximum transmission speed, $c_s = 1/t_s$.

Each link maintains a separate queue for each virtual circuit VC passing through it. We indicate with $x_{i,j}(t)$ the occupancy at time t of the queue associated with link i and VC_j, and with $X^\circ_{i,j}$ the corresponding queue threshold level. The control law computes the source input rate $u(t)$ (cell/s). A key parameter for this system is the bandwidth delay product td_j/t_j which represents the number of cells "in flight" on the transmission link.

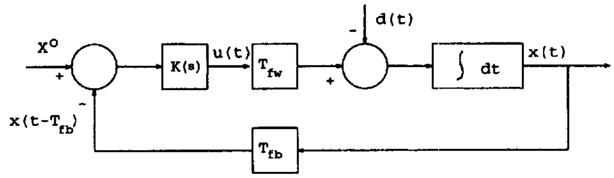


Fig. 1. Queue dynamic model with a proportional controller.

2.2. *Model of the queue behavior*

In this section we present a model of the dynamic behavior of each queue in response to input and output rate changes. We assume a deterministic fluid model approximation of cell flow. Each link maintains a separate queue for each virtual circuit (VC) passing through it. The reason for this choice is to ensure, through a round robin service discipline, the fair sharing of the link by each VC. Considering the queue associated with the virtual circuit VC_j at link i, the level of occupancy $x_{i,j}(t)$ at time t, starting at $t = 0$ with $x_{i,j}(0) = 0$, is the integral over the time $(0, t)$ of the difference between the rate of packets entering the queue (say $u_{i,j}(t)$) and rate of packets leaving the queue (say $d_{i,j}(t)$):

$$x_{i,j}(t) = \int_0^t [u_{i,j}(t' - T_\mathrm{d}) - d_{i,j}(t')]\,\mathrm{d}t', \tag{1}$$

where T_d is the transmission delay from the input source to the i, j queue.

2.3. *The rate control law*

In this section, we present the control algorithm to regulate source rates. We propose a closed-loop control based on feeding back the bottleneck queue occupancy. The feedback mechanism requires that each traffic source emits one control cell (RM cell) every N data cells. Switches are required to write in the RM cells the queue occupancy if this value is greater than the one already stored. In this way, at the destination, each RM cell carries the maximum queue level over all the encountered switches and it comes back to the source. Upon receiving of this value, the source updates the input rate. In order to control the queue level $x(t)$ [1] for a specific VC, we initially use a simple proportional controller. Letting X° be a set point for the queue level, we compute the difference between it and the current queue level $x(t)$. This difference, the error $e(t)$, is amplified by a positive constant gain K, so that $Ke(t)$ is the input rate imposed to the VC source. The proposed control implements the reasonable idea of enforcing an input rate proportional to the room available in the queue. This mechanism tends to "fill the queue", thus keeping link utilization high.

The calculated input rate $Ke(t)$ at time t will have effect on rate adjustments only after the round trip delay along the path, i.e. the time that the computed rate needs to reach the source, change the rate value, and finally returns to the queue as an inflow rate $Ke(t)$. Fig. 1 depicts the block diagram of this system, where T_fw, T_fb are the respective propagation delays involved, and $\mathrm{RTD} = T_\mathrm{fw} + T_\mathrm{fb}$ is the round trip delay. Note that, in wide area networks, the round trip delay is mostly determined by the propagation delay, so we assume that this quantity is fixed and known in advance to the controller at source site. Notice that,

[1] From now on we drop the i, j subscripts, for the sake of simplicity.

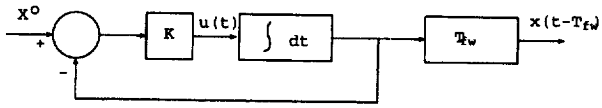


Fig. 2. Equivalent model of the queue dynamic in response to the threshold level.

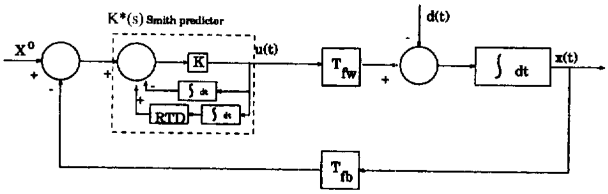


Fig. 3. Queue dynamic model using a proportional controller plus a Smith Predictor.

in Fig. 1, a generic controller $K(s)$ is depicted where s stands for Laplace variable, rather than a simple proportional controller K.

Let us first use a simple proportional controller, $K(s) = K$. Then, due to the large delay, the dynamic behavior of the queue level might exhibit oscillations, and even become unstable. In order to reduce oscillations it is necessary to reduce the amplification gain K, but this carries the drawback of a very long transient, i.e. the input rate is not able to fill in rapidly the queue, making the outgoing link underutilized [6].

To stabilize this system, still preserving the ability of quickly "filling in" the available queue space, we propose a classical Smith Predictor [9]. Following Smith's principle, we substitute the controller $K(s)$ in Fig. 1 with a controller $K^*(s)$ (see Fig. 3) such that the resultant system dynamic is that of a first-order system in cascade with a pure delay (Fig. 2), in the absence of the input $d(t)$.

Thus, equating the transfer functions of the systems in Fig. 3 and in Fig. 2 one can verify that the Smith Predictor controller $K^*(s)$ is given by

$$K^*(s) = \frac{K}{1 + K((1 - \mathrm{e}^{-\mathrm{RTD}\cdot s})/s)},$$

where s is the Laplace transform variable. The Smith Predictor shown in Fig. 3 gives the following input rate control equation:

$$\begin{aligned} u(t) &= K\left[X^{\mathrm{o}} - x(t - T_{\mathrm{fb}}) - \int_0^t u(t')\,\mathrm{d}t' + \int_0^{t-\mathrm{RTD}} u(t')\,\mathrm{d}t'\right] \\ &= K\left[X^{\mathrm{o}} - x(t - T_{\mathrm{fb}}) - \int_{t-\mathrm{RTD}}^{t} u(t')\,\mathrm{d}t'\right]. \end{aligned} \tag{2}$$

Note that this equation implements a simple proportional control action with the difference that the actual queue level is increased by the number of cells transmitted during the last round trip delay. Thus the physical interpretation is that the controller reacts as if all the "in flight" cells were in the bottleneck queue.

To describe the dynamic of the system it is helpful to look at the equivalent system shown in Fig. 2. In this figure we can observe two parts:

(a) The first one, containing the integrator, the constant gain K, and the delay free feedback loop is a first-order system, and thus is stable for any positive value of the parameter K. This parameter affects the transient behavior only. Namely $1/K$ is the system time constant T (meaning that after $4T$ intervals the system reaches stationary condition). Moreover, the dynamic response to a step function does not exhibit oscillations in reaching the stationary state. This implies that the queue occupancy never overshoots the set point level X^{o}, and hence the set point can be set equal to the queue capacity without ever incurring cell loss,

(b) The second part consists of a pure delay block that causes a shift in time of the queue level $x(t)$.

Concluding, the resulting behavior of the queue occupancy, starting at $t = 0$ with an empty queue, is given by the first-order system response to a step function delayed by the round trip RTD, that is: $x(t) = X^{\mathrm{o}}[1 - \exp(-(t - T_{\mathrm{fw}})/T)] * 1(t - T_{\mathrm{fw}})$ where $1(t - T_{\mathrm{fw}})$ is the delayed step function (see Fig. 4). Finally, note that since the calculated input rate cannot be greater than the maximum source transmission speed $1/t_{\mathrm{s}}$, $KX^{\mathrm{o}} = 1/t_{\mathrm{s}}$. Thus the system time constant is $T = X^{\mathrm{o}}t_{\mathrm{s}}$.

The system shown in Fig. 3 has a behavior equivalent to the system depicted in Fig. 2, in response to the input X^{o}. Now we consider the behavior of the queue level $x_d(t)$ in response to the output rate $d(t)$, where $d(t)$ can be modeled as a step function $a * 1(t)$, and a is the fraction of bandwidth, normalized to one, given to each connection.

Using Laplace transform method, after some calculations, we find

$$
\begin{aligned}
x_d(t) = & -a[t * 1(t) - (t - \mathrm{RTD}) * 1(t - T_{\mathrm{fw}})] - \frac{a}{K}[1 - \mathrm{e}^{-K(t - T_{\mathrm{fw}})}] * 1(t - T_{\mathrm{fw}}) \\
& + x(0) * 1(t) - x(0)[1 - \mathrm{e}^{-K(t - T_{\mathrm{fw}})}] * 1(t - T_{\mathrm{fw}}),
\end{aligned}
$$

where $x(0) \geq 0$ is the queue level at $t = 0$. The overall response to $d(t)$ and X^{o}, therefore, is given by

$$x_{\mathrm{tot}}(t) = x_d(t) + x(t) \tag{3}$$

and the steady state ($t \to \infty$) queue level is

$$x_{\mathrm{tot}}(\infty) = X^{\mathrm{o}} - a\mathrm{RTD} - \frac{a}{K}. \tag{4}$$

Fig. 4 shows the transient behavior $x(t)$ in response to X^{o}, the transient behavior in response to $d(t) = 1(t) - 0.5 \times 1(t - \text{offset})$ (where $1(t)$ is the step function), and the overall transient $x_{\mathrm{tot}}(t)$. The initial offset is the time when the bandwidth $d(t)$ drops below the input rate of the corresponding VC queue.

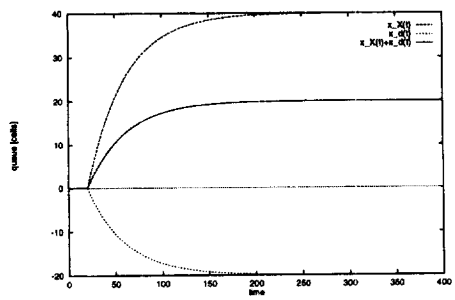


Fig. 4. Queue level transient dynamic.

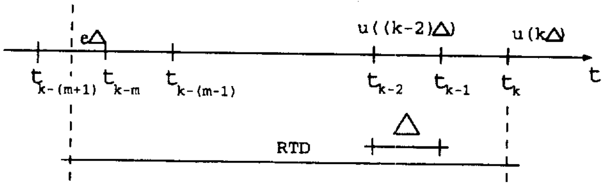


Fig. 5. Discrete time notation.

3. Discrete time rate-based control

So far we have dealt with continuous time models only. However, in ATM networks, feedback information is relayed in cells, and thus not available in continuous time, but rather in sampled form. Fortunately, the discrete time implemention of the Smith Predictor is simpler than the continuous one [1].

From Nyquist sampling theorem and from control theory it is known that, in order to have a "continuous like" performance of the system under digitized control, the ratio of the time constant of the system over the sampling time must fall within the interval (2, 4) [1]. Indicating by Δ the sampling time and recalling that $T = X^{\mathrm{o}} t_{\mathrm{s}}$, it follows:

$$\frac{X^{\mathrm{o}} t_{\mathrm{s}}}{\Delta} = [2, 4]. \tag{5}$$

To write the discrete time version of the control equation (2) we must consider two cases:

(i) $\mathrm{RTD} \geq \Delta$: The ratio $\mathrm{RTD}/\Delta = m + \epsilon$, where m is an integer and $\epsilon \in [0, 1)$. Rewriting the continuous time equation (2) in its discrete version, we obtain the input rate at time $t_k = k\Delta$ [2]:

$$u(k\Delta) = K\left[X^{\mathrm{o}} - x(k\Delta - T_{\mathrm{fb}}) - u(k\Delta - (m+1)\Delta)\epsilon\Delta - \sum_{i=1}^{m} u((k-i)\Delta)\Delta\right]. \tag{6}$$

(ii) $\mathrm{RTD} < \Delta$:

$$u(k\Delta) = K[X^{\mathrm{o}} - x(k\Delta - T_{\mathrm{fb}}) - u((k-1)\Delta)\mathrm{RTD}]. \tag{7}$$

The notation used in the previous equations is illustrated in Fig. 5 and will be followed throughout the paper. It results: $t_k = t_{k-1} + \Delta$.

3.1. End to end credit based interpretation of the control equation

For the sake of simplicity, we consider the case in which the ratio RTD/Δ is an integer ($\epsilon = 0$)[3]. The summation on the right-hand side of Eq. (6) can be rewritten as the sum of two parts:

$$I = u(t - T_{\mathrm{fw}} - \Delta)\Delta + u(t - T_{\mathrm{fw}} - 2\Delta)\Delta + \cdots + u(t - \mathrm{RTD})\Delta,$$
$$II = u(t - \Delta)\Delta + u(t - 2\Delta)\Delta \cdots + u(t - T_{\mathrm{fw}})\Delta.$$

[2] Note that k should not be confused with the gain K.

[3] The extension of this section to the case $\epsilon \neq 0$ is trivial.

The first one represents the number of cells that have already arrived at the bottleneck queue but are not yet known at the source due to the feedback propagation delay T_{fb}. The second one represents the number of cells that are traveling from the source to the queue. Therefore the input rate computation at time t can be rewritten as: $u(t) = K[X^{\mathrm{o}} - x(t - T_{\mathrm{fb}}) - (I) - (II)]$. We can interpret $x(t - T_{\mathrm{fb}}) + (I) + (II)$ as an "effective queue level at time t". So the calculation of the input rate $u(t)$ is made as if all "in flight" cells were already at the queue. In this way the dynamic is delay free, which results in stability and lack of oscillations.

Finally we would like to interpret the difference between the queue capacity and the "effective queue level" as the number of cells that can be transmitted by the source without causing overflow to the bottleneck queue.

4. From periodic to aperiodic feedback

To implement the proposed discrete time control algorithm we need to supply the controllers located at the sources with periodic feedback information (every Δ units of time, with Δ satisfying Eq. (5)). This can be obtained if the upstream node of a congested link sends the feedback information, at every sampling time, to all the sources in the upstream direction, as in the backward congestion notification (BCN) scheme. This is what is assumed in [3]. We call this type of scheme "periodic feedback control". In systems where forward congestion notification (FCN) scheme is used, like in the PRCA scheme, the source is responsible for transmitting a resource management (RM) cell every NRM data cells. The control cell itself has to complete for bottleneck link bandwidth, since it has to reach the destination node before being relayed back to the source through either the same or an alternative reverse path. Clearly, under this scheme, it is not possible to guarantee the periodic feedback information used in the discrete-time control equation. Due to the sharing of the congested link, the rate of the feedback cells that can be received by the sources is $B_{\mathrm{av}}/[N_{\mathrm{vc}} \times (1 + \mathrm{NRM})]$, where N_{vc} is the number of VCs sharing the same bottleneck link and B_{av} is the available bandwidth. Thus the interarrival time of the feedback cells increases with $N_{\mathrm{vc}} \times (1 + \mathrm{NRM})$. In order to guarantee the minimum sampling time prescribed by Eq. (5), we need a control algorithm capable of operating well even if no RM cells are received for a large period of time. If the source receives the feedback information, the control algorithm will adjust the rate accordingly. Otherwise it will compute the rate by estimating the missing feedback information in a conservative way. We call this type of control "virtual feedback control".

4.1. An EPRCA control algorithm

We now propose a version of the previous discrete time control algorithm suitable for the FCN feedback relay scheme. The feedback information is provided by RM cells which collect the maximum buffer level along the path. Note that, regardless of the bottleneck location along the VC path, RTD is always the same. The system will still be cell loss free even if we are not able to guarantee the required periodicity of feedback information.

The basic idea is to update the source rate at least after each Δ sampling interval, regardless whether the source gets the feedback information or not. Let t_k, t_{k+1} be the instants at which the source receives the last and actual feedback information, respectively. Two cases need to be considered:

(i) $t_{k+1} - t_k \le \Delta$. The source stores the rate $u(t_k)$, as well as its duration Δ_k, so that $u(t_k)\Delta_k$ becomes one of the terms of the summation in the control equation. Thus, the rate updating equation is:

$$u(t_k + \Delta_k) = K\left[X^{\mathrm{o}} - x(t_k + \Delta_k - T_{\mathrm{fb}}) - \sum_{i=0}^{m} u(t_{k-i})\Delta_{k-i} - u(t_{k-m-1})\left(\mathrm{RTD} - \sum_{i=0}^{m} \Delta_{k-i}\right)\right],$$

where $\sum_{i=0}^{m} \Delta_{k-i} \le \mathrm{RTD} < \sum_{i=0}^{m+1} \Delta_{k-i}$, $\Delta_{k-i} \le \Delta \; \forall i$, $t_k = t_{k-1} + \Delta_{k-1}$.

(ii) The interval Δ expires before the source receives its control packet. In this case, the algorithm has to estimate the queue level $x(t_k + \Delta - T_{\mathrm{fb}})$. In order to be conservative, and to prevent cell loss, we propose the following "worst case" estimate of the missing queue level. We conservatively assume that in the time interval $[t_k, t_k + \Delta]$ (with $\Delta = \Delta_k$) the queue has zero output rate. Thus the "worst case" queue level is the last value $x(t_k - T_{\mathrm{fb}})$ plus what has been received in the interval $[t_k, t_k + \Delta]$. The accrued term corresponds to the number of cells pumped into the network during the interval $[t_k - \mathrm{RTD}, t_k - \mathrm{RTD} + \Delta]$. Therefore, the "worst case"estimate of the queue level at time $t_k + \Delta_k$ is:

$$x(t_k + \Delta_k - T_{\mathrm{fb}}) = x(t_k - T_{\mathrm{fb}}) + u(t_{k-m-2})\left(\mathrm{RTD} - \sum_{i=1}^{m+1} \Delta_{k-i}\right) + u(t_{k-m-1})\left(\Delta - \left(\mathrm{RTD} - \sum_{i=1}^{m+1} \Delta_{k-i}\right)\right).$$

We call "virtual feedback" this worst case estimation of the queue level. Note that this is equivalent to storing the last received feedback value, $x(t - T_{\mathrm{fb}})$, and adding the new term $u(t_k)\Delta$ to the last sum of "in flight" cells, say sum F, i.e.

$$\mathrm{sum}\ F = \sum_{i=1}^{m+1} u(t_{k-i})\Delta_{k-i} + u(t_{k-m-2})\left(\mathrm{RTD} - \sum_{i=1}^{m+1} \Delta_{k-i}\right)$$

and the rate is

$$u(t_k + \Delta_k) = K[X^{\mathrm{o}} - x(t_k - T_{\mathrm{fb}}) - u(t_k)\Delta_k - \mathrm{sum}\ F].$$

In this proposed EPRCA algorithm, the sources at the edge nodes of the network update their input rates at least every Δ units of time. If they do not get information about the occupancy of the congested queue, they decrease their rates based on a "worst case" estimate of the congested queue level. When they get the next feedback information they will increase their rates because the actual queue level cannot be larger than the conservative estimate. In other words, the algorithm behaves as a "positive feedback", decreasing the rate when feedback is not available and increasing it when feedback information resumes. Note that this is a very important aspect to guarantee stability in any feedback congestion control because, due to congestion, it is not possible to guarantee the rate at which feedback cells are received.

4.2. Steady state analysis and full link utilization

Under stationary conditions equation (2) or (6) reduces to:

$$u_{\mathrm{s}} = K(X^{\mathrm{o}} - x_{\mathrm{s}} - u_{\mathrm{s}} \cdot \mathrm{RTD}), \tag{8}$$

where u_s and x_s stand for the stationary values of $u(t)$ and $x(t)$, respectively. It results

$$x_s = X^o - u_s(T + \mathrm{RTD}), \tag{9}$$

where $T = 1/K$ is the system time constant. To guarantee full link utilization it is sufficient that a link has always data to send, that is $x_s > 0$ which gives

$$X^o > u_s(T + \mathrm{RTD}).$$

In other words full link utilization is guaranteed by using queue capacities of the order of magnitude of bandwidth-delay product.

5. PRCA scheme and the proposed EPRCA: a comparison

In the ATM Forum PRCA proposal [2], an additive increase/multiplicative decrease rate control is exercised at the sources. Binary feedback information (congested/not congested) is received at the sources, and rate increase (additive) is performed in case a "not congested" feedback is received. Failure to receive the "not congested" notification causes multiplicative rate decrease at the source after each time interval Δ, thus making the scheme conservative.

Our proposed EPRCA uses the delayed queue occupancy as the feedback information. Like in the PRCA scheme, if no feedback is received, the source calculates the rate at fixed intervals Δ related to the system time constant. The calculation is performed using a "worst case" estimate of the queue level.

In the following, we study the dynamic behavior of the rate when the source lacks feedback information. Let $u(0) = (1/X^o t_s)(X^o - x(t - T_{\mathrm{fb}}) - \sum_{\mathrm{RTD}} u(t_i)\Delta_i)$ be the rate computed based on the last received feedback cell. If no feedback information is received since then, the rate must be updated every Δ unit of time, using the "worst case" estimate. It follows

$$\begin{aligned}
u(1) &= \frac{1}{X^o t_s}\left[X^o - x(t - T_{\mathrm{fb}}) - u(0)\Delta - \sum_{\mathrm{RTD}} u(t_i)\Delta_i\right] = \frac{u(0)}{X^o t_s}(X^o t_s - \Delta),\\
u(2) &= \frac{u(1)}{X^o t_s}(X^o t_s - \Delta)\\
&\;\vdots\\
u(k) &= \frac{u(k-1)}{X^o t_s}(X^o t_s - \Delta) = u(0)\left[\frac{(X^o t_s - \Delta)}{X^o t_s}\right]^k,
\end{aligned}$$

i.e. the rate decreases exponentially. When the source resumes receiving feedback information, the rate jumps to $K(X^o - x(t - T_{\mathrm{fb}}) - \sum_{\mathrm{RTD}} u(t_i)\Delta_i)$.

Therfore, we note that our EPRCA scheme, developed from a precise and simple mathematical model, operates according to a "positive feedback" mechanism, much like the PRCA scheme. The important difference is that the dynamic behavior of our regulation is related to the network state and parameters. In fact, the rate decreases exponentially with a base related to the sampling time/time constant ratio. More importantly, the increasing jumps are related to the queue level and to the number of cells released from the source during the last round trip interval. As a consequence, our EPRCA scheme does not drop cells. In constrast, the conventional PRCA scheme does not use precise information on the queue level and does

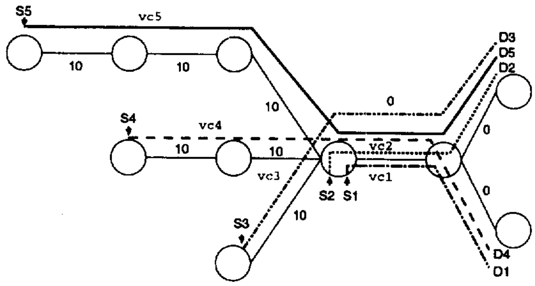


Fig. 6. Network topology.

Table 1
VC connections activity

Connection	1	2	3	4	5
Start time	500	2500	1000	4000	5500
End time	7000	10 000	8500	10 000	10 000

not take into account the number of cells released during the last round trip delay. Consequently, it cannot perform the correct rate increase so as to prevent congestion and cell loss.

6. Simulation results

In this section, we present results of a discrete event simulation of the proposed control scheme. We first show the performance of the periodic control algorithm under the same scenario considered by Benmohamed and Meerkov [3]. Then we compare the proposed EPRCA scheme with the conventional PRCA scheme [2].

The network topology, shown in Fig. 6, is the same presented in [3]. Links have uniform speeds, normalized to 1 cell per unit of time [cell/s]. Links to the right of the bottleneck have a bandwidth-delay product of 10 cells, while the links to the left of the bottleneck have zero propagation delay (similar to [3]).

Five VC connections compete for bandwidth resources of a bottleneck link. VC connection activity (i.e. start and end time) is described in Table 1. We assume infinite backlog at each source. We set a queue level $X^{\mathrm{o}} = 40$ for each queue, in order to have a sampling time of the system of $40/4 = 10$, i.e., one fifth of the interarrival time of the feedback cells under the FCN scheme, with NRM $= 10$ and $N_{\mathrm{vc}} = 5$.

6.1. *Periodic feedback*

We first show the performance of a periodic feedback version of the control scheme, in conditions similar to [3].

According to Eq. (5) we choose a sampling time $\Delta = 10$. Fig. 7(a) shows the behavior of the five input rates, corresponding to connections $S1$–$S5$, at source nodes. For the sake of comparison with PRCA, we assume an initial cell rate of 0.1 [cells/s], equal to the PRCA minimum cell rate. After the start/end

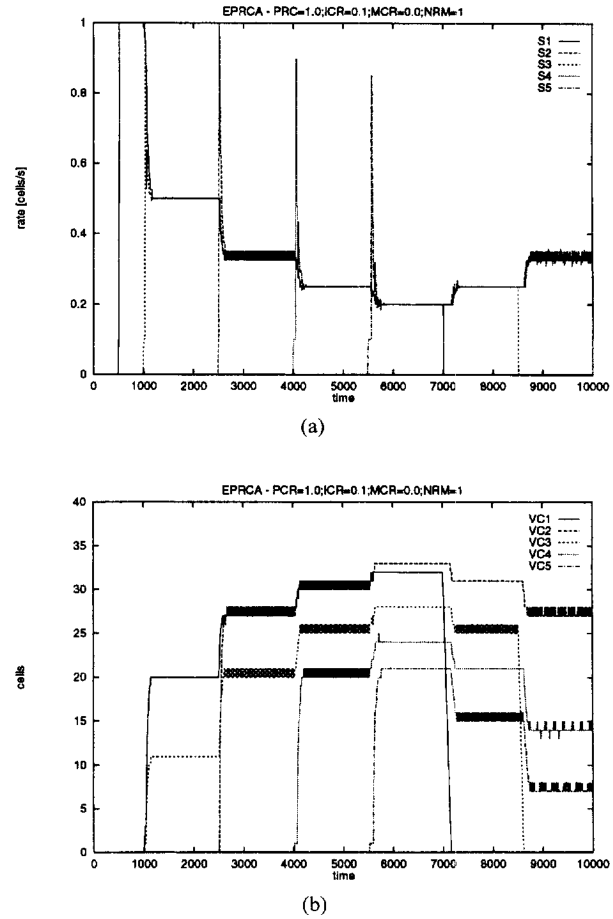


Fig. 7. Periodic feedback: (a) VCs input rates; (b) bottleneck queue levels.

of a connection, each rate rapidly settles to the new fair stationary value [4]. Fig. 7(b) shows the dynamic behavior of the five queues at the bottleneck link, corresponding to VC1–VC5 bottleneck queues. As can be noted, no queue overflow occurs. Moreover, each stationary level is in accordance with Eq. (9). The overall performance is similar to the periodic control of Benmohamed and Meerkov [3], without having dynamic tuning of control parameters.

6.2. Aperiodic feedback

The aperiodic control scheme, with $\Delta = 10$, requires NRM $= 1$ (i.e. one control cell every data cell) in order to guarantee the minimum feedback frequency rate. The value NRM $= 1$ derives from the fact that, under the heaviest traffic condition (five connections), the feedback cell interarrival time is

[4] When there are three active connections, the figure shows small oscillations, due to the fact that the control equation tries to regulate the queue occupancy to a value between two integers.

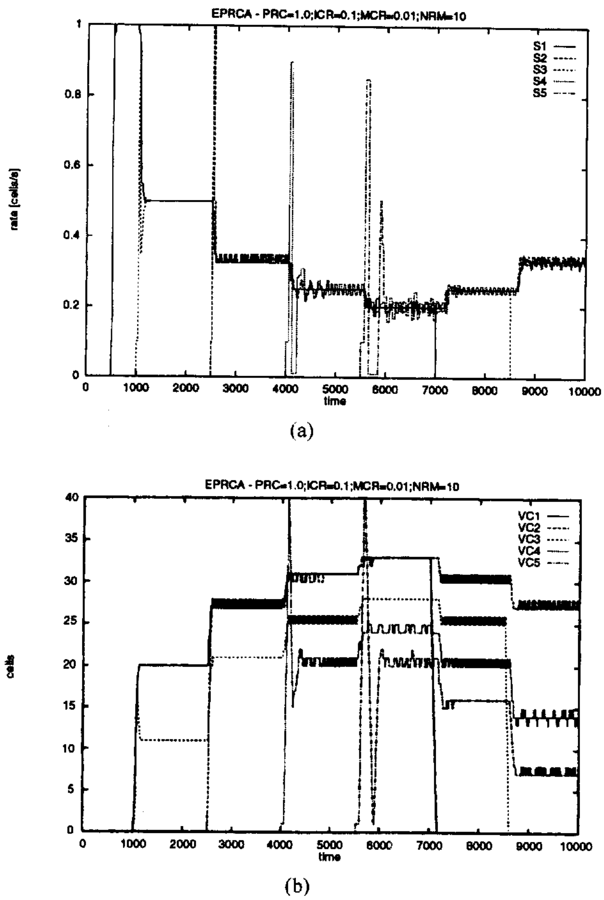


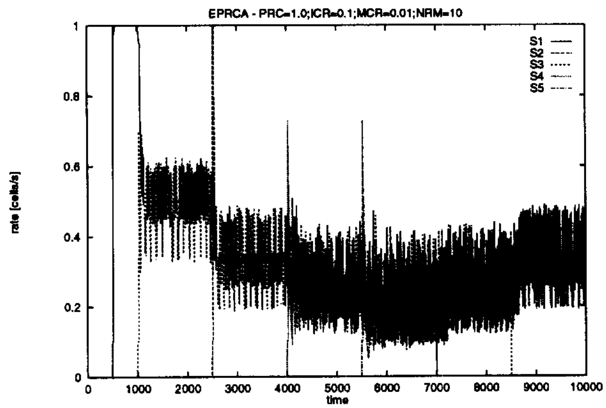
Fig. 8. Aperiodic feedback: (a) VCs input rates; (b) bottleneck queue levels.

$\Delta = N_{\rm vc}(\mathrm{NRM} + 1)$. Since the minimum feedback rate is maintained, simulation results are identical to the ones under periodic control, as expected, and hence are omitted.

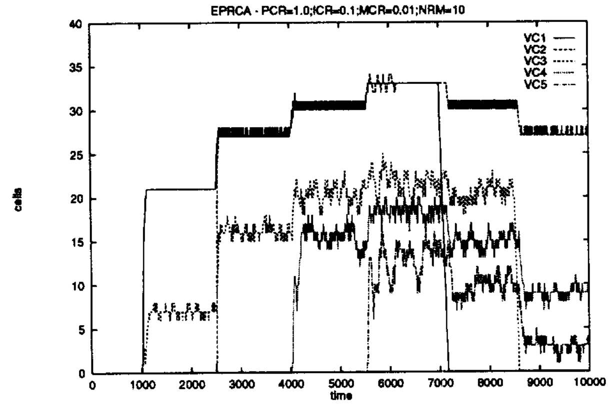
We next show performance degradation in case Eq. (5) is not respected. By setting NRM $= 10$, under the heaviest traffic condition, the feedback interarrival time is $55 > 10$. We see from Fig. 8(a) that rates does not reach rapidly the stationary condition anymore. Moreover, Fig. 8(b) shows that overflow occurs in VC4 and VC5 queues. Other simulation results, not reported here, show that the greater the NRM value, the less controlled the queue levels are.

6.3. EPRCA using virtual feedback

Next, we study the performance of the proposed EPRCA under the same conditions and feedback frequency (NRM $= 10$), as used above. Fig. 9(a) shows the oscillatory behavior of the controlled rates. This is so because the control operates in the "positive feedback" mode, i.e. increasing promptly the rate when a feedback cell is received, and decreasing exponentially otherwise. However, the oscillations are constant in amplitude, and centered at the fair rate value, hence throughput performance is preserved. The



(a)



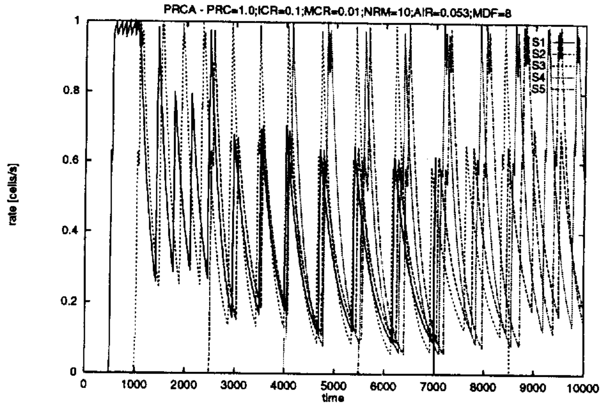
(b)

Fig. 9. Aperiodic + virtual feedback: (a) VCs input rates; (b) bottleneck queue levels.

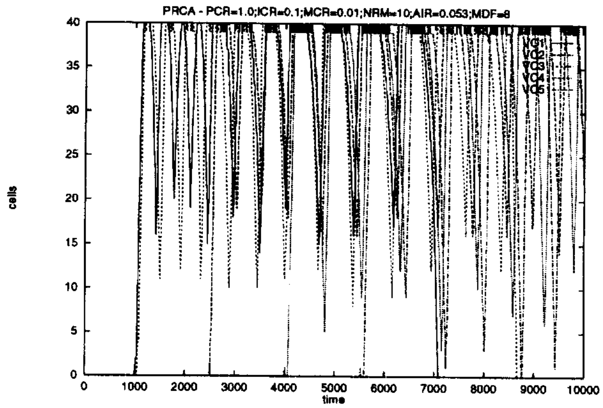
frequency of oscillations is high because the virtual feedback period is $\Delta = 10$, while the actual feedback interarrival time is about 50. In fact, the control algorithm decreases the rate every $\Delta = 10$, in a conservative way, increasing it promptly when a feedback cell is received (approx. every 50 units of time). Fig. 9(b) shows that the queue levels are still bounded, guaranteeing no cell loss. Thus, the major advantage of the virtual feedback is to prevent cell loss, due to congestion, even if it is not possible to guarantee the frequency of feedback cells.

6.4. Conventional PRCA

For the sake of comparison, the PRCA scheme has been simulated under the same traffic conditions as before, with parameters: NRM = 10; AIR = 0.053; MDF = 8. The results are shown in Fig. 10. As expected, the PRCA scheme does not prevent cell loss, because it cannot account for the bottleneck queue level and the number of cells "in flight".



(a)



(b)

Fig. 10. PRCA control algorithm: (a) VCs input rates; (b) bottleneck queue levels.

7. Conclusion

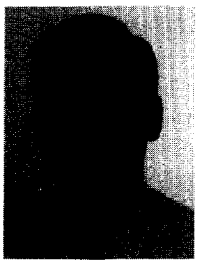
Theoretical arguments and simulation results show that the proposed control algorithm performs an effective congestion control in high speed networks, guaranteeing no cell loss and fairness. The control scheme performs very well even under the practical constraints of the EPRCA implementation in an ATM network. With the proposed EPRCA, every source adjusts its rate based on the delayed queue occupancy value fed back from the congested link and on the number of cells transmitted during the last round trip time. Therefore, as a difference from a "blind" additive increase/multiplicative decrease policy, our scheme implements a feedback regulation which is based on a rigorous control model. Some of the assumptions that have been made in the course of this report can be relaxed in a real network scenario. For instance, if queueing delays at intermediate nodes are significant when compared with propagation delays, then a periodic estimation of the ABR connection round trip delay including such delays can be used in conjunction with the control scheme. Also, a version of the control schemes using a common queue per output link are also possible [5,10].

Acknowledgements

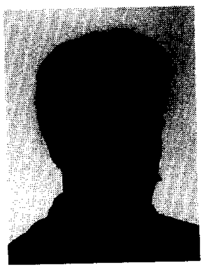
We would thank anonymous reviewers for their helpful comments.

References

[1] K.J. Astrom and B. Wittenmark, *Computer Controlled System*, Prentice-Hall, Englewood Cliffs, NJ (1990).
[2] A.W. Barnhart, Baseline performance using PRCA rate-control, ATM Forum/94-0597, July 1994.
[3] L. Benmohamed and S.M. Meerkov, Feedback control of congestion in packet switching networks: The case of a single congested node, *IEEE/ACM Trans. Networking* **1** (b) (1993) 693–708.
[4] F. Bonomi and K.W. Fendick, The rate-based flow control framework for the available bit rate ATM service, *IEEE Network* (March/April 1995) 25–39.
[5] D. Cavendish, M. Gerla and S. Mascolo, ATM rate based congestion control using a Smith Predictor: Implementation issues, *Proc. 1st Workshop on ATM Traffic Management WATM'95* (1995) 289–296.
[6] D. Cavendish, Y. Oie, M. Murata and H. Miyahara, Proportional rate-based congestion control under long propagation delay, *Int. J. Commun. Systems* **8** (1995) 79–89.
[7] H.T. Kung, R.M. Morris, T. Charuhas and D. Lin, Use of link-by-link flow control in maximizing ATM network performance: Simulation results, *Proc. IEEE Hot Interconnects Symp.*, Palo Alto, CA, August 1993.
[8] S. Mascolo, D. Cavendish and M. Gerla, SP-EPRCA: an ATM rate based congestion control scheme based on a Smith predictor, UCLA Technical Report No. 960001, February 1996.
[9] O.J. Smith, A controller to overcome dead time, *ISA J.* **6** (2) (1959) 28–33.
[10] S. Mascolo and M. Gerla, Classical control approach to congestion avoidance in high speed ATM networks, *Proc. IEEE ATM'97* (1997) 361–367.



Saverio Mascolo was born in Bari, Italy, in 1966. He received the Laurea degree in Electrical Engineering, with honor, in 1991 and the Ph.D., with a dissertation on deadlock avoidance in flexible manufacturing systems, in 1995, both from Polytechnic of Bari. During 1995 he was a visiting scholar at the Computer Science Department of University of California, Los Angeles. Since 1996 he is an Assistant Professor in Automatic Control at the Electrical and Electronic Department of Polytechnic of Bari. His main research interests include flow control in high speed communication networks, control of discrete event systems and nonlinear control theory. He is a member of IEEE, of Control System Society and of Communication Society.



Mario Gerla was born in Milan, Italy. He received the graduate degree in Electrical Engineering from Politecnico di Milano, Italy in 1966 and the M.S. and Ph.D. degrees in Computer Science from University of California, Los Angeles in 1970 and 1973, respectively. From 1973 to 1976, Dr. Gerla was a manager in Network Analysis Corporation, Glen Cove, NY, where he was involved in several computer network design projects for both government and industry, including performance analysis and topological updating of the ARPANET under a contract from DoD. From 1976 to 1977, he was with Tran Telecommunication, Los Angeles, CA, where he participated in the development of an integrated packet and circuit network. Since 1977, he has been on the Faculty of the Computer Science Department of UCLA. His research interests include the design, performance evaluation, and control of distributed computer communication systems and networks.