Reinforcement Learning-Based Neural Network Congestion Controller for ATM Networks

Ahmed A. Tarraf, Ibrahim W. Habib, and Tarek N. Saadawi
Electrical Engineering Department
The City College of New York
New York, N.Y., 10031
e-mail: tarraf@fuwutai.att.com

Abstract This paper presents a new approach to the problem of congestion control arising at the User-to-Network Interface (UNI) of the ATM-based Broadband Integrated Services Digital Networks (B-ISDN). Our approach employs an adaptive rate-based feedback control algorithm using reinforcement learning Neural Networks (NNs). The reinforcement learning NN controller provides an adaptive optimal control solution. This is achieved via the formulation of a performance measure function (cost function) that is used to, adaptively, tune the weights of the NN. The cost function is defined in terms of two main objectives: 1) to minimize the Cell Loss Rate (CLR), i.e., control congestion and 2) to preserve the quality of the voice/video traffic via maintaining the original coding rate of the multimedia sources. The results show that the NN control system is adaptive in the sense that it is applicable to any type of multimedia traffic. Also, the control signal is optimal in the sense that it maximizes the performance of the system which is defined in terms of its performance measure function. Hence, our novel approach is very effective in controlling congestion of the multimedia traffic in ATM networks.

I. Introduction

Broadband Integrated Services Digital Networks (B-ISDN) are designed to support the integration of both non real-time traffic (e.g., data file transfer) and real-time traffic (e.g., voice and video interactive services) over the same medium. The Asynchronous Transfer Mode (ATM) solution has been recommended by the ITU [1] to be the transport method of the B-ISDN networks. The ATM solution offers enough flexibility to statistically multiplex different types of multimedia traffic with different quality of services (QOS) (e.g., cell loss rate (CLR), delay, and delay variability) requirements and a broad range of statistical characteristics.

One of the problems that makes congestion control in ATM networks difficult is the uncertainties and the highly time-varying nature of the different traffic patterns that require different quality of service (QOS) parameters from the network. Furthermore, due to the very small cell transmission time and the small buffers sizes, it is imperative that any effective congestion control algorithm must be simple with minimal reaction time and highly adaptive to the different QOS requirements. Hence, not only must the congestion control algorithm be a preventive-type one, but also performance-driven.

Most of traditional control strategies are based upon the results obtained from analytical performance evaluation methods such as queuing models. A major shortcoming of the currently available queuing models [2],[3] is that only steady-state results are tractable and consequently, any control algorithm tailored on the basis of such models can ensure optimal performance only under steady-state conditions. However, performance control methods that dynamically regulate traffic flows according to changing network conditions require understanding of its dynamics, which is almost impossible in ATM networks. Furthermore, in these models, hypotheses concerning the detailed knowledge of systems under study can seldom be justified in real-life.

This paper overcomes the shortcomings of the existing methods through a novel approach. It adopts an adaptive congestion control model using NNs that can meet the foregoing requirements of congestion control. Our motivation to employ NNs is to apply their powerful adaptive and learning capabilities [4],[5] to congestion control problems for which conventional approaches have proven ineffective or constructive procedures do not exist for adaptive control. Furthermore, a detailed knowledge of the mathematical relationships that exist among the many involved variables is not required.

II. Rate-based Feedback Control Using NNs

A block diagram of the system under study is shown in figure (1). The NN controller uses the number of cells in the multiplexer buffer as a measure of potential congestion problems. It, then, generates an output

0-7803-2489-7/95 $4.00 © 1995 IEEE

optimal control signal which is fed-back to the input traffic sources in order to change the cells' arrival rate. During the periods of buffer overload, the control signal throttles the arrival rate via decreasing the source coding rate. For example, for ADPCM variable bit rate (VBR) voice sources the coding rate could be decreased from 4 bits/sample to 3 bits/sample. Whereas during periods of buffer underload the coding rate is returned back to its previous level. Similarly, the same principle can be applied to video traffic sources, see [6] for a detailed mathematical analysis for our proposed rate-based control algorithm.

The NN controller generates the feedback control signal which is optimal in the sense that it maximizes the system performance through a given cost function which combines two important system performance measures:

1) The input multiplexer buffer overflow (which reflects the CLR).

2) The level of the coding rate of the input source(s) (which reflects the quality of the voice/video traffic).

The advantages of the proposed algorithm are several: 1) It can be classified as a preventive type congestion control mechanism since the algorithm is applied at the input access node of the network, and its speed is not limited by the propagation delay. Hence, any control action will be in time to avoid the potential congestion. 2) The statistical multiplexing gain is enhanced, since more sources can be supported for each multiplexer.

The optimal control nature of the proposed algorithm introduces the following advantages: 1) The proposed algorithm optimizes the system behavior not only by minimizing the CLR but also by maximizing the level of the coding rate. 2) Since the performance index contains two performance measures (the CLR and the level of the coding rate), one can weigh the relative importance of these two performance measures.

In this application, an optimal control low is a one that minimizes the performance index of the system by minimizing the first performance measure (buffer overflow) and maximizing the second performance measure (the level of the coding rate). The importance of minimizing the buffer overflow is to minimize the CLR of the accepted calls during the call progress phase. On the other hand, the importance of maximizing the level of the coding rate is to increase the quality of the coding information (voice or video), reflected by a higher SNR.

III. Reinforcement Learning Method

It clear now that the congestion control problem can be formulated as an optimization problem. It is very difficult for the classical optimal control methods to solve this problem because these methods rely upon a very accurate mathematical model for the system to be controlled. In our case, these methods need a very accurate model that can characterize the arrival process with the distribution of the number of the cells in the input multiplexer buffer. Since this model does not exist,furthermore if it exists it should be adaptive to the changing time-dynamics of the arrival process and would involve very long computation time, hence it will be infeasible for real-time implementation. These limitations of using the classical optimal control methods in our application have motivated us to use reinforcement learning NN method to solve the optimal control problem in our application. The reinforcement learning method is based upon the improvement of the performance of the system in terms of the defined performance index and can rely upon a very simple model for the system to be controlled.

The reinforcement learning method evaluates the performance of the system in terms of a defined performance index (cost function), consequently generating an evaluation signal. This signal is function in the deviation of the system performance from the optimal one. This evaluation signal is used to adjust the weights of a NN controller such that the produced feedback control signal, when applied to throttle the coding rates of the input sources, results in maximization of the system performance (minimization of the performance index). Hence the reinforcement leaning NN method depends mainly on the defined performance index of the system and always tends to minimize that index even though it might use very simple mathematical model for the system.

The proposed congestion control algorithm consists of a critic part and a NN controller part as shown in figure (2). The system to be controlled (the environment) is composed of the input traffic sources and the input multiplexer buffer at the access node to the network (UNI). The cells' arrival process to the input multiplexer buffer depends upon the coding rate of the sources while the departure rate of the cells form the input multiplexer buffer depends upon the link capacity (in Mbps). In this application, the link capacity is constant, hence, the number of the cells in the multiplexer buffer depends upon the arrival process. The inputs to the control algorithm are taped delay-values of the number of the cells in the multiplexer buffer and taped delay values of the feedback control signal. The controller output is the feedback control signal which is fed to the input sources to alter their coding rates. The critic part involves the performance index of the system (cost function) to be minimized. According to this cost function, the critic part evaluates the system performance and generates an evaluation signal which is function of the deviation of the system performance from the desired optimal one and is used to change the weights of the of the NN controller.

We defined the cost function (J) as:

$$J(P)=\sum_{k=1}^{L} R_n S_n(k+1)\,(n_d(k+1)-n(k+1))^2+R_u(u_d(k+1)-u(k+1))^2 \quad (1)$$

where

P: is the trial number

L: is the length of the optimization interval.

n(k+1): is the number of cells in the buffer at the sample k+1

$n_d(k+1)$: is the congestion threshold level within the multiplexer buffer

$$n_d(k+1) \le n_{\max} \quad (2)$$

n_{max}: is the maximum length of the input multiplexer buffer

$S_n(k+1)$: is the given by

$S_n(k+1)=1$ if $n(k+1) \ge n_d(k+1)$

$S_n(k+1)=0$ if $n(k+1) < n_d(k+1)$

R_n: is the weight value on the buffer overflow performance measure

u(k+1): is the feedback control signal at the sample k+1

$u_d(k+1)$: is the desired value of the feedback control signal, it is also the maximum value of the feedback control signal:

$$u(k+1) \le u_d(k+1) \quad (3)$$

R_u: is the weight value the on the level of the coding rate performance measure. This term reflects the weight on achieving a certain voice/video quality.

The controlled source coding rate is defined by the equation:

$$C(k)=C_0 u(k) \quad (4)$$

where C_o is the maximum uncontrolled coding rate of the source, and C(k) is the controlled coding rate at sample k, where u(k) is the feedback control signal produced by the controller at sample k. ($u(k) \le 1$). Since $C(k) \le C_o$, therefor, the maximum value of the control signal is unity.

The NN of the controller has three layers with no inner feedback loop and no direct connection from the input layer to the output layer as shown in figure (3). Both the hidden and the output layers have a sigmoid function (f) [7] to provide the nonlinear mapping capability. The NN output is the environment input, u(k). This NN is expressed by the following equation:

$$u(k)=f\{w^T(P)\, f[W(P)\, I(k)]\} \quad (5)$$

where **w** is the weight vector from the hidden layer to the output layer, **W** is the weight matrix from the input layer to the hidden layer. I is the input vector to the controller, expressed as follows:

$$I^T(k)=[n)k),n(k-1),u(k-1),u(k-2)] \quad (6)$$

Both the weight vector, **w**, and the weight matrix, **W**, are tuned so as to minimize the cost function, J, which is defined by (1). There are many methods to tune the weights of the NN such as steepest descent method, Newton's method, quasi Newton method, and conjugate gradient method [8]. Because the steepest descent method has the advantages of being simple and highly parallel, we used it to tune the weights of the NN as following:

$$w(P+1)=w(p)-\eta\frac{\partial J(P)}{\partial w(P)} \quad (7)$$

$$W(P+1)=W(P)-\eta\frac{\partial J(P)}{\partial W(P)} \quad (8)$$

where η is the parameter determining the convergence speed of the weights tuning. The work reported in [7] and [8] has proved the conversion of the cost function using the weight tuning algorithm given in equations (7), (8).

IV. Simulation

In simulation of the controller, the voice sources and video sources are simulated on SunSparc work-station using C language. A voice source is simulated using the ON/OFF model [9]. In this model, the ON and OFF time periods are assumed to be exponentially distributed random variable with mean $1/\beta$ and $1/\alpha$ respectively. During the On period a fixed number of packets is generated, each having duration of T msec. A video source is simulated using the first order autoregressive (AR) Markov process X(n) that takes into consideration the autocorrelation of the sequence. The definition of the AR process is as follows [10]:

$$X(n)=aX(n-1)+bG(n) \quad (9)$$

where X(n) is the bit-rate during the nth frame, G(n) is a sequence of independent Gaussian random variables, *a* and *b* are constants. The coding rate of the voice/video sources will be reduced in steps (corresponding to number of bit per sample). The controlled source coding rate will take the following values according to the feedback control signal:

$$C(k)=C_0,\ C(k)=0.75C_0,\ C(k)=0.5C_0$$

These three values of the coding rate correspond to three values of the feedback control signal:

$u(k)=1$, $u(k)=0.75$. and $u(k)=0.5$, respectively.

Hence, the control signal produced by the controller is quantized to those three values before applied to the voice/ video sources. The number of the cells in the buffer is measured at every sampling period (T_s), and also the control signal is applied at every T_s. At given sampling instant (k) the controller measures the number of the cells in the buffer, and also apply the feedback control signal.

V. Results

In this section we show some of the experiments using different values of the control parameters. In these experiments, the voice sources were simulated using the ON/OFF model with $1/\beta = 0.350$ sec and $1/\alpha = 0.650$ sec. The video sources were simulated using the first order autoregressive process mentioned above. The parameter used in this model are: $a=0.8781$, $b=0.1108$, the mean of $G(n)$ is 4.3 Mbps, and the variance of $G(n)$ is unity.

Experiment (1):

In this experiment, we used a finite buffer length of 50, to limit the delay to 125 μsec. 150 voice sources were involved with the link capacity of 1.53 Mb/sec (about 110% utilization). This experiment demonstrates the effect of the parameters R_n, and R_u on the performance of the controller.The control parameters were:

$n_d=30$, $\zeta=0.2$, $T_s=0.01$sec. $L=5T_s$. The ratio R_n/R_u were changed from 0.01 to 0.1. Figure (3) shows the CLR and the voice quality versus R_n/R_u. As shown, as the weighting parameter of the CLR-part of the cost function (R_n) increases, the CLR gets better (decreases) and the voice quality decreases. This justifies the trade-off between decreasing the CLR and increasing the voice quality. However, one might increase the weight of either one to achieve certain required performance, hence the proposed controller allows the network operator to choose between satisfaction of the required CLR or mean bits/sample.

The following experiments were performed to test the proposed algorithm with the video arrival process. 16 video sources were statistically multiplexed at the input multiplexer buffer. We used a finite buffer of length of 20 cells with a link capacity of 62.4 Mbps to limit the delay to 125 μsec.

Experiment (2):

In this experiment, the control parameters were:

$R_n=0.2$, $R_u=10$, $n_d=15$, $\zeta=0.5$, $T_s=0.001$sec. $L=10T_s$

Figure (4) shows the histogram of the number of the video cells in the buffer under the control of the proposed algorithm compared with the histogram of that number for the uncontrolled case. The controller has reduced the CLR to 10^{-6} while maintaining the video sources operate at the maximum coding rate, most of the experiment time. The corresponding video quality was 0.889 from its maximum value (i.e., minimum effect on the quality of video-picture).

Experiment (3):

In this experiment, we show the effect of the parameter R_n in determining the value of the obtained CLR. The control parameters were:

$R_n=0.3$, $R_u=10$, $n_d=15$, $\zeta=0.5$, $T_s=0.001$sec. $L=10T_s$

The value of R_n was increased in this experiment compared with the experiment (2) ($R_n=0.2$). Figure (5) shows the histogram of the number of the cells in the buffer where the corresponding CLR was 10^{-7}. Figure (6) shows the histogram of the feedback control signal, from which it is clear that the control-action has maintained the video sources operating at 0.75 of their maximum coding rate most of the time, and the corresponding video quality was 0.85 of the maximum value. It is clear form those results that by emphasizing more weight on decreasing the buffer overflow (by increasing R_n), we get less CLR at the expense of a decrease in the video quality. This experiment assures the sensitivity of the proposed algorithm to the relative values of the parameters R_n and R_u. Hence, those two parameters can be used by the network operator to achieve certain performance objectives.

VI. Conclusions

This paper presents an adaptive congestion control scheme for the packetized voice/video sources in ATM networks, implemented at the input access node. The scheme employees the reinforcement learning method to tune the weights of the NN-based controller. The controller produces a feedback control signal to reduce the coding rate of the input voice/video process. The reinforcement algorithm uses a pre-defined a performance measure in order to evaluate the performance of the system. The cost function includes two parts with different weights, one is the CLR part and the other is the voice/video quality part. The weight on the CLR and the weight on the voice/video quality are used to achieve a certain performance of the system. The above experiments have assured the potential and the effectiveness of the proposed controller to maintain a low CLR while maintaining a high voice/video quality.

References

[1] ITU Recommendations, I series (B-ISDN), July 1992.

[2] F. Denissen, E. Desmet, and G. H. Petit "The policing function in an ATM network," in Proc. 1990 Int. Zurich Sem. Digit. Commun., Zurich, MAR. 1990, PP. 131-144.
[3] M. Butto', E. Cavallero, and A. Tonietti "Effectiveness of the leaky bucket policing mechanism in ATM networks," IEEE JSAC, Vol 9, No.3, April 1991.
[4] A. Tarraf, I. Habib, and T. Saadawi "Characterization of packetized voice traffic in ATM using neural networks," IEEE Globcom'93.
[5] A. Tarraf, I. Habib, and T. Saadawi "A novel neural network traffic enforcement mechanism for ATM networks," IEEE JSAC, special issue on Network Management.
[6] I. Habib and T. Saadawi "Congestion control in video multiplexers in broadband networks," Journal of computer comm. June 1992.
[7] T. Yamada and T. Yabuta "Neural network controller using autotuning method for nonlinear function," IEEE Trans. on Neural Networks, Vol. 3, No. 4, July 1992.
[8] K. Narendra and M. Thathachar "Learning automata," Prentice Hall, Englewood Cliffs, NJ, 1989.
[9] J Diagle, and J Langford "Models for analysis packet voice communications systems," IEEE JSAC, Vol. SAC-4, NO. 6, Sept. 1986.
[10] B. Maglaris et al "Performance models of statistical multiplexing in packet video communications," IEEE Trans. Comm, Vol. 36, No. 7, July 1988.

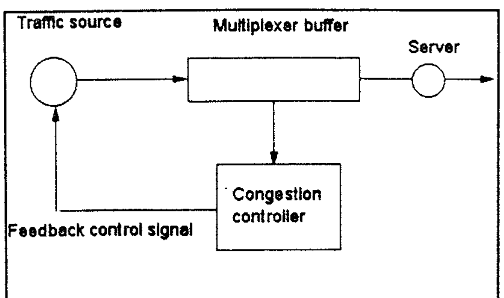

Fig. 1. Congestion Controller

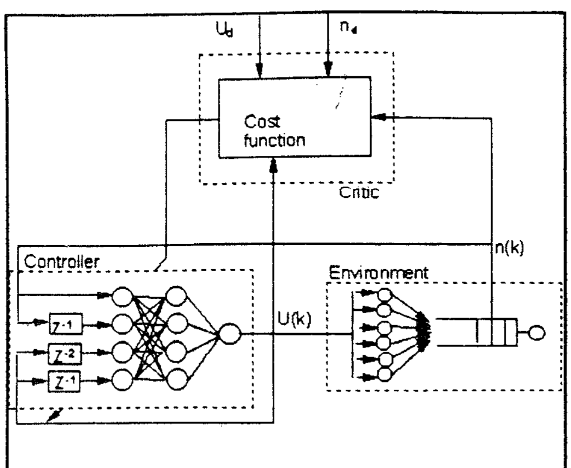

Fig. 2 Neural Network Congestion Controller

Fig. 3 Cell loss rate and voice quality versus Rn/Ru

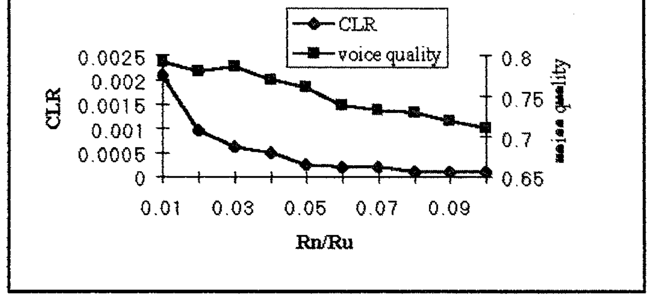

Fig. 4 Histogram of the number of the video cells in the buffer

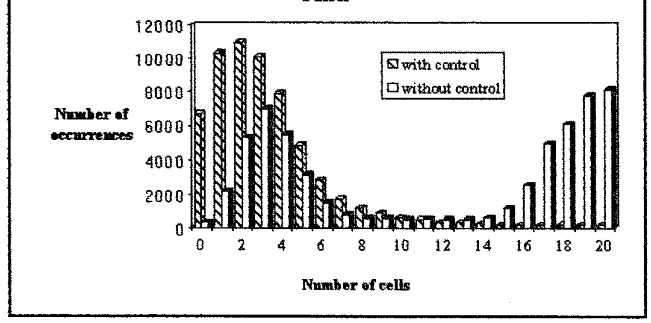

Fig. 5 Histogram of the number of the video cells in the buffer

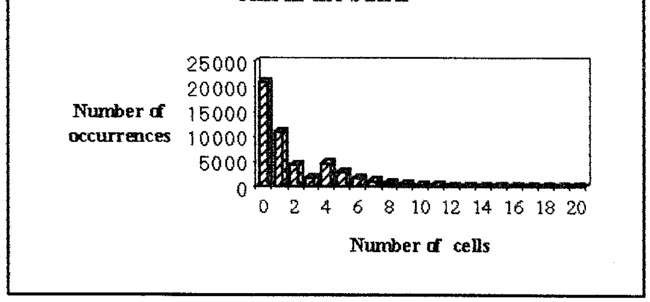

Fig. 6 Histogram of the feedback control signal