STASTICAL-LEARNING CONTROL OF AN ABR EXPLICIT RATE ALGORITHM FOR ATM SWITCHES

Chaouki T. Abdallah*
EECE Department
The University of New Mexico
Albuquerque, NM 87131
USA
E-mail:chaouki@eece.unm.edu

Marco Ariola
Dipartimento di Informatica e Sistemistica
Universitá degli Studi di Napoli Federico II
Napoli
ITALY
E-mail: ariola@disna.dis.unina.it

Ray Byrne
Sandia National Laboratories
MS 1003, PO Box 5800
Albuquerque, NM 87185-1003
USA
E-mail:rhbyrne@sandia.gov

Abstract

This paper illustrates the application of statistical-learning control results for several different Available Bit Rate (ABR) congestion control algorithms. The ABR service category is used with Asynchronous Transfer Mode (ATM) networks to handle highly bursty and varying data applications. ATM was selected by the International Telecommunication Union (ITU) for Broadband Integrated Service Digital Network (B-ISDN), and is detailed in [1]. ATM requires the transmission of fixed size cells (each containing 53 bytes) and is a connection-based network combining the advantages of packet and circuit switching [2].

ABR traffic sources receive explicit feedback from the ATM switches and adjust their transmission rates in order to match their share of the network resources. ATM networks and specifically their ABR service control has provided a fertile area of applications for control designers as witnessed by the recent flury of papers [3, 5, 6, 7]. Most of these papers however have made simplifying assumptions regarding the model of the network or its connectivity. In this paper, we use the nonlinear model and controller provided in [3] and show considerable improvements to the control

* **Corresponding Author:** The research of C.T. Abdallah is partially supported by NSF INT-9818312

algorithm using statistical learning methods. The model of [3] is nonlinear although the control designs used are based on linearized methods. The author in [3] described how the new generation of ATM switches provide per virtual connection (VC) queuing and scheduling, resulting in fair sharing of the link bandwidth. This then frees the controller to concentrate on the congestion control problem for each VC. The designs in [3, 4] concentrate on designing linear controller for the linearized version of the buffer dynamics. This however will only provide local design and analysis tools. In fact, we show for example that the Linear Quadratic Regulator (LQR) designs in [3] do not stabilize the nonlinear system in the case of small buffer size.

From [3], the local closed-loop dynamics of a VC are described by equation (1) where R denotes the explicit rate (ER) computed by a switch for a given VC and Q denotes the buffer occupancy of that VC at the switch.

$$\begin{aligned} Q(n+1) &= Sat_B \left\{ Q(n) + \sum_{i=0}^{d} l_i(n) R(n+1-i) - \mu(n) \right\} \\ R(n+1) &= Sat_C \left\{ R(n) - \sum_{j=0}^{J} \alpha_j (Q(n-j) - Q^0) - \sum_{k=0}^{K} \beta_k R(n-k) \right\} \end{aligned} \tag{1}$$

where $\lambda(n)$ is the rate at the ABR switch, $\mu(n)$ is the service rate (considered an external input) at the switch during the interval $[n, n+1]$, l_i are unknown plant parameters, d is the roundtrip delay between switch, source and back to switch (see [5] for a discussion of the importance of knowing d exactly), B is the buffer size, C is the maximum ER, and Q^0 is the desired buffer occupancy. The number of controller parameters J and K along with the parameters themselves α_j and β_k are to be found such that:

1. The closed-loop system is stable, so that the buffer occupancy converges to Q^0
2. The steady-state behavior is acceptable, i.e. guarantee fairness in the rate allocation
3. Overshoot in the buffer is minimized in order to minimize cell loss
4. Undershoot in the buffer is minimized in order to maximize resource utilization.

The model in (1) does not account for the disturbances due to uncontrolled information flow, nor does it account for the fact that the delay d may be unknown. Such issues were considered in [5] and may be handled in our statistical-learning methodology. Note that the Q equation in (1) describes the plant (Buffer) dynamics, while the R equation describes the controller structure and that

$$Sat_a(x) = \begin{cases} 0 & \text{if } x < 0, \\ a & \text{if } x > a, \\ x & \text{otherwise} \end{cases} \tag{2}$$

The control design problem consists of finding the α's and β's in order to drive Q to the desired Q^0. The controller part of equation (1) can be rewritten as

$$R(n+1) = Sat_C\{R(n) - GY(n)\} \tag{3}$$

It was shown in [4] that it is sufficient (at least in the case of linearized model) to take $J = 1$ and $K = d$ in order to completely place the poles of the closed-loop system. In other words, such controller need only have 2 feedforward gains α_0, α_1 and $d+1$ feedback gains $\beta_k; k = 0, \cdots d$. In that case, let the gain vector be:

$$G = [\alpha_0, \alpha_1, \beta_1, \beta_2, \cdots, \beta_d] \tag{4}$$

$$Y(n) = \Big[Q(n) - Q^0, Q(n-1) - Q^0, R(n), R(n-1), \cdots, R(n-d)\Big]' \tag{5}$$

The approach in [3] consists of removing the saturation functions to linearize the system, so that the dynamics may be expressed as the following state-space system

$$\begin{aligned} Y(n+1) &= AY(n) + B_1 u(n) + B_2 \mu(n) \\ u(n) &= -GY(n) = -\sum_{j=0}^{1} \alpha_j [Q(n-j) - Q^0] - \sum_{k=0}^{d} \beta_k R(n-k) \end{aligned} \tag{6}$$

so that

$$Y(n+1) = A_1 Y(n) + B_2 \mu(n) \tag{7}$$

where

$$A_1 = \begin{pmatrix} 1 & 0 & \cdots & \cdots & 0 & 1 & 0 \\ 1 & 0 & \cdots & \cdots & 0 & 0 & 0 \\ -\alpha_0 & -\alpha_1 & 1-\beta_0 & -\beta_1 & -\beta_2 & \cdots & -\beta_d \\ & & 1 & & & & \\ & & & 1 & & & \\ & & & & \ddots & & \vdots \\ & & & & & 1 & 0 \end{pmatrix} \quad B_1 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}; \; B_2 = \begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix} \tag{8}$$

In order to write the equations in a more traditional control setting, we let

$$\begin{aligned} Q(n+1) - Q^0 &= Q(n) - Q^0 + R(n+1-d) - \mu(n) \\ Q(n) - Q^0 &= Q(n-1) - Q^0 + R(n-d) - \mu(n-1) \end{aligned}$$

and assuming $\mu(n)$ to be a constant μ we obtain,

$$X(n+1) = AX(n) + Bu(n) \tag{9}$$

where the new state vector is given by

$$X(n) = \begin{pmatrix} Q(n) - Q^0 \\ Q(n-1) - Q^0 \\ R(n) - R(n-d) \\ R(n-1) - R(n-d) \\ \vdots \\ R(n-d+1) - R(n-d) \end{pmatrix} \tag{10}$$

and the system dynamics are described by

$$A = \begin{pmatrix} 2 & -1 & 0 & & \cdots & 0 & 1 \\ 1 & 0 & & & \cdots & & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 & -1 \\ & & 1 & & & & -1 \\ & & & 1 & & & -1 \\ & & & & \ddots & & \vdots \\ & & & & & 1 & -1 \end{pmatrix}, B = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \tag{11}$$

A stabilizing controller designed to drive $X(n)$ to zero, is then given by

$$\begin{aligned} u(n) &= R(n+1) - R(n) \\ &= -KX(n) \\ &= -\alpha_0[Q(n) - Q^0] - \alpha_1[Q(n-1) - Q^0] - \sum_{k=0}^{d-1} \beta_k [R(n-k) - R(n-d)] \end{aligned} \tag{12}$$

where we have used the constraint that $\beta_d = \sum_{k=0}^{d-1} \beta_k$ and where

$$\begin{aligned} K &= [k_1 \;\; k_2 \;\; k_3 \;\; \cdots \;\; k_{d+2}] \\ &= [\alpha_0 \;\; \alpha_1 \;\; \beta_0 \;\; \cdots \;\; \beta_{d-1}] \end{aligned} \tag{13}$$

In [3], the linear system model was used to design an LQR controller which was then implemented on the full nonlinear system in a simulation environment. It turns out however that such linearized approach will result in LQR gains which are highly dependent on the buffer size. As an example, and in order to make the gains obtained in [3] stable with the nonlinear system, we needed to increase the buffer size Q^0 from 10 to 100.

In this paper, we use a different track by designing the controller directly for the nonlinear model (1) using statistical-learning control. Note that the saturation nonlinearities pose no additional problems for such design methodology which was first developed in [?] and extended by one of the authors and his colleagues [9]. The methodology is able to provide a stabilizing controller which also satisfies other performance objectives for the nonlinear system (1). The control design technique relies on the following algorithm [9]:

Algorithm 1 *Given:*

- *Sets $\mathcal{F}$ of plant parameters ($l_i, i = 0, \;\cdots d$) and $\mathcal{G}$ (the $\alpha_j, j = 0, 1;\; \beta_k,\; k = 0, \cdots K$) of controller parameters,*
- *Probability measures P on $\mathcal{F}$ and Q on $\mathcal{G}$,*
- *A measurable function $f : \mathcal{F} \times \mathcal{G} \longrightarrow [0, 1]$, and*
- *An accuracy parameter $\varepsilon \in (0, 1)$, a level parameter $\alpha \in (0, 1)$, and a confidence parameter $\delta \in (0, 1)$.*

Let $J(F, G)$ be the performance index to be minimized and $R_P(J(\cdot)) = \mathbb{E}_P[J(F, \cdot)]$ and

$$R_{P_n}(\cdot) = \frac{1}{n} \sum_{j=1}^{n} J(F_j, \cdot)$$

Then,

1. *Choose m controller parameters G_i with random entries according to the distribution Q where*

$$m \geq \frac{\log(2/\delta)}{\log[1/(1-\alpha)]}$$

2. *Choose n plant parameters F_j with random entries according to the distribution P, where*

$$n = \left\lfloor \frac{20}{\varepsilon^2} \log\left(\frac{8}{\delta}\right) \right\rfloor + 1$$

3. *Evaluate the stopping variable*

$$\gamma = \max_{1 \le i \le m} \left| \frac{1}{n} \sum_{j=1}^{n} r_j J(F_j, G_i) \right|$$

where r_j are Rademacher random variables, i.e. independent identically distributed random variables (also independent of the plant sample) taking values $+1$ and -1 with probability $1/2$ each. If $\gamma > \frac{\varepsilon}{K_1}$, add n more independent plants with parameters having distribution P to the plant samples, set $n := 2n$ and repeat step 3

4. *Choose the controller which minimizes the cost function R_{P_n}.*

This algorithm will allow us to find the controller parameters α_i, β_j needed to stabilize the closed-loop system and minimize the overshoot, undershoot, etc. for the nonlinear system (1). We will provide simulation results illustrating our design and comparing its performance with the more traditional LQR design.

References

[1] "ATM Forum Traffic Management Specification," Version 4.0, ATM Forum Traffic Management Working Group AF-TM-0056.000, April 1996.

[2] M. De Prycker, *Asynchronous Transfer Mode: Solution for Broadband ISDN,* Prentice-Hall, Englewood Cliffs, NJ, 1995.

[3] L. Benmohamed, and Y.T. Wang, "A Control-Theoretic ABR Explicit Rate Algorithm for ATM Switches with Per-VC Queuing", in *Proc. Infocom98*, San Francisco, CA, 1998, pp. 183–191.

[4] L. Benmohamed, "Control-Theoretic Applications in Congestion Control", NSF/ONR Workshop on Future Directions in Systems and Control: Research in Communication Networks, November 6-7, 1998, available at: http://www.ge.uiuc.edu/nsfonr.html

[5] F. Blanchini, R. Lo Cigno, and R. Tempo, "Control of ATM Networks: Fragility and Robustness Issues" in *Proc. American Control Conference*, Philadelphia, PA, 1998, pp. 2847–2851.

[6] Special Sessions TA-03, TM-03, TP-03, in *Proc. IEEE Conference on Decision and Control*, San Diego, CA, 1997.

[7] Special Session, *ATM ABR Services I-Control theory Application and Flow Control*, in *Proc. Infocom97*, Kobe, Japan, 1997, pp. 277–302.

[8] M. Vidyasagar, *A Theory of Learning and Generalization with Applications to Neural Networks and Control Systems,* Springer-Verlag, Berlin 1996.

[9] V. Koltchinskii, C.T. Abdallah, M. Ariola, P. Dorato, and D. Panchenko, "Improved Sample Complexity Estimates for Statistical Learning Control of Uncertain Systems" Accepted, *IEEE Transactions on Automatic Control*, 1999. Also, UNM Tech Report, EECE 2000-001, "Statistical Learning Control of Uncertain Systems: It is better than it seems"