Stability analysis of ABR congestion control algorithms

Omar Ait-Hellal

INRIA sophia Antipolis,
BP 93, 2004 route des lucioles,
06902 Sophia Antipolis Cedex, France.
{oaithel, altman}@sophia.inria.fr

Abstract

In this paper, we investigate performances (stability, convergence ...) of the *Fixed Point [16], ERAQLES [18]* and ERICA+ [12] ABR congestion control algorithms. By using the stability theory of delayed differential equations, we get the stability conditions for each algorithm and examples where these algorithms are not stable (the stability definition is given bellow) are given. We propose parameters for which ERICA+ is always stable. We show that for a range of available bandwidth neither the *Fixed Point* nor *ERAQLES* algorithms are stable, whatever the parameters are (round trip times, ...).

Keywords

ATM, Available Bit Rate, closed-loop congestion control, delayed differential equations, fluid approximation.

1 Introduction

Adaptive mechanisms for congestion control have a central role in the efficient sharing of the network resources between many users. These mechanisms have also the role of preventing congestion in the network [7, 11]. The fact, however, that the control is performed by the sources (that are not policed by the network), and not by the network, make it hard to protect the network from applications that might not use such mechanisms (e.g. from video conferences that use UDP).

In ATM networks, the ABR (Available Bit Rate) service [1] has been defined for supporting best effort applications, in which the control decisions are taken by the network in the switches, ensuring to achieve fairness among the active connections and controlling the loss cell ratios. This service manages the bandwidth leftover by applications that have guaranteed performance (VBR and CBR), and shares it between the ABR sources by signaling to them their allowable transmission rate. The behavior of the source and destination is specified in [1] as well as the manner in which feedback information should be conveyed back to the source [1, 15]. The behavior of the switches, however, is left to the designer of the switch.

Several controllers have been proposed for the switches. They are either based on (i) the EFCI (Explicit Forward Congestion Indication) bit which originates from the approach of the DEC bit [21]; it indicates, whether the congestion is detected or not, or (ii) on ER (Explicit Rate) which informs the source on the bandwidth that is available (see next section and [9, 10, 13, 14, 20, 23] and references therein).

In the present paper, performances of the widely known ("interesting") algorithms in each class are investigated. We show that for a range of available bandwidths, there are always losses for both *ERAQLES* and *Fixed Point* algorithms. The stability domain is computed for the latters as well as for *ERICA+* algorithm and examples of instability are given.

The paper is structured as follows: In section 2 we state different mathematical tools used a long the paper (stability definition, and theorems related to the stability). EFCI-based and Queue length-based algorithms (Fixed Point) are presented in section 3. In section 4, ERAQLES and ERICA algorithms are presented and analyzed. Finally, We conclude with some remarks and future work in section 5.

2 Background

In what follows we shall use the stability theory of delayed differential equations. We are interested in this method, because an exact evaluation of such equations (differential equations with delay) is not known in many cases, hence only the stability can be analyzed in some cases. This method is sufficient to discuss the stability of many rate-based flow control algorithms (*ERICA+* [12], *ERAQLES* [18], Fixed-Point [16],... etc).

We begin by stating the definition of *the stability of a solution* [6] (**local stability**). We will illustrate this for the following differential-difference equation

$$\dot{x}\ (t) = f(x(t), x(t-\tau), t), \qquad \dot{x}\ (t) \triangleq \frac{dx(t)}{dt}. \tag{1}$$

Definition 2.1. *Let $w(t)$ be a functions, continuous for $t > 0$, which satisfies the equation in (1) for $t > \tau$. This solution is said to be* **stable** *as $t \to \infty$ if, given two positive numbers t_0 and ϵ, there exists a corresponding positive number δ such that every continuous solution $x(t)$ of the equation in (1) which satisfies*

$$\max_{t_0 \le t \le t_0+\tau} |x(t) - w(t)| \le \delta \tag{2}$$

will also satisfy

$$\max_{t_0 \le t} |x(t) - w(t)| \le \epsilon \tag{3}$$

The solution is said to be **uniformly stable** *if, given ϵ, there exists a δ such that for any $t_0 > 0$ and any solution $x(t)$ which satisfies (2) will also satisfy (3).*

The solution $w(t)$ is said to be **asymptotically stable** *if*

(a) it is stable;

(b) for each $t_0 \ge 0$ there is a δ such that every solution $x(t)$ which satisfies (2) will also satisfy

$$\lim_{t\to\infty} [x(t) - w(t)] = 0. \tag{4}$$

Theorem 2.1. *Consider the following differential equation*

$$\frac{dx(t)}{dt} + \alpha \frac{dx(t-\tau)}{dt} + \beta x(t) + \gamma x(t-\tau) = 0$$

Assume $|\alpha| \ne 1$, then the following statements are true:

1. *If $|\alpha| > 1$, then $x(t)$ is unstable for all positive delay τ.*

2. *If $|\alpha| < 1$, $\gamma^2 < \beta^2$, or $\gamma = \beta \ne 0$, then increasing τ does not change the stability of $x(t)$.*

3. *If $|\alpha| < 1$, $\gamma^2 > \beta^2$, and*

   (i) *$\gamma + \beta < 0$, then $x(t)$ is unstable for all positive delay τ;*

   (ii) *$\gamma + \beta > 0$, then $x(t)$ is uniformly asymptotically stable when $\tau < \tau_0$ and unstable when $\tau > \tau_0$, where $\tau_0 = \theta/w$, and*

$$\begin{aligned} w &= \left(\left(\gamma^2 - \beta^2\right)\left(1 - \alpha^2\right)^{-1}\right)^{1/2}, \\ \theta &= arccot\left(-\frac{\alpha w^2 + \beta\gamma}{w\left(\gamma - \beta\alpha\right)}\right) \end{aligned}$$

Next we state a generalization of the Poincaré-Lyapunov theorem [6] to differential-difference equations.

Theorem 2.2. *Consider the following nonlinear equation*

$$a_0 u'(t) + b_0 u(t) + b_1 u(t-\tau) = f(u(t), u(t-\tau)), \quad t > \tau; \quad a_0 \ne 0,\ b_0,\ b_1 \text{ are constants} \tag{5}$$

with initial condition

$$u(t) = g(t), \qquad 0 \le t \le \tau. \tag{6}$$

Suppose that

(a) every continuous solution of the linear part

$$a_0 u'(t) + b_0 u(t) + b_1 u(t - \tau) = 0; \tag{7}$$

approaches zero as $t \to \infty$*;*

(b) $f(u, v)$ *is a continuous function of* u *and* v *in a neighborhood of the origin* $|u| + |v| \le c_1$*; and*

(c)

$$\lim_{|u|+|v| \to 0} \frac{|f(u, v)|}{|u| + |v|} = 0. \tag{8}$$

Then, provided $\max_{0 \le t \le \tau} |g(t)|$ *is sufficiently small (depending on* c_1, a_0, b_0 *and* b_1*), any solution of the nonlinear equation (5), can be continued over the interval* $0 \le t < \infty$*, and each such solution satisfies*

$$\lim_{t \to \infty} |u(t)| = 0. \tag{9}$$

The proofs of the Theorems can be found in [17] *resp.* [6] (Theorem 3.2,pp. 82 *resp.* Theorem 11.2 pp. 336).

Remark 2.1. *In the sequel, we shall talk about the stability of the zero-solution. We will encounter equations where* $x(t) = 0$ *is a solution, then we will analyze the stability of such a solution. If the zero solution is uniformly asymptotically stable, this means that all solutions converge to* $x(t) = 0$*.*

Also, we shall ignore the delay between RM cells, and consider that the ACR is continuously updated. This assumption has a negligible effect on the stability domain, since ACR is constant between two RM cells.

3 EFCI-based and Queue length-based Algorithms

Several algorithms based on the congestion notification were proposed, PRCA, EPRCA ... (see [8, 19, 20, 23]), most of them translate the congestion to the buffer occupancy, so that the congestion is signaled depending whether the buffer is larger or smaller than a given threshold. Theoretical approach in studying these schemes always consider a single bottleneck node.

In recent study [2, 3] where, mainly, EFCI scheme was considered, we showed that the maximum buffer size in these schemes, strongly, depends on the number of switches in the path when considering a single source and it may be worst in case of multiple sources. This is due to the fact that the notion of congestion differs from a switch to another if the switches have different thresholds, and the informations about the congestion are binary (there is a congestion or there is not). Hence a continued informations on the buffer occupancy is needed.

Following the latter idea, other algorithms, based on the queue length are proposed. The information conveyed by the switches back to the source are a function of the queue length at a time a backward RM cell is received. In [16], a simple Queue-Length-based scheme, called a *Fixed Point* Algorithm is presented. The scheme shows very interesting features mainly the convergence, and the smoothness in the rate at the source. However no extensive simulations neither stability analyzes were studied. In the following we present this algorithm, as an example of the queue length based algorithms, and analyze its stability. We show that for a range of available bandwidths, the scheme is not stable whatever the round trip time is.

3.1 Fixed Point Algorithm

This new algorithm is, essentially, based on the queue size at the intermediate switches. More precisely, it is based on the degree of saturation of each switch (i.e. the fraction of the queue size over a certain threshold which may be the maximum buffer size at the switch). The aim is to keep the queue length close to the threshold, by varying the rate at which the source sends its data, inversely proportional to the buffer occupancy.

It is obvious that this algorithm is better than those based on the linear increase-exponential decrease of the rate, namely, EFCI based switch algorithms in all performances criterions. It has the same complexity while it requires much less buffers and allows different sources to get relatively the same bandwidth (fairness is achieved for the long run)

In this algorithm, RM cells are sent following the recommendation of *ATM Forum* [1]. The source is allowed to send data at rate no more than PCR_i. At the beginning the source i sends its data at rate ICR_i. In our analysis we don't take into account the initial conditions (they don't affect the analysis (study)). The rate at the source may increase or decrease according to the information carried in the backward RM Cells $(y(t)/Q)$. Each time an RM cell is received at the source, the source adjusts its rate as follows:

$$ACR := ACR + AIR * N_{rm} \left(1 - \frac{ACR}{PCR}\right) - TQ * ACR \tag{10}$$

where TQ is the field in the coming RM cell, updated by the switch, we suppose that backward RM cells are updated.

When a backward RM cell is received at the switch, the field TQ is set to the queue size at the switch divided by the maximum queue size (buffer capacity).

$$TQ := \frac{\text{queue size}}{\text{maximum queue size}} \tag{11}$$

Analytic approach

Here we aim to analyze this Algorithm and to find out its insufficiencies. We use the differential equations to model it. We consider several equivalent saturated sources sending to the same sink and sharing a bottleneck node whose capacity is μ, as constant. The round trip time for an RM cell is τ. The buffer size is of Q buffers.

Define:

$\beta_i \triangleq$ The time that takes a cell from the bottleneck (queue) to reach the source i.

$\gamma_i \triangleq$ The time that takes a cell from source i to reach the bottleneck (queue).

$\tau_i \triangleq \beta_i + \gamma_i$.

$y(t) \triangleq$ Queue size at time t.

$x(t) \triangleq$ The rate of the source at time t.

$\bar{y} \triangleq$ queue size when the system stabilizes.

Remark 3.1. *In all the paper we will consider the stability of algorithms in a system where equivalent sources are considered. This gives us a global idea on the algorithm in other systems. Indeed,* **generally**, *if in a system where equivalent sources are considered, a given algorithm is unstable for all $\tau > \tau_0$* resp. *stable for all $\tau < \tau_1$; then, in a system where non equivalent sources exist, the algorithm will be unstable if* $\min_{i\geq 1}(\tau_i) > \tau_0$ resp. *stable if* $\max_{i\geq 1}(\tau_i) < \tau_1$.

We suppose that the queue never empties and fairness is guaranteed among the N sources. From equations (10) and (11), we have the following:

$$\begin{cases} x_i(t) = x_i(t^-) + AIR * N_{rm}\left(1 - \dfrac{x_i(t^-)}{PCR}\right) - \dfrac{y(t-\beta_i)}{Q} x_i(t^-) \\ \\ y(t+\delta t) = \left(y(t) + \left(\displaystyle\sum_{i=1}^{N} x_i(t-\gamma_i) - \mu\right)\delta t\right)^+ \end{cases} \tag{12}$$

Where $(x)^+ = max(x, 0)$, and t^- is the last time the ACR was updated (last time an RM cell has arrived). Since the queue is supposed to be always non empty (the rate of RM cells is constant), and if the delay between RM cells is neglected, ACR can be supposed , indeed, as the fixed point of the above equation $x(t) = x(t^-)$ (the rate reacts immediately to the buffer changes such that ACR will be stable). Hence from (12) we get

$$\begin{cases} x_i(t) \cong AIR * N_{rm} \left(\dfrac{PCR * Q}{AIR * N_{rm} * Q + PCR * y(t-\beta_i)} \right) \\ \dot{y}\ (t) = \displaystyle\sum_{i=1}^{N} x_i(t-\gamma_i) - \mu \end{cases} \tag{13}$$

From (13) we have

$$\begin{aligned} \dot{y}\ (t) &= \sum_{i=1}^{N} x_i(t-\gamma_i) - \mu = \sum_{i=1}^{N} x_i(t-\gamma_i) + \sum_{i=1}^{N} AIR * N_{rm} - \frac{AIR * N_{RM}}{PCR} \sum_{i=1}^{N} x_i(t-\gamma_i) \\ &\quad -\frac{1}{Q} \sum_{i=1}^{N} (y(t-\tau_i) x_i(t-\gamma_i)) - \mu \\ &= \left(1 - \frac{AIR * N_{RM}}{PCR}\right) \dot{y}\ (t) + \frac{(N * PCR - \mu) AIR * N_{rm}}{PCR} \\ &\quad -\frac{1}{Q} \sum_{i=1}^{N} \left(y(t-\tau_i) \left(x_i(t-\gamma_i) - \frac{\mu}{N} + \frac{\mu}{N} \right) \right) - \frac{AIR * N_{RM}}{PCR} \left(\sum_{i=1}^{N} x_i(t-\gamma_i) - \mu \right) \end{aligned} \tag{14}$$

Let

$$H(t) \triangleq y(t) - \frac{(N * PCR - \mu) AIR * N_{rm} * Q}{\mu * PCR} \quad \text{and} \quad \dot{H}_i\ (t) \triangleq x_i(t-\gamma_i) - \frac{\mu}{N} \Longrightarrow \sum_{i=1}^{N} \dot{H}_i\ (t) = \dot{H}\ (t)$$

Then from (14), we have

$$\dot{H}\ (t) + \frac{\mu^2}{N^2 * AIR * N_{rm} * Q} \sum_{i=1}^{N} H(t-\tau_i) = -\frac{\mu}{N * Q * * AIR * N_{rm}} \sum_{i=1}^{N} H(t-\tau_i)\ \dot{H}_i\ (t) \tag{15}$$

Proposition 3.1. *For equivalent sources ($\tau_i = \tau\ \forall i$), if*

$$\tau < \tau_0 \triangleq \frac{\pi}{2} \frac{N * Q * AIR * N_{rm}}{\mu^2}$$

then the zero-solution of (15) $\left(H(t) = 0 \Longleftrightarrow y(t) = \frac{(N * PCR - \mu) AIR * N_{rm} * Q}{\mu * PCR} \right)$ *is uniformly asymptotically stable.*

Proof:

The proof follows from Theorems 2.1 and 2.2 after remarking that for equivalent sources, we have

$$(15) \quad \Longleftrightarrow \quad \dot{H}\ (t) = -\frac{\mu^2}{N * AIR * N_{rm} * Q} H(t-\tau) \left(\frac{N * AIR * N_{rm} * Q}{N * AIR * N_{rm} * Q + \mu H(t-\tau)} \right)$$

$$\iff \dot{H}(t) + \frac{\mu^2}{N*AIR*N_{rm}*Q}H(t-\tau) = \frac{1}{N*AIR*N_{rm}*Q} \times$$

$$\frac{\mu^3\,(H(t-\tau))^2}{N*AIR*N_{rm}*Q+\mu H(t-\tau)} \triangleq f(H(t-\tau)) \qquad (16)$$

and since,

$$\lim_{|u|\to 0}\frac{|f(u)|}{|u|} = 0$$

then from Theorem 2.2, the stability of the zero-solution of the linear part of equation (16) implies the stability of the zero-solution of the hole equation (16). The necessary and sufficient condition for the stability of the linear part of (16), follows by a direct application of Theorem 2.2. ∎

Figure 1 shows that the instability region of the *Fixed Point* algorithm is very close to τ_0 ($\tau > \tau_0$). The parameters considered are the following: We considered one source going through a single switch where the available bandwidth is $\mu = 5000\ cells/sec$ (1.272 Mbps). $AIR = 128$, $PCR = 33333.3250\ cells/sec = 14.13\ Mbps$ $Q = 200$ and $ICR = 2000\ Cells/sec$. From Proposition 3.1, we get $\tau_0 = 0.05147\ sec$. Thus the trivial solution of (16) is stable for $\tau \leq \tau_0$. In Figure 1, we can see that for the *Fixed Point* algorithm, the system is not table for that value ($\tau = 0.050\ sec$). However it is stable for values, lightly, smaller ($\tau = 0.044\ sec > \tau_0 - 1/AIR$).

In fact the value obtained in Proposition 3.1 is optimistic. At the same time the unstable region for the *Fixed Point* algorithm is, indeed, very close to that value. The following heuristic is generally true: for the *Fixed Point* algorithm, the system is stable for $\tau < \tau_1$ and unstable for $\tau > \tau_1$ where τ_1 satisfies $\tau_0 > \tau_1 > \tau_0 - 1/AIR$.

Now suppose that the system is stable, (the rate at the sources stabilizes as well as the queue length) then (after setting $Q = \xi B$, $\xi < 1$), from Proposition 3.1 we get

$$\bar{y} = \frac{(N*PCR-\mu)AIR*N_{rm}*Q}{\mu*PCR} = \frac{(N*PCR-\mu)AIR*N_{rm}*\xi B}{\mu*PCR} \qquad (17)$$

This result can be found in [16]. It is easy to remark that the buffer size when the system stabilizes can exceed the maximum buffer size for some values of μ. In other words let state the following lemma:

Lemma 3.1. *The switch experiences losses (there is no stability), for*

$$\mu < \frac{N*PCR}{1+\frac{PCR}{AIR*N_{rm}*\xi}}$$

This is obtained from Proposition 3.1 by setting $\bar{y} > B$. The limitation due to the smaller available bandwidth constitutes the main (and huge) inconvenient for the proposed algorithm, even if a fraction of the maximum queue size (a certain threshold) is considered rather than the maximum queue size itself. On the other hand large available bandwidth causes the instability of the system (see Proposition 3.1).

4 ER-based Algorithms

4.1 ERAQLES Algorithm

ERAQLES algorithm [18] (**E**xplicit **R**ate **A**lgorithm using **Q**ueue **LE**ngth **S**tates) proposes to evaluate the left over CBR VBR bandwidth in an another manner than *ERICA*, by using the variation of ABR queue length. In the sequel we suppose that the available bandwidth is well estimated (as in *ERICA* algorithm). This algorithm is close to *ERICA* algorithm, since the attributed rate to each source is proportional to its ratio in the total incoming ABR traffic $\frac{CCR_i}{CCR}$ (utilization of the $Load\_Factor$ in *ERICA*, see next section for the *ERICA* description). It can be summarized, briefly, as follows: At the source no changes, the rate is updated according to the ER field in incoming RM cells to the source.

At the switch

the explicit rate for source i is computed using

$$ER_i(t) = ER(t).s_i(t)$$

where $s_i(t)$ is the fair share function, it is computed using

$$s_i(t) = (1 - s_0)\frac{CCR_i(t-T)}{CCR(t-T)} + s_0$$

where $CCR_i(t)$ is the rate of source i at time t and $CCR(t)$ is the sum of the rates of all sources $(0 < s_0 < 1)$, T is the time between the source and the switch.

$ER(t)$ is the bandwidth not used by other higher priority (CBR, VBR) traffics plus the bandwidth able to fill up the buffer capacity during a feedback (τ). Finally, ER is set to the minimum of ER in RM cell and $ER_i(t)$ calculated. $ER(t)$ is given by:

$$ER(t) = C^* + h\frac{Q - y(t)}{T}$$

Where C^* is the left over CBR VBR traffic, and $y(t)$ is the queue length at time t. Q is a given threshold $(Q = \xi B$, $\xi < 1$ and B is the maximum buffer size).

Analytic Approach

Consider the same model and parameters as previous, then the evolution of $x(t)$ and $y(t)$ is given by:

$$\begin{cases} x_i(t) = \left((1-s_0) \dfrac{x_i(t-\tau_i)}{\sum_{\ell=1}^{N} x_\ell(t-\gamma_\ell-\beta_i)} + s_0 \right) \left(\mu + h \dfrac{Q - y(t-\beta_i)}{T} \right) \\ y(t+\delta t) = \left(y(t) + \left(\sum_{i=1}^{N} x_i(t-\gamma_i) - \mu \right) \delta t \right)^{+} \end{cases} \tag{18}$$

$$\text{Let} \qquad H(t) \triangleq y(t) - Q - \frac{(N-1)s_0\mu T}{((N-1)s_0+1)h} \tag{19}$$

After some calculations, with considering that the buffer never empties and equivalent sources, and by following the same steps as previous, we get

$$\dot{H}(t) + h\frac{(N-1)s_0+1}{T} H(t-\tau) = 0 \tag{20}$$

From Theorem 2.1, the zero-solution of (20) $\left(H(t) = 0 \Longleftrightarrow y(t) = Q + \frac{(N-1)s_0\mu T}{((N-1)s_0+1)h}\right)$ is uniformly asymptotically stable if and only if

$$\tau < \tau_0 = \frac{\pi T}{2h((N-1)s_0+1)}$$

Hence, the scheme is not stable for all parameters, its stability depends, mainly, on h and s_0. If h satisfies ($s_0 = 1/N$, $2T = \tau$)

$$\frac{\pi}{4h((N-1)s_0+1)} < 1 \Longrightarrow \frac{\pi}{4((N-1)s_0+1)} \leq 0.785 \leq h \tag{21}$$

then no stability is guaranteed. However the parameters recommended in [18] ($s_0 = 1/N$ and $h = 0.1839$) insure the stability in all cases. Figure 2 illustrates the case of one source for $h = 0.785$ (unstable) and $h = 0.70$ (stable). The parameters considered are the same parameters as previous (*Fixed Point* algorithm, stable case $T = 0.05$ sec).

As we can see from equation (19), when the system stabilizes the queue length depends on the available bandwidth, as the case for the *Fixed Point* algorithm. After some calculations (we consider that $Q = \xi B$ where B is the maximum buffer size; $\xi < 1$), in the same manner as for the *Fixed Point* algorithm we prove the following lemma (using equation (21)):

Lemma 4.1. *The switch experiences losses (there is no stability) for*

$$\frac{\tau}{2} \geq \frac{(1-\xi)B((N-1)s_0+1)h}{(N-1)s_0\mu} \Longrightarrow \tau \geq \frac{\pi}{2}\frac{(1-\xi)B}{(N-1)s_0\mu}$$

We can consider $s_0 = 0$ to get a stable system in all cases, however the proof of the fairness [18] (even the proof was stated, only, for equidistant sources) will not be valid. In fact for s_0 too small, the fairness is achieved in a time relatively long, and for s_0 large the stability might be non guaranteed, as well as the queue length might exceed the maximum buffer size (see Lemma 4.1). This is the main disadvantage of this algorithm.

4.2 ERICA(+) switch algorithm

ERICA+ [12], proposed by Jain et al, is an algorithm based on the available bandwidth together with the buffer occupancy at intermediate switches. It is, actually, the most referenced switch algorithm. Extensive simulations were done and many enhancements were proposed. In all simulations, the scheme is reported as a stable one. In the present section we aim to investigate the stability of the scheme with considering different parameters.

ERICA switch algorithm works as follows (for the lack of space many parameters were neglected, and only a sketch of the algorithm is presented in this paper. For more details see [12]):

The switch determines periodically on each link the load factor, z, the available capacity and the number of active VCs (N). The load factor is computed as:

$$z \leftarrow \frac{\text{ABR Input Rate}}{\text{ABR Capacity}}$$

$$\text{Where} \quad \text{ABR Capacity} \leftarrow \text{Target Utilization} \times \text{Link Bandwidth} - \text{VBR Usage} - \text{CBR Usage}$$

In order to use 100 % of the available bandwidth *ERICA* introduces the buffer occupancy as a control parameter (*ERICA+*) and the *Target Utilization* is set to one, so that ABR Capacity is revised as (we consider a constant threshold (Q)).

$$\text{ABR Capacity} \leftarrow \text{ABR Capacity} \times \frac{b * Q}{(b-1) \times \text{Queue Length} + Q}$$

Where the recommended values for b are

$$b = 1.15 \text{ if Queue Length} > Q; \quad b = 1.05 \quad \text{otherwise}$$

Finally the rate for each connection is updated as follows (recall that the *Load Factor* also is changed):

$$ER\_Calculated \leftarrow \text{Max}\ \left(\frac{\text{ABR Capacity}}{\text{Number Of Active Sources}}, \frac{CCR[VC]}{z}\right)$$

$CCR[VC]$ is the rate in the incoming forward RM cell of the VC, if the per-VC counting is not supported; otherwise it is the *Input Rate* of that VC (Number of Input Cells/Averaging Interval).

Analytic Approach

To analyze the stability of *ERICA+*, we considered the case where *ERICA* aims a 100 % utilization, and we used a constant queue threshold (Q). The rate and the buffer size update in *ERICA+* can be summarized in the following equations (we neglect the delay between RM cells):

Consider the same model and parameters as previous:

$$\begin{cases} x_i(t) = max\left(\frac{1}{N}, \frac{x_i(t-\tau_i)}{\sum_{\ell=1}^{N} x_\ell(t-\gamma_\ell-\beta_i)}\right) * ER\frac{b*Q}{(b-1)y(t-\beta_i)+Q} \\ y(t+\delta t) = \left(y(t) + \left(\sum_{i=1}^{N} x_i(t-\gamma_i) - \mu\right)\delta t\right)^+ \end{cases} \tag{22}$$

b takes two values, as discussed above, depending whether the queue length is larger or (*resp.*) smaller than Q (the recommended values are 1.15 resp. 1.05). In the analyze we consider the value as a unique and we will identify the stability domain in the worst case (which will correspond to the smallest value of b).

$$\text{Let} \qquad H(t) \triangleq y(t) - Q$$

Proceeding in the same manner as in the previous section, after some calculations for the case of equivalent sources, we get from (22)

$$\dot{H}(t) + \frac{(b-1)\mu}{bQ} H(t-\tau) = -\frac{(b-1)}{bQ} H(t-\tau)\,\dot{H}(t) \tag{23}$$

Proposition 4.1.

$$\textit{If} \qquad \tau < \tau_0 = \frac{\pi}{2}\frac{bQ}{(b-1)\mu},$$

then the zero-solution of (23) ($H(t) = 0 \Longleftrightarrow y(t) = Q$) is uniformly asymptotically stable.

Proof:

The proof is similar to that of Proposition 3.1, after remarking that

$$(23) \quad \Longleftrightarrow \dot{H}(t) = -\frac{(b-1)\mu}{bQ} H(t-\tau)\left(\frac{bQ}{bQ+(b-1)H(t-\tau)}\right)$$

$$\Longleftrightarrow \dot{H}(t) + \frac{(b-1)\mu}{bQ} H(t-\tau) = \frac{\mu((b-1)H(t-\tau))^2}{(bQ)^2 + bQ(b-1)H(t-\tau)} \triangleq f(H(t), H(t-\tau))$$

∎

Figure 3 (a) illustrates an example where, *ERICA+* algorithm is not stable. The parameters considered in this example are the following: the available bandwidth is $\mu = 100000\ Cells/sec$, the threshold Q is set to $200 Cells$ and $\tau = 0.064\ sec$. From Proposition 4.1 we get $\tau_0 = 0.06597$ for $b = 1.05$, and $\tau_0 = 0.02408$ for $b = 1.15$. Hence we deduce that the instability region is between these two values. In many examples, we found that ERICA+ is unstable if $\tau > \tau_0$, where τ_0 is the value obtained for $b = 1.05$. Figure 3 (a) confirms these claims.

From Proposition 4.1, we can choose b such that we insure the stability in all cases. For example instead of $fn(Queue) \triangleq \dfrac{bQ}{(b-1)y(t)+Q}$, we define $fn(Queue)$ as : $fn(Queue) \triangleq \dfrac{Q}{by(t)+(1-b)Q}$.

$$\text{Where} \quad b \triangleq \begin{cases} Min\left(0.05, \dfrac{\pi Q}{6 * ER * \tau_{max}}\right) & \text{if} \quad \text{Queue size} < Q \\ Min\left(0.15, \dfrac{\pi Q}{2 * ER * \tau_{max}}\right) & \text{Otherwise} \end{cases}$$

Figure 3 (b) shows the effects of the new function $fn(Queue)$. The rate as well as the queue length are constant rather than oscillatory (as it was the case in ERICA+).

5 Conclusion

In the present work, the stability of different ABR congestion control algorithms was studied, and the stability domain of each one is stated. Although the *Fixed Point* algorithm is, fair, and very simple, it guarantees no stability under some configurations. *ERAQLES* algorithm allows a very fast convergence, however, the buffer size at intermediate switches might be very large (very aggressive). Finally, we showed that even for *ERICA+* there is situations where the algorithm is not stable, and proposed values for the parameter b which enables to get better performances.

ERICA+ is known for the multiplicity of parameters. For the fairness purpose and the full utilization, ERICA+ has to store *Last_Allocated_rate* for each active connection, its contribution, the average active sources, *CCR* for each connection and so on... This constitutes the main inconvenient of ERICA. We are, currently, working on a new explicit rate scheme which enables to get better performances than ERICA+. This scheme requires only informations on the *CCR* of each connection and the number of established ABR connections.

References

[1] The ATM forum Technical Committee, *Traffic Management Specification,* Version 4.0, April 1996.

[2] O. Ait-Hellal, E. Altman, D. Elwadghiri and M. Erramdani, “Performance Evaluation of the Rate-Based Flow Control Mechanism for ABR Service”, *INRIA Research Report No. RR-3131. Second IFIP Workshop on Traffic Management and synthesis of ATM networks*, Montreal, 24-26 Sept. 1997.

[3] O. Ait-Hellal, E. Altman, “Performance Evaluation of the Rate-Based Flow Control Mechanism for ABR Service: Generalization”, *INRIA Research Report*.

[4] O. Ait-Hellal, E. Altman, T. Basar, “Rate based flow control with bandwidth information”, *proceeding of the 35th Conference on Decision and Control*, pp. 2911-2916, Kobe, Japon. December 1996.

[5] E. Altman, F. Baccelli and J. C. Bolot , “Discrete-time analysis of adaptive rate control mechanisms” , *High Speed Networks and their performance*, H. G. Perros and Y. Viniotis Eds., North Holland, pp. 121-140, 1994.

[6] R. Bellman, K. L. Cooke, “Differential-Difference Equations”, *Academic press*, 1963.

[7] L.S. Brakmo, L.L. Peterson, “TCP Vegas : End to End Congestion Avoidance on a Global Internet,” *IEEE Journal on selected Areas in communications,* vol. 13, pp. 1465-1480, 1995.

[8] M. O. Brouillet and U. Madhow, “Rate control for adaptive bit rate sources on ATM networks using one bit congestion notification”, Research report UILU-ENG-96-2214, University of Illinois at Urbana-Champaign, May, 1996.

[9] Y. Chang, N. Golmie, D. Siu, "A Rate-Based Flow Control Switch esign for ABR Service in an ATM Network,” *Twelfth International Conference on Computer Communication ICCC'95,* August 1995.

[10] A. Charny, D.D. Clark, R.Jain, "Congestion Control With Explicit Rate Indication,”*Proc. ICC'95,* June 1995, 10 pp.

[11] Van Jacobson, “Congestion avoidance and control,” *ACM SIGCOMM 88,* pages 273-288, 1988. ftp.ee.lbl.gov/papers/congavoid.ps.Z

[12] R. Jain, S. Kalyanaraman, R. Goyal, S. Fahmy, R. Viswanathan, "ERICA switch algorithm: A Complete Description," *ATM Forum/96-1172. http://www.cis.ohio-state.edu/ jain/atmf/a96-1172.htm.*

[13] R. Jain, S. Kalyanaraman, R. Viswanathan, "The OSU Scheme for Congestion Avoidance in ATM Networks using Explicit Rate Indication,” *Proceedings WATM'95 First Workshop on ATM Traffic Management,* Paris, December 1995.

[14] R. Jain, "Congestion Control and Traffic Management in ATM Networks: Recent Advances and A Survey,” *invited submission to Computer Networks and ISDN Systems.*

[15] R. Jain, S. Kalyanaraman, R. Goyal, S. Fahmy, "Source Behavior for ATM ABR Traffic Management: An Explanation," *IEEE Communications Magazine,* November, 1996.

[16] B.K. Kim, B.G. Kim, I. Chong, "Fixed Point Algorithm for ABR Congestion Control", AF-TM 96-0061, April 1996.

[17] Y. Kuang, "delay differential equations, with applications in population dynamics", *Academic press*, 1993.

[18] Y. Moret, S. Fdida, "ERAQLES an Efficient Explicit Rate Algorithm for ABR", *GLOBECOM'97*, Phoenix, USA, November 1997.

[19] H. Ohsaki, G. Hasegawa, M. Murata and A.Miyahara, "Parameter Tuning of Rate-Based Congestion Control Algorithms and its Application to TCP over ABR," *Proceedings WATM'95 First Workshop on ATM Traffic Management,* Paris, December 1995.

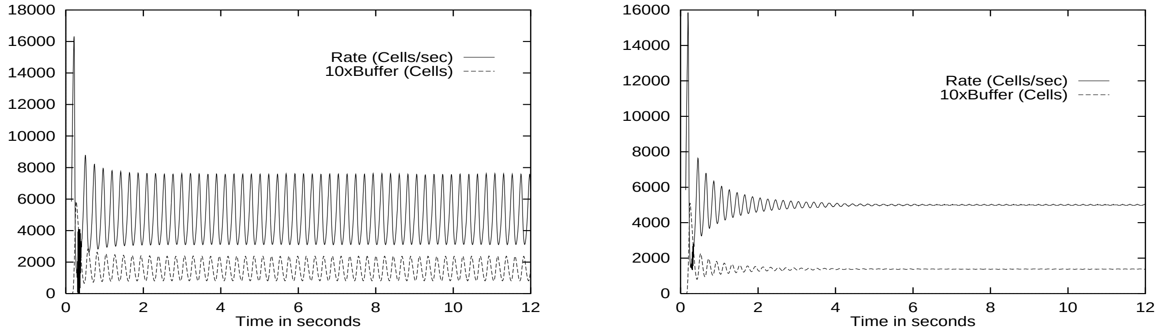
[20] H. Ohsaki, M. Murata, H. Suzuki, C. Ikeda and H. Miyahara, "Rate-based control for ATM Networks," *Computer Communication Review, acm-sigcomm,* special issue in ATM, ed. R. Jain and K.Y. Siu, Vol 25, No. 2, pp. 60-71, 1995.

[21] K. Ramakrishnan and R. Jain, "A Binary Feedback Scheme for Congestion Avoidance in Computer Networks with Connectionless Network Layer," *Proc. SIGCOMM'88,* August 1988.

[22] Michael Ritter, "Network Buffer Requirements of the Rate-Based Control Mechanism for ABR services," *IEEE INFOCOM'96,* san francisco, USA, March 1996.

[23] K.Y. Siu and H.Y. Tzeng, "Intelligent Congestion Control for ABR Service in ATM Networks," *Computer Communication Review, acm-sigcomm,* Vol 24, No. 5, pp. 81-106, 1995.

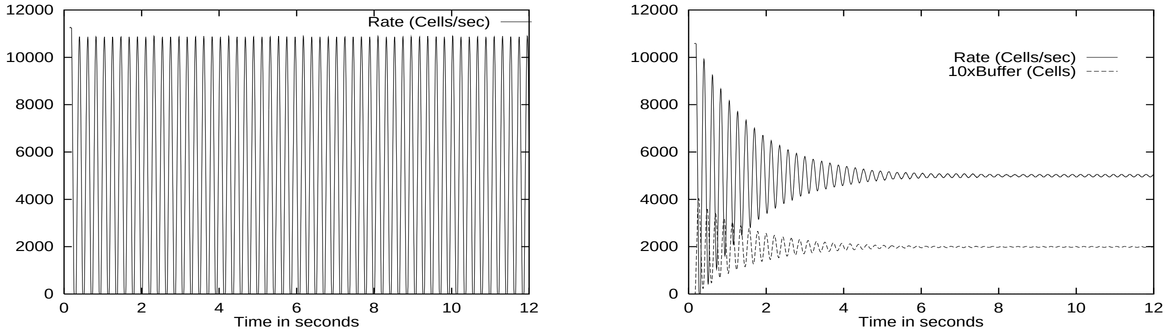
[24] N. Yin and M. G. Hluchyj, "On closed-loop rate control for ATM cell relay networks", *Proc. IEEE Infocom '94*, pp. 99-108, June 1994.



(a) $\tau = 0.050$

(b) $\tau = 0.044$

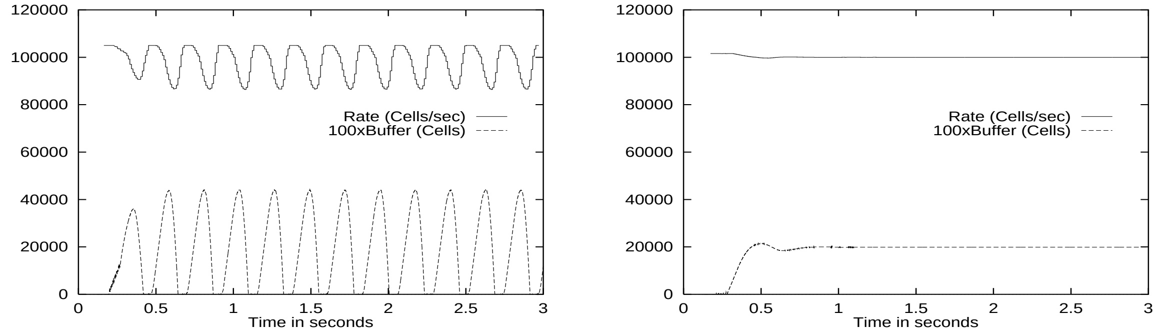
Figure 1: Buffer occupancy (Cells) and the rate (Cells/sec), for $\tau = 0.044$ sec (stable) and $\tau = 0.050$ sec (non stable), for the Fixed Point algorithm.



(a) $h = 0.785$

(b) $h = 0.700$

Figure 2: Rate in Cells/sec for $h = 0.70$ and $h = 0.785$, for ERAQLES algorithm.



(a) *ERICA+*

(b) enhanced ERICA+

Figure 3: Buffer occupancy ($100 \times Cells$) and the rate (Cells/sec), for $\tau = 0.064$ sec, $\mu = 100000$ $Cells/sec$ and $Q = 200$ $Cells$ for $ERICA+$ (before and after introducing a new function $fn(Queue)$).