APPLICATION OF ADAPTIVE CONTROL TO ATM ABR CONGESTION CONTROL

Kenneth P. Laberteaux and Charles E. Rohrs
Tellabs Research Center
3740 Edison Lakes Parkway, Mishawaka, IN 46545, USA
{klaberte, crohrs}@trc.tellabs.com

Abstract *- One of the more challenging and yet unresolved issues which is paramount to the success of ATM networks is that of congestion control for Available Bit Rate (ABR) traffic. Presented here is a new congestion control algorithm based on a well-understood Adaptive Controller, namely the Minimum Prediction Error Adaptive Controller. As such, stability and convergence characteristics can be rigorously proved. In addition, the work presented here distinguishes itself from others in its direct estimation of key parameters and removal of constrained and thus unresponsive sources.*

Topic: Congestion control in ATM networks, feedback control, traffic management.

Section 1 - Introduction

Critical to the success of ATM networks is the development of effective and fair methods for traffic management. One of the most challenging aspects of ATM traffic management is determining resource allocation for unpredictable and highly varying traffic. Generally, such traffic is carried with the Quality of Service (QoS) designation of *Available Bit Rate* (ABR). The standard for ABR traffic does not impose requirements for cell transfer delay or cell loss ratio, but both should be minimized. Key to this goal is avoiding congestion at any switching node in the ATM network; cells which arrive to a nearly full switch input buffer will experience excessive delay, while cells arriving to a completely full buffer are lost entirely.

Several schemes for congestion control of ABR traffic have been proposed [Jain96], the most promising algorithms are those designated as *Closed-Loop Rate-Based Traffic Management*, as proposed by Hluchyj [Hluchyj94]. *Closed-loop* refers to the fact that switches, which are receiving cells from a variety of ABR sources, send back to the sources instructions which will alter the rate at which each source sends cells into the network. The mechanism used for this switch-to-source communication consists of Resource Management (RM) cells, which are inserted with regularity by the source, returned by the destination back to the source, and then modified by the switches as needed to effect the congestion control algorithm. The term *Explicit-rate* indicates the fact that switches use the RM cells to request a specific desired rate as opposed to simply requesting an increase or decrease in rate. Each source is expected to adjust its cell rate to the new explicit rate as soon as possible after receiving the RM cell. A source which routes cells through multiple switches will observe the minimum explicit rate requested by the switches in its path.

Many algorithms for determining the explicit rate a switch should specify have been proposed. Most of these algorithms assume sources to be greedy but compliant. The ERICA algorithm and its extensions, presented by Jain [Jain95], and the Uniform Tracking (UT) algorithm, presented by Fulton and Li [Fulton97], have gained widespread attention.

In steady state, both algorithms achieve fairness and efficiency if they equally divide the bandwidth available for ABR traffic among the *competing* sources. Since each source sends cells at the minimum explicit rate specified by the switches in its path (each switch calculates its explicit rate independently), it is quite likely that a switch will carry traffic from a source constrained in its rate by another switch. Both algorithms supply the constrained sources their needed bandwidth, and, at least in steady state, equally divide the remaining bandwidth among the unconstrained sources.

Each algorithm reaches this goal in a distinct way. ERICA sends each constrained source a Fair Rate that it is unable to achieve. The resulting under-utilization of the available bandwidth pushes each unconstrained source up towards its equal share of the contested bandwidth.

The UT approach assumes there is one fair explicit rate that results from equally dividing the contested bandwidth by the number of contesting sources. However the contested bandwidth and number of contesting sources are not found directly. Instead the fair rate is found iteratively by comparing past explicit rates to the current total input rate.

The algorithm for congestion control presented in this paper is distinct in that it directly approximates the switch's total incoming bandwidth due to constrained sources and removes it before calculating a fair rate. In addition, this paper continues the effort of applying the methods of Control Theory to the issue of Closed-Loop ABR Congestion Control. Previous efforts include those by Rohrs, Berry, and O'Halek ([Rohrs96], [Rohrs97]), as well as others. Zhao and Li apply a known control structure to produce what they call the H2 scheme and prove its closed-loop stability in [Zhao96]. Fulton and Li then developed UT which "can be viewed loosely as

an adaptive H2 control" [Fulton97]. Stability of UT was not demonstrated analytically but instead with an extensive simulation study. This paper provides analytical results that confirm and clarify Fulton and Li's simulation results.

Contributed here is an application of an algorithm thoroughly understood in the literature of Adaptive Control, and as such its stability and convergence characteristics can be rigorously proven, even for generalized plants where the reaction times of various sources differ. Under certain conditions and assumptions, this algorithm bears a strong resemblance to the UT algorithm. Contributions to the understanding of the modeling of rate control problems also appear in this paper.

The remainder of this paper is organized as follows: Section 2 defines the plant to be controlled and states required assumptions. Section 3 makes further assumptions and then presents a one-parameter controller. This controller is then shown to be similar to the UT algorithm. Section 4 presents a two-parameter controller where the additional term directly estimates the constrained bandwidth present at the switch. Section 5 generalizes the plant from that presented in Section 2. The controller in Section 4 is then appropriately generalized. Concluding remarks are made in Section 6.

Section 2 - Preliminaries and Plant Definition

Consider the following plant: At each time index *n*, the node of interest *S* is required to carry *N(n)* simultaneous Available Bit Rate (ABR) sessions. All session sources are greedy, i.e. will send cells continuously at the maximum rate allowed by the switches through which they pass. Node *S* treats each of the *N(n)* sessions fairly. At each time *n*, node *S* has bandwidth $y^*(n)$ available for ABR traffic, and thus generates a single desired rate $u(n)$ that is sent to each of the *N(n)* sources. The switch expects each source to respond as quickly as possible to the new desired rate. The goal of *S* is then to choose $u(n)$ so as to minimize $|y(n)-y^*(n)|$, where $y(n)$ is the aggregate received input rate from the *N(n)* sources.

Although *N(n)* sources share the $y^*(n)$ of available bandwidth, we assume that a subset $N_c(n)$ of the *N(n)* sources are constrained to a rate lower than $u(n)$ by other nodes in each of the $N_c(n)$ respective paths. Thus only the $N_u(n)=N(n)-N_c(n)$ unconstrained sources will react to $u(n)$. The aggregate of the $N_c(n)$ constrained sources $C(n)=\sum_{i\in N_c} y_i(n)$ is assumed to be independent of $u(n+a)$ for any positive or negative *a*. The switch is assumed not to know the value of $N_u(n)$ or $C(n)$.

The round trip response delay for each of the $N_u(n)$ unconstrained sources is assumed to be equal and known by the switch to be *d*. Thus

$$y(n)=N_u(n)u(n-d)+C(n) \tag{1}$$

Section 3 - The One Parameter Controller

Fulton and Li propose an equivalent plant for Eq. (1) [Fulton97] that is slightly simpler than Eq. (1) but also has somewhat less fidelity to the real situation. Define the desired fair rate

$$u^*(n-d)=\frac{y^*(n)-C(n)}{N_u(n)} \tag{2}$$

Fulton and Li define a new *effective number of sources* $N_{eff}(n)$ where

$$N_{eff}(n)=\frac{y^*(n)}{u^*(n-d)} \tag{3}$$

Thus, they define their plant as

$$y(n)=N_{eff}(n)u(n-d) \tag{4}$$

Assuming for now that the plant in Eq. (4) is a valid model, we proceed to create a simple Minimum Prediction Error Adaptive Controller (Direct Approach) to determine the $u(n)$ which minimizes $|y(n)-y^*(n)|$ [Goodwin84]. As with the design of most adaptive controllers, for the purposes of analysis, we assume that the parameter $\theta_o=N_{eff}$ is constant within the order of time needed to generate an estimate $\hat{N}_{eff}(n)$ with accuracy. We will make similar assumptions in future sections.

Since we assume knowledge of *d*, we can use the following:

$$\hat{N}_{eff}(n)=\hat{N}_{eff}(n-1)+\frac{\mu}{u^2(n-d)}u(n-d)\left(y(n)-u(n-d)\hat{N}_{eff}(n-1)\right) \tag{5}$$

$$u(n)=\frac{y^*(n+d)}{\hat{N}_{eff}(n)} \tag{6}$$

Eq. (5) converges to the desired value if $0<\mu<2$, which we see by defining

$$e(n)=y(n)-y^*(n) \tag{7}$$

and the parameter estimation error

$$\tilde{N}_{eff}(n) = \hat{N}_{eff}(n) - N_{eff} \quad (8)$$

Substituting into (5), we have

$$\tilde{N}_{eff}(n) = \tilde{N}_{eff}(n-1)\left[1 - \frac{\mu u^2(n-d)}{u^2(n-d)}\right] \quad (9)$$

which clearly shows the trend towards a zero parameter estimation error. In fact, if $\mu = 1$, then from Eq. (5),

$$\begin{aligned}\hat{N}_{eff}(n) &= \hat{N}_{eff}(n-1) \\ &\quad + \frac{u(n-d)}{u^2(n-d)}y(n) - \frac{u^2(n-d)}{u^2(n-d)}\hat{N}_{eff}(n-1) \quad (10)\\ &= \frac{y(n)}{u(n-d)}\end{aligned}$$

which indicates convergence in one step, as predicted by Eq. (9). Since Eq. (4) gives that there are no plant poles or zeros outside the unit disk, then we can directly assert [Goodwin84] that the adaptive controller of Eq. (5) and Eq. (6) applied to the plant Eq. (4) yields:

1. $\{y(n)\}$ and $\{u(n)\}$ are bounded sequences
2. $\lim_{n\to\infty} y(n) - y^*(n) = 0$
3. $\lim_{N\to\infty} \sum_{n=d}^{N} [y(n) - y^*(n)]^2 < \infty$

Fulton and Li [Fulton97] propose the following adaptive controller for Eq. (4):

$$\hat{N}_{eff}(n) = \frac{y(n)}{\bar{u}(n\text{-}1)} \quad (11)$$

$$u(n) = \frac{\bar{y}^*(n)}{\hat{N}_{eff}(n)} \quad (12)$$

where $\bar{u}(n-1)$ is the time average of a sequence of previous values of u. (We comment on the time averaging of $y^*(n)$ in the next section). In addition, if Eq. (11) produces an $\hat{N}_{eff}(n) < 1$, then $\hat{N}_{eff}(n)$ is replaced by 1 in Eq. (12). This bounding is not pertinent to the suitability of Eqs. (11) and (12), so is hereafter omitted.

If for a moment we ignore the time averaging of $u(n\text{-}1)$ in Eq. (11) and map Eq. (11) into something similar to Eq. (5), we have

$$\hat{N}_{eff}(n) = \frac{y(n)u(n-1)}{u(n-1)u(n-1)}$$

$$\begin{aligned}\hat{N}_{eff}(n) &= \hat{N}_{eff}(n-1) \\ &\quad + \frac{u(n-1)}{u^2(n-1)}\left(y(n) - u(n-1)\hat{N}_{eff}(n-1)\right) \quad (13)\end{aligned}$$

which is equivalent to Eq. (5) with $\mu = 1$ and d = 1. Thus the Fulton and Li controller without the averaging of $u(n-1)$ is equivalent to our one-parameter controller, Eqs. (5) and (6), where d is assumed to be 1. Without the averaging of $u(n-1)$, if the d = 1 assumption is not correct, a non-zero steady state parameter estimation error will occur, which can be seen by

$$\begin{aligned}\tilde{N}_{eff}(n) &= \tilde{N}_{eff}(n-1) \\ &\quad + \frac{u(n-1)}{u^2(n-1)}\left(u(n-d)N_{eff} - u(n-1)\hat{N}_{eff}(n-1)\right) \\ &= \tilde{N}_{eff}(n-1) + N_{eff}\frac{u(n-d)}{u(n-1)} - \hat{N}_{eff}(n-1) \quad (14)\end{aligned}$$

The averaging of $u(n-1)$ proposed by Fulton and Li should bring the parameter error closer to zero. To see this, substitute $\bar{u}(n-1)$ for $u(n-1)$ in Eq. (13). The parameter estimation error then becomes

$$\tilde{N}_{eff}(n) = \tilde{N}_{eff}(n-1) + \frac{N_{eff}\,u(n-d)}{\bar{u}(n-1)} - \hat{N}_{eff}(n-1) \quad (15)$$

Fulton and Li suggest that the time average "should be taken over the maximum expected round trip delay time of the ABR sources" [Fulton97]. In this case, $u(n-1)$ should be averaged over at least d steps. The purpose of the averaging can now be seen clearly: to make $u(n-d)/\bar{u}(n-1) \approx 1$. If the averaging of $u(n-1)$ achieves this goal, then Eq. (15) goes to zero and convergence of the UT algorithm is achieved.

Section 4 - The Two-Parameter Controller

Let us reexamine the plant model developed in Eqs. (2) and (3). Solving for $y^*(n)$, we have

$$y^*(n) = N_u(n)u^*(n-d) + C(n) = N_{eff}(n)u^*(n-d) \quad (16)$$

If we use the assumption that $N_u(n)$ and $C(n)$ are constant in the time-scale needed for parameter convergence, then Eq. (16) can be understood by examining Figure 1.

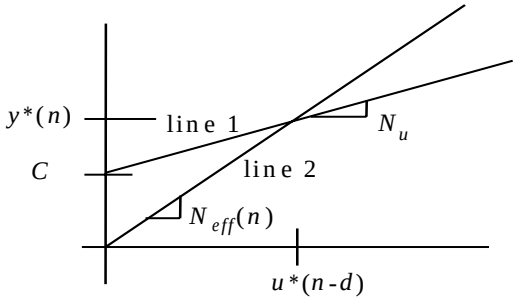

Figure 1 – A Graphical Interpretation of N_{eff}

$u^*(n-d)$ is determined by finding the horizontal coordinate of line 1 which corresponds with its vertical component $y^*(n)$. $N_{eff}(n)$ can then be determined by calculating the slope of a line extending through the origin to the point $(u^*(n-d), y^*(n))$.

Clearly, if $y^*(n)$ varies time, $N_{eff}(n)$ is not constant, making the task of the adaptive controller of Eqs. (11) and (12) (where the delay is not known and incorrectly assumed) or even Eqs. (5) and (6) (where the correct delay is used) very difficult.

Fulton and Li address this issue by performing time averages on $y^*(n)$. Li and Zhao make a good case for averaging the available bandwidth in [Zhao96]. However persuasive these arguments are, it is clear that modeling the plant with an effective number of sources N_{eff}, as in Eq. (4), requires some mechanism to reduce variability in $y^*(n)$ to produce a $N_{eff}(n)$ which varies slow enough to make tracking effective.

However, a two parameter controller may be more appropriate where $y^*(n)$, and thus $N_{eff}(n)$, is not, or should not, be constrained in its variance. In such cases, we fall back on our original plant model given by Eq. (1). The suggested corresponding controller is as follows:

$$\theta = \begin{bmatrix} N \\ K \end{bmatrix}, \quad \hat{\theta}(n) = \begin{bmatrix} \hat{N}(n) \\ \hat{K}(n) \end{bmatrix}, \quad \phi(n) = \begin{bmatrix} u(n-d) \\ P \end{bmatrix} \tag{17}$$

where K and P are constants such that $KP = C$. Then

$$\begin{aligned} y(n) &= \theta^T \phi(n), \\ \hat{y}(n) &= \hat{\theta}(n-1)^T \phi(n), \\ e(n) &= y(n) - \hat{y}(n) \end{aligned} \tag{18}$$

The parameter estimate matrix is updated as follows:

$$\hat{\theta}(n) = \hat{\theta}(n-1) + \frac{\mu\phi(n)}{\phi(n)^T \phi(n)} \left[y(n) - \hat{\theta}(n-1)^T \phi(n) \right] \tag{19}$$

and the control law is given by

$$u(n) = \frac{y^*(n+d) - P\hat{K}(n)}{\hat{N}(n)} \tag{20}$$

If $0 < \mu < 2$, the parameter estimate error $\tilde{\theta}(n) = \hat{\theta}(n) - \theta$ converges to zero, as shown by

$$\begin{aligned} \tilde{\theta}(n) &= \tilde{\theta}(n-1) + \frac{\mu\phi(n)}{\phi(n)^T \phi(n)} \left[\phi^T(n)\theta - \phi^T(n)\hat{\theta}(n-1) \right] \\ &= \tilde{\theta}(n-1) - \frac{\mu\phi(n)\phi^T(n)}{\phi(n)^T \phi(n)} \tilde{\theta}(n-1) \end{aligned} \tag{21}$$

$$\begin{aligned} \left\| \tilde{\theta}(n) \right\|^2 &= \tilde{\theta}(n)^T \tilde{\theta}(n) \\ &= \tilde{\theta}(n-1)^T \tilde{\theta}(n-1) - \frac{2\mu \left(\tilde{\theta}(n-1)^T \phi(n) \right)^2}{\phi(n)^T \phi(n)} \\ &\quad + \frac{\mu^2 \left(\tilde{\theta}(n-1)^T \phi(n) \right)^2 \phi(n)^T \phi(n)}{\left(\phi(n)^T \phi(n) \right)^2} \\ \left\| \tilde{\theta}(n) \right\|^2 &= \left\| \tilde{\theta}(n-1) \right\|^2 + \left[-2 + \mu \right] \frac{\mu e^2(n)}{\phi(n)^T \phi(n)} \end{aligned} \tag{22}$$

Since $0 < \mu < 2$ by assumption, we see that $\left\| \tilde{\theta}(n-1) \right\|^2$ is monotonically decreasing. $\tilde{\theta}(n)$ will converge to zero if the signal $y^*(n)$, and thus $u(n)$, is persistently exciting [Goodwin84], i.e. $y^*(n)$ is not constant.

One more check must be made before declaring the controller given by Eq. (19) and (20) stable. Specifically, the inverse function mapping $y(n)$ to $u(u-d)$ must be stable [Goodwin84] so that the control law produces well-behaved *u(n)*. From Eq. (1), assuming constant $C(n)$

$$u(n-d) = -\frac{C}{N_u} + \frac{y(n)}{N_u} \tag{23}$$

and since both C and N_u are finite by assumption, stability is clear and thus Eqs. (19) and (20) provide a stable controller.

As a side-note, the constant P can be optimally chosen to give the quickest convergence of Eq. (19). The optimal P is that which minimizes the eigenvalue spread of $\phi(n)\phi(n)^T$.

Section 5 - The Multi-parameter Controller

A more realistic plant than that given by Eq. (1) would have sources responding to a switch's explicit rate $\{u\}$ with varying amounts of delay. Consider now the generalized plant where

$$y(n) = B_0 u(n-d) + B_1 u(n-d-1) + \cdots + B_L u(n-d-L) + C = \theta^T \phi(n) \quad (24)$$

where

$$\theta^T = \begin{bmatrix} B_0 & B_1 & B_2 \cdots B_L & K \end{bmatrix},$$
$$\phi(n)^T = \begin{bmatrix} u(n-d) & (n-d-1) & \cdots & u(n-d-L) & P \end{bmatrix},$$
$$PK = C.$$

Thus there are B_0 sources that respond with delay d, B_1 sources that respond with delay $d+1$, etc. We can form a generalized controller very similar to the Two-Parameter Controller presented in the previous section. Let $\hat{\theta}^T(n) = \begin{bmatrix} \hat{B}_0(n) & \hat{B}_1(n) & \cdots & \hat{B}_L(n) & \hat{K}(n) \end{bmatrix}$ and reuse Eq. (19) to perform updates on the parameter estimates, copied again here

$$\hat{\theta}(n) = \hat{\theta}(n-1) + \frac{\mu \phi(n)}{\phi(n)^T \phi(n)} \left[y(n) - \hat{\theta}(n-1)^T \phi(n) \right] \quad (25)$$

The control law is then given by

$$\phi(n)^T \hat{\theta}(n) = y^*(n+d) \quad (26)$$

or equivalently

$$u(n) = \frac{1}{\hat{B}_0(n)} \left[y^*(n+d) - \hat{B}_1(n) u(n-1) - \hat{B}_2(n) u(n-2) - \cdots - \hat{B}_L(n) u(n-L) - P\hat{K}(n) \right] \quad (27)$$

Clearly Eqs. (19) and (20) are simply Eqs. (25) and (27) with $B_0 = N_u$ and $B_1 = B_2 = \cdots = B_L = \hat{B}_1(n) = \hat{B}_2(n) = \hat{B}_L(n) = 0$.

The parameter vector error power $\left\| \tilde{\theta}(n) \right\|^2$ will converge to a constant, as is shown in Eq. (22), and will further converge to zero if the signal $u(n-d)$ is persistently exciting.

However, only if the inverse mapping from $y(n)$ to $u(n-d)$ is stable can we declare this a suitable controller. This requires that $PK(n) = C(n)$ is finite (which it is by assumption) and that roots of the polynomial $B_0 + B_1 z^{-1} + \cdots B_L z^{-L}$ all lie within the unit disk $|z| < 1$. Clearly there are situations where this is not so, e.g. $B_0 = 1, B_1 = 3; B_2, B_3, \cdots B_L = 0$ has a root at $z = -3$. In such a case, the $u(n-d)$ generated from $y^*(n)$ may not be well behaved. On a related note, the algorithm requires that $B_0 \neq 0$, i.e. the minimum delay d must not be underestimated. Underestimating d also has the effect of placing a root of $B(z)$ outside the unit disk and thus produces an unstable controller. The practical consequences of these limitations are a topic of current study.

Section 6 - Summary

In this paper, Minimum Prediction Error Adaptive Controllers were adapted for use as congestion control algorithms for ATM ABR traffic. A one-parameter controller was developed in Section 3 and shown to converge when the bandwidth available for ABR traffic is constant. The Uniform Tracking (UT) algorithm was shown to be an approximation to our one-parameter controller. Section 4 introduced a mechanism for directly estimating and removing constrained source traffic. This improves convergence when the available bandwidth for ABR traffic changes in time. Section 5 generalized the controller of Section 4 to the case of sources with non-identical response delays. This generalized controller can be proven to be stable only under certain conditions. An ongoing study is evaluating the practical consequences of these limitations and the effects of violating these conditions in realistic deployments.

References

[Fulton97] C. Fulton and S. Q. Li , ``UT: ABR Feedback Control with Tracking," Proc. IEEE Infocom'97 Conference, April 1997. http://www.ece.utexas.edu/~sanqi/papers/ut_abr.ps

[Goodwin84] *Adaptive Filtering Prediction and Control*, Prentice-Hall, New Jersey, 1984.

[Hluchyj94] M. Hluchyj, et al., "Closed-Loop Rate-Based Traffic Management," AF-TM 94-0211R3, April 1994.

[Jain95] Raj Jain, Shiv Kalyanaraman, Rohit Goyal, Sonia Fahmy, and Fang Lu, "ERICA+: Extensions to the ERICA Switch Algorithm," ATM Forum/95-1346, October 1995.
http://www.cis.ohio-state.edu/~jain/atmf/af_erc22.htm

[Jain96] R. Jain, "Congestion Control and Traffic Management in ATM Networks: Recent Advances and A Survey", Computer Networks and ISDN Systems, November 1996.
http://www.cis.ohio-state.edu/~jain/papers/cnis.htm

[Rohrs96] Rohrs, Charles E., Berry, Randall A., and O'Halek, Stephen J. "Control engineer's look at ATM congestion avoidance,"
Computer Communications. v 19 n 3 Mar 1996. pp. 226-234.

[Rohrs97] Rohrs, Charles E. and Berry, Randall A.,

“A Linear Control Approach to Explicit Rate Feedback in ATM Networks,” IEEE Infocom '97 Kobe, Japan v. 3 1997 pp. 277-282

[Zhao96] Y. D. Zhao, S. Q. Li and S. Sigarto , ``A Linear Dynamic Model for Design of Stable Explicit-Rate ABR Control Schemes," ATM Forum Contribution 96-0606, April. 1996.
http://www.ece.utexas.edu/~sanqi/papers/ABR-ATM-Forum2.ps