Adaptive Congestion Control in ATM Networks

Farzad Habibipour, Mehdi Galily, Masoum Fardis, Ali Yazdian

Iran Telecommunication Research Center, Ministry of ICT, Tehran, IRAN
habibipor@itrc.ac.ir

Abstract. In this paper an adaptive minimum variance controller is proposed to minimize the rate of stochastic inputs from uncontrollable high priority sources. This method avoids the computations needed for pole placement design of the minimum variance controller, and utilizes an online recursive least squares algorithm in direct tuning of the controller parameters.

1 Introduction

Congestion control of ATM (Asynchronous Transfer Mode) network with its wide use in high bandwidth communication systems is the source of attention and subject of active research [1]. There are different types of communication services which are categorized in high priority sources including Constant Bit Rate (CBR) and Variable Bit Rate (VBR), and best effort sources often considered as Available Bit Rate (ABR) sources. On the basis of QoS (Quality of Service) requirements, congestion control is possible by regulating the queue length at bottleneck nodes via active controlling of Available Bit Rate (ABR) [2]. Another important factor is the unavoidable delay of closed loop systems in high speed links such as satellite ATM networks or IP ATM. The Round Trip Time (RTT) delay is the time from the moment control information is sent to the source until an appropriate action takes place, and is the source of instability in simple control systems [3]. In this paper, a direct minimum variance self tuning regulator is proposed to be used with an online recursive least squares algorithm to estimate the appropriate control parameters and to adaptively regulate the queue length to the nominal value. The simulation results show the efficiency of the method in comparison to a proportional-integral rate matching controller.

2 Queue Length Dynamics

Each bottleneck node of an ATM network has an output buffer to prevent cell loss, but the queue length of cells in the limited size buffer should be controlled to avoid overflow. Denoting the queue length at time n by $q(n)$, the queue length dynamics is written by a simple linear equation

$$q(n+1) = q(n) + r(n) - \mu(n) \tag{1}$$

where $r(n)$ is the total number of cells receiving in the time interval $[n, n+1)$, and $\mu(n)$ is the number of cells that depart from this node at the same time. The rate of input cells to the buffer, $r(n)$, consists of inputs from *M* controllable ABR sources and a rate of cells from uncontrollable high priority sources (CBRs and VBRs) denoted by $r^u(n)$. Clearly:

$$r(n) = \sum_{m=1}^{M} r_m^c(n) + r^u(n) \tag{2}$$

A high performance tracking control method, actually results in optimal use of buffer and network capacity. $q(n)$ is referred to as the controlled variable and $r_m^c(n)$s are the *M* control signals. The available bandwidth for ABR, $\mu(n) - r^u(n)$, is a stochastic value since the rate of VBR traffic is time varying. Therefore the uncontrolled traffic, $r^u(n)$, can be simply modeled by a filtered random disturbance sequence to the system.

There are noticeable round trip time delays in a congestion controlled feedback loop:

$$r_m^c(n) = u_m(n - d_m) \tag{3}$$

where $u_m(n)$ is the available bit rate to the *m*th source calculated at time *n*, but is considered by the source d_m time units later. We suppose minimum and maximum limits for these time delays:

$$0 \le d_{\min} \le d_1 \le d_2 \le \cdots \le d_M \le d_{\max} \tag{4}$$

By defining a nominal queue length value (Q), and the error variable $q(n) - Q$, a simple proportional integral control law can be used

$$u_m(n) = a_m\left[u_m(n-1) + k_1 q(n) + k_2 q(n-1) - (k_1 + k_2)Q\right] \tag{5}$$

where a_m is the rate allocation coefficient for source *m*, and k_1 and k_2 are control parameters which are constant for all of the sources. Typically

$$\sum_{m=1}^{M} a_m = 1 \tag{6}$$

The control signals of the different sources are computed by dividing a unified control signal proportional to the rate allocation coefficients:

$$r_m^c(n + d_m) = u_m(n) = a_m u(n) \tag{7}$$

To design the pole placement controller, the queue length dynamics are reformulated in frequency domain (Z-domain). A colored noise process is first assumed for the rate of uncontrolled sources:

$$r^u(n) = C(z)e(n) \tag{8}$$

Where $e(n)$ denotes a Gaussian random sequence. By definition of $y(n) = q(n) - Q$, the tracking problem is simplified to the regulation problem, and the dynamical model is described by

$$A(z)y(n) = B(z)u(n) + C(z)e(n) \tag{9}$$

in which

$$A(z) = z^{d_{\max}+1} + z^{d_{\max}} \quad ; \quad \deg(A(z)) = d_{\max} + 1 \tag{10}$$

and

$$B(z) = a_d z^d + a_{d-1} z^{d-1} + \ldots + a_0 \quad ; \quad \deg(B(z)) = d = d_{\max} - d_{\min} \tag{11}$$

3 Minimum Variance Controller

The minimum variance control law is designed to minimize the cost function defined as the expectation of the controlled signal in equation 9:

$$J = E\{y^2(n)\} \tag{12}$$

Equation (9) is then reconfigured as

$$y(n + d_0) = \frac{B(z)}{A(z)} u(n + d_0) + \frac{C(z)}{A(z)} e(n + d_0) \tag{13}$$

where $d_0 = d_{\min}$ is the minimum time delay for a control action to appear in output, and hence is the prediction horizon of the minimum variance controller. Equation (13) can be further modified to yield

$$y(n + d_0) = \frac{B(z)}{A(z)} u(n + d_0) + F(z)e(n+1) + \frac{zG(z)}{A(z)} e(n) \tag{14}$$

$F(z)$ and $G(z)$ are computed as the quotient and remainder polynomials of dividing $z^{d_0-1}C(z)$ to $A(z)$ from the following Diophantine equation:

$$z^{d_0-1}C(z)=A(z)F(z)+G(z) \tag{15}$$

By a few mathematical manipulations through the noise innovation model, the following equation is obtained [4]:

$$y(n+d_0)=F(z)e(n+1)+\frac{zB(z)F(z)}{C(z)}u(n)+\frac{zG(z)}{C(z)}y(n) \tag{16}$$

The second part of which is considered as the prediction model

$$\hat{y}(n+d_0|n)=\frac{zB(z)F(z)}{C(z)}u(n)+\frac{zG(z)}{C(z)}y(n) \tag{17}$$

And to minimize the prediction error, $y(n+d_0)-\hat{y}(n+d_0|n)$, the minimum variance control law is obtained

$$u(n)=-\frac{G(z)}{B(z)F(z)}y(n) \tag{18}$$

4 Self Tuning Regulator

The pole placement design of the minimum variance controller via equations (15) and (18) is just applicable if the polynomials of the model in equation (9), i.e. $A(z)$, $B(z)$, and $C(z)$, are definite; but this is not the case in real situation. So there is a need to utilize an estimation method either for these parameters or directly for the control parameters in equation (18). Using an identification method to estimate the parameters of the model in equation (9) is followed by the hard computation of the Diophantine equation and is not efficient. Another approach is the direct tuning of the controller parameters. To start, equation (16) is parameterized in backward difference form as follow

$$y(n+d_0)=\frac{1}{C^*(z^{-1})}\left(R^*(z^{-1})u(n)+S^*(z^{-1})y(n)\right)+R_1^*(z^{-1})e(n+d_0) \tag{19}$$

in which $R_1^*(z^{-1})=F^*(z^{-1})$. Recursive Least Squares (RLS) algorithm is proposed to estimate the polynomials $R^*(z^{-1})$ and $S^*(z^{-1})$ as the coefficients of the regressors of input ($u(n)$) and output ($y(n)$). The $\frac{1}{C^*\left(z^{-1}\right)}$ coefficient can be considered as a filter on regressors, and is commonly replaced by a stable filter of the rational form $\frac{Q^*\left(z^{-1}\right)}{P^*\left(z^{-1}\right)}$:

$$u_f(n)=\frac{Q^*\left(z^{-1}\right)}{P^*\left(z^{-1}\right)}u(n) \quad and \quad y_f(n)=\frac{Q^*\left(z^{-1}\right)}{P^*\left(z^{-1}\right)}y(n) \tag{20}$$

Therefore the RLS algorithm is formulated to estimate the coefficients of $R^*\left(z^{-1}\right)$ and $S^*\left(z^{-1}\right)$ in the following model

$$y(n+d_0)=R^*\left(z^{-1}\right)u(n)+S^*\left(z^{-1}\right)y(n)+\varepsilon(n+d_0) \tag{21}$$

where

$$\begin{aligned} R^*\left(z^{-1}\right)&=r_0+r_1z^{-1}+\ldots+r_kz^{-k}\\ S^*\left(z^{-1}\right)&=s_0+s_1z^{-1}+\ldots+s_lz^{-l}\end{aligned} \tag{22}$$

The recursive least squares estimation is performed via

$$\begin{aligned}&\varepsilon(n)=y(n)-R^*\left(z^{-1}\right)u_f(n-d_0)-S^*\left(z^{-1}\right)y_f(n-d_0)=y(n)-\phi^T(n-d_0)\hat{\theta}(n-1)\\ &\phi^T(n)=[u(n)\ \ \ldots\ \ u(n-k)\ \ y(n)\ \ \ldots\ \ y(n-l)]\\ &\theta^T=[r_0\ \ \ldots\ \ r_k\ \ s_0\ \ \ldots\ \ s_l]\end{aligned} \tag{23}$$

5 Simulation Results

Three ABR sources with different round trip time delays are assumed, one of which has an allocation rate coefficient of 0.5 and the others have equal coefficients of 0.25. The output service rate of the node is 10000 cells per time unit and the traffic of high priority sources is modeled as a filtered random process with a Gaussian input sequence (m_x=5000, σ_x=2500). Nominal time delays of ABR sources are $d_1=3$, $d_2=4$, $d_3=5$; $M=3$, and the desired queue length is 3000. The nominal queue length is 3000 and the maximum buffer size is 5000. Simulation results of the proposed controller are compared to the simple control structure of equation (4). Fig. 1 presents a comparison of the queue length values for the proportional integral control method, and the adaptive minimum variance controller. Both methods have regulated the queue length to 3000, but their mean values and standard deviations are different. Obviously, the minimum variance controller has resulted in lower variance of the queue length about the nominal value. Figs. 2 and 3 depict the robustness of the system when one of the ABR sources is failed.

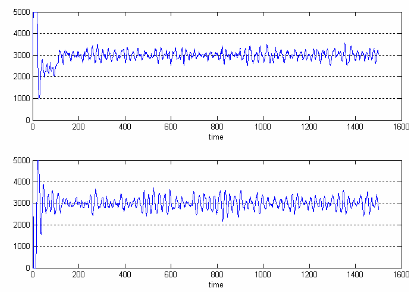

Fig. 1. Tracking control of queue length, Upper: Control feedback loop, Lower: The self tuning minimum variance regulator

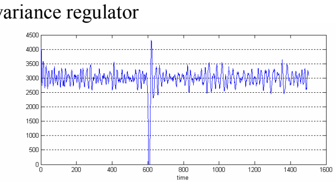

Fig. 2. Queue length when a failure is occurred to ABR source 2 at t=600

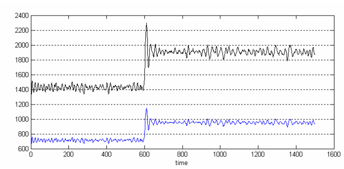

Fig. 3. Bit rate available to ABR sources

6 Conclusions

The self tuning minimum variance regulator proposed in this article, is designed to minimize the effect of stochastic disturbance inputs of the high priority sources to the system. While the queue length dynamics at bottleneck nodes is undetermined and the round trip time delays are uncertain and time varying for controlled ABR sources, an online recursive least squares algorithm can directly tune the control parameters to achieve the desired performance. The proposed controller is automatic and just needs good estimations of the minimum and maximum limits of the time delays. This adaptive system is also robust to the changes in network conditions, and the failure of ABR sources, to prevent buffer overflow and efficient use of network resources.

Reference:

1. Imer O.C., Compance S., Basar T., Srikant R.: Available bit rate congestion control in ATM networks, IEEE Control Systems Magazine, Vol. 135, (2001) 38-56
2. Altman E., Basar T., Srikant R.: Congestion control as a stochastic control problem with action delays, Automatica, Vol. 35, (1999) 1937-1950
3. Liansheng Tan, Yang S.H.: Rate-based congestion controllers for high speed computer networks, IFAC 15th Triennial World Congress, (2002) Barcelona, Spain
4. Astrom K.J., Wittenmark B.: Adaptive Control, Addison-Wesley, 2nd Edition, (1995)