Journal of Zhejiang University SCIENCE
ISSN 1009-3095
http://www.zju.edu.cn/jzus
E-mail: jzus@zju.edu.cn

Congestion control for ATM multiplexers using neural networks: multiple sources/single buffer scenario

DU Shu-xin (杜树新)†, YUAN Shi-yong (袁石勇)

(*National Laboratory of Industrial Control Technology, Institute of Intelligent Systems and Decision-Making, Zhejiang University, Hangzhou 310027, China*)

†E-mail: shxdu@iipc.zju.edu.cn

Received June 10, 2003; revision accepted Aug. 11, 2003

Abstract: A new neural network based method for solving the problem of congestion control arising at the user network interface (UNI) of ATM networks is proposed in this paper. Unlike the previous methods where the coding rate for all traffic sources as controller output signals is tuned in a body, the proposed method adjusts the coding rate for only a part of the traffic sources while the remainder sources send the cells in the previous coding rate in case of occurrence of congestion. The controller output signals include the source coding rate and the percentage of the sources that send cells at the corresponding coding rate. The control methods not only minimize the cell loss rate but also guarantee the quality of information (such as voice sources) fed into the multiplexer buffer. Simulations with 150 ADPCM voice sources fed into the multiplexer buffer showed that the proposed methods have advantage over the previous methods in the aspect of the performance indices such as cell loss rate (CLR) and voice quality.

Key words: Congestion control, ATM networks, Neural networks, Source coding rate
doi:10.1631/jzus.2004.1124 **Document code:** A **CLC number:** TP393

INTRODUCTION

As one of the transport methods for the broadband integrated services digital networks (B-ISDN) recommended by ITU, the asynchronous transfer mode (ATM) offers enough flexibility to statistically multiplex different types of traffic with different quality of services (QoS) requirements, e.g., cell loss rate (CLR), cell delay, delay variability. Because of the uncertainties in the traffic pattern and unpredictable statistical fluctuation of traffic flows, congestion may still occur even though an appropriate connection admission control (CAC) scheme is provided by the traffic control (Jagannathan and Talluri, 2002). In order to prevent the QoS from severely degrading during short-term congestion, an appropriate congestion control scheme is required.

Congestion control at the user network interface (UNI) of ATM networks is viewed as changing the source rates and regulating the traffic submitted by these sources onto the network connections, shown in Fig.1, where N stands for the number of input traffic sources. Congestion closed-loop feedback control schemes can be divided into reactive control type and preventive control type. In the reactive control scheme (Habib and Sasdawi, 1995), the current queue length (i.e., number of cells) in the multiplexer buffer is monitored to detect congestion, and when the current queue length is greater than a threshold limit, i.e., possible traffic congestion is detected, feedback signals are sent back to all sources and the source coding rate is decreased by a fixed rate. During periods of buffer

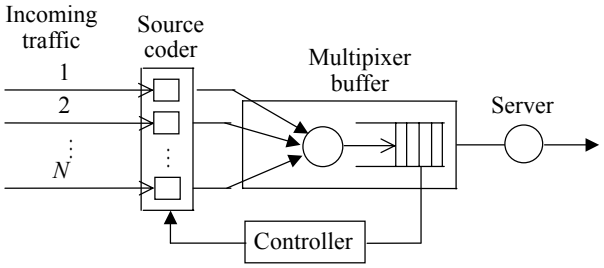

Fig.1 Closed-loop feedback control system in ATM neural networks

under-load, the coding rate is returned back to its previous level. Such a control usually uses a predefined threshold for the queue length. However, the determination of the threshold is a big problem. In the preventive control scheme, by monitoring the current queue state, the future queue behavior is predicted by intelligent modeling methods such as neural networks (Fan and Mars, 1997) and fuzzy neural networks (Lee and Hou, 2000), and based on the predicted results, the required source coding rate as the feedback control signal is sent back to all sources. The preventive control methods are concentrated in the ATM congestion control because its control action taken can be in time to alleviate the potential congestion.

In the preventive control scheme, various congestion control methods have been proposed. Fan and Mars (1997) predicted the incoming traffic to the multiplexer buffer on the basis of the current queue length, and the buffer queue length of the next time cycle was calculated according to Lindley's equation. Lee and Hou (2000) predicted cell loss by the current queue length, current queue change rate and previous queue change rate, and as soon as cell loss was detected, a feedback control signal was generated to change the source rate by decreasing the coding rate (number of bits per sample). In literature (Habib *et al*., 1997; Tarraf *et al*., 1995), as the controller output, the current coding rate was directly calculated by current and previous queue length and previous coding rate. From the point of control implementation, the control methods of Fan and Mars (1997) and Lee and Hou (2000) are indirect while the methods of Habib *et al*.(1997) and Tarraf *et al*.(1995) are direct and inverse-modeling control. In these methods, the coding rate for all input traffic sources is taken as the controller output signal, and the encoding rates for all input traffic sources are regulated in a body. For example, for ADPCM variable bit rate (VBR) voice sources, when potential congestion is predicted, the coding of all sources could be decreased from 4 bits/sample to 3 bits/sample while the corresponding code rate decrease from 1.0 to 0.75. However, the congestion control system is optimized not only by minimizing the cell loss rate but also maximizing the level of the coding rate since the quality of incoming cells to the multiplexer buffer is directly related to the coding rate for the traffic sources. Thus in the case of many incoming traffic sources, when potential congestion is detected, in order to maximize the coding rate statistically, the coding rate should not decrease wholly, but decrease partially, that is to say, the coding rate of only a part of sources decreases to a lower level while the coding rate of remainder sources does not change. This idea motivates us to research a new congestion control method proposed in this paper.

In the proposed novel congestion control method based on neural networks, the coding rate for input traffic sources and the corresponding source percentage are taken as the controller outputs, and are used to adjust the cells' arrival rate to the multiplexer buffer. The coding rate and the corresponding source percentage are obtained on the basis of neural network inverse modeling methods. Unlike the methods of Fan and Mars (1997), Habib *et al*.(1997), Lee and Hou (2000) and Tarraf *et al*.(1995) where the coding rate is regulated for all input traffic sources in a body, the proposed method regulates the coding rate of only a part of the sources in order that the quality of incoming cells to the multiplexer buffer are guaranteed while the cell loss rate is minimized.

The rest of the paper is organized as follows. Section II describes the proposed congestion control method including controller structure and neural network training algorithm. Simulation results are given in Section III. Finally, conclusions are summarized in Section IV.

CONGESTION CONTROL METHOD BASED ON NERUAL NETWORKS

In the proposed ATM network congestion control method based on neural networks, controller outputs at k time include the coding rate $u(k)$ of traffic sources and its corresponding user percentage $n(k)$. That is to say, at k time, the control system requires that $n(k)N$ traffic sources send the cells at the coding rate of $u(k)$ while the remainder sources send the cells at the coding rate of $u(k)$–0.25. For example, at k time, the controller outputs are $u(k)$=1.0, $n(k)$=0.8, which implies 80% of all sources send the cell at the coding rate of 1.0 to the multiplexer buffer while 20% of all sources send the cell at the coding rate of 0.75 in order to avoid congestion. When the previous congestion methods presented in literature (Fan and Mars, 1997; Habib *et al*., 1997; Lee and Hou, 2000; Tarraf *et al*., 1995) are used, all sources should send the cells at the same coding rate of 0.75. In that case, the control system is not optimal, and QoS is not optimal. Compared with the previous methods, the proposed method will improve QoS to a great extent.

Neural network controller and its structure

The three-layer neural network of the controller has 6 input neurons, 6 hidden neurons and 2 output neurons (denoted by 6-6-2), as shown in Fig.2, where $q(k)$, $u(k)$, $n(k)$ represent the queue length (or cell number) of the multipixer buffer, the source coding rate and the corresponding user percentage at sample time k, respectively. The hidden and the output layers have a sigmoid function $f(x)$ to provide the nonlinear mapping capability, which is defined by

$$f(x)=\frac{1}{1+\mathrm{e}^{-x}} \tag{1}$$

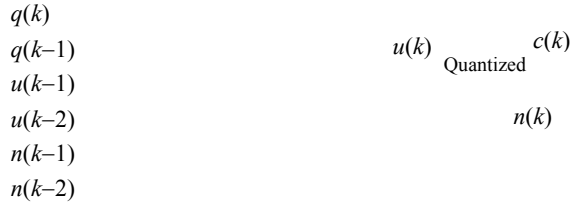

Fig.2 The structure of congestion controller

The controller outputs include $c(k)$ and $n(k)$. Since controlled source coding rate is discrete value from the set {1.0, 0.75, 0.5, 0.25, 0}, the output $u(k)$ of the neural network must be quantized.

Training for neural networks

In order to determinate the weight between the neurons, the neural network is trained so as to optimize the performance index function and get the accurate nonlinear mapping between real-value inputs and outputs. If the off-line trained neural network holds throughout the control process, it will not accurately describe the dynamic input-output modeling due to the traffic sources' time-varying, uncertainties and burst, and the control system performance will degrade greatly. If online training algorithm is used for neural networks, the control system will not satisfy the real-time requirement and high burst of the traffic sources will result in updating the weight sensitively, which leads to loss of the robustness of the control system. Therefore, the neural network training method based on moving-window is used, i.e., at every L sample times, the neural network is re-trained on the basis of the previous L sample data.

The length L of the moving window reflects the sensitivity of the controller to the dynamics of the system since the weights of neural networks are updated only at the end of each L measure periods. This is a tradeoff in the selection of this parameter. On one hand, a small L implies frequent updates of the weights of neural networks and better performance results at the expense of the possible instabilities. On other hand, relatively longer value for L assures stable control actions at the expense of long training time and the danger of inaccurate modeling.

The objective of congestion control at the user network interface (UNI) is to minimize the buffer overflow and guarantee the quality of the coding information (voice or video). Minimizing the buffer overflow is important for minimizing the cell loss rate. On the other hand, the guaranteeing of the coding information quality is important for maximizing the level of the coding rate and maximizing the corresponding user percentage. The performance index function is defined as

$$\begin{aligned}\min J(P) &= \min \sum_{k=1}^{L} \{R_q S_q(k)(q_d - q(k))^2 \\ &\quad + R_u (u_d - u(k))^2 + R_n (n_d - n(k))^2\} \\ &= \min \sum_{k=1}^{L} \{R_q S_q(k) e_q^2(k) + R_u e_u^2(k) + R_n e_n^2(k)\} \\ &= \min \sum_{k=1}^{L} \{J_k(P)\} \end{aligned} \quad (2)$$

where P: window number; L: length of the moving window; $q(k)$: number of cells (or queue length) in the buffer at sample time k; $u(k)$: the source coding rate at the sample time k; $n(k)$: percentage of the traffic sources which send the cells to multiplexer buffer in coding rate $u(k)$; q_d: desired number of cells in the buffer and $q_d \le q_{\max}$, where $q_{\max}$ is maximum length of the buffer; u_d: desired coding rate of traffic sources, generally $u_d=1$; n_d: desired percentage of the traffic sources which send the cells in the coding rate u_d, generally $n_d=1$; R_q, R_u, R_n: the corresponding weight values, the choice of which depends on the designer's objectives for the control system. If R_u and R_n are large than R_q, the control strategy will tend to give priority for achieving a good voice quality over minimizing the cell loss rate, and vice versa. $S_q(k)$: a saturation function defined by

$$S_q(k) = \begin{cases} 1 & q(k) > q_d \\ 0 & \text{other} \end{cases} \quad (3)$$

$$\begin{aligned} &e_q(k) = q_d - q(k),\ e_u(k) = u_d - u(k), \\ &e_n(k) = n_d - n(k) \end{aligned} \quad (4)$$

$e_q(k)$ represents the cell loss, and $e_n(k)$, $e_u(k)$ represent the deviation of the quality from their requirement. Hence, to minimize Eq.(2) implies minimizing the cell loss rate and minimizing the deviation of incoming quality from its original source quality.

The weight values W_{ij} between neurons are calculated by using BP learning algorithm, and W_{ij} are updated as follows

$$W_{ij} = W_{ij} - \eta \frac{\partial J(P)}{\partial W_{ij}} \quad (5)$$

$$\frac{\partial J(P)}{\partial W_{ij}} = \sum_{k=1}^{L} \frac{\partial J_k(P)}{\partial W_{ij}} \quad (6)$$

where $\eta>0$ is the learning rate parameter. Detailed description of BP algorithm is given in Haykin (1994). It is noted that $q(k)$ is the function of $u(k)$ and $n(k)$, i.e.,

$$\begin{aligned} q(k) &= q(k-1) + \text{sum}(Incell([1{:}n(k-1)N])) u(k-1) \\ &\quad + \text{sum}(Incell([n(k-1)\text{N}+1{:}N]))(u(k-1) - 0.25) \\ &\quad - Outcell \\ &= q(k-1) + \text{sum}(Incell(([1{:}N] u(k-1) \\ &\quad - 0.25\,\text{sum}(Incell([n(k-1)N+1{:}N])) - Outcell \end{aligned} \quad (7)$$

where sum() stands for calculation of sum, and $Incell([i{:}j])u(k)$ stands for the cell number sent by the sources from i to j at coding rate $u(k)$, and $Outcell$ stands for the cell output from the buffer within a sample cycle. Thus, $\partial J/(P)/\partial \boldsymbol{u}$ and $\partial J/(P)/\partial \boldsymbol{n}$ are calculated by

$$\begin{aligned} \frac{\partial J(P)}{\partial \boldsymbol{u}} &= \sum_{k=1}^{L} \frac{\partial J_k(P)}{\partial \boldsymbol{u}} \\ &= \sum_{k=1}^{L-1} \{-2R_q S_q(k+1) e_q(k+1)\,\text{sum}(Incell([1:N])) \\ &\quad - 2R_u e_u(k)\} - 2R_u e_u(L) \end{aligned} \quad (8)$$

$$\begin{aligned} \frac{\partial J(P)}{\partial \boldsymbol{n}} &= \sum_{k=1}^{L} \frac{\partial J_k(P)}{\partial \boldsymbol{n}} \\ &= \sum_{k=1}^{L-1} \{-2R_n e_n(k) - 2R_q S_q(k+1) e_q(k+1) \frac{\partial y(k)}{\partial n(k)}\} \\ &\quad - 2R_n e_n(L) \end{aligned} \quad (9)$$

where

$$y(k) = 0.25\,\text{sum}(Incell([n(k)N+1{:}N])),$$

$\boldsymbol{u}=[u(1),u(2),\ldots,u(k)]^{\mathrm{T}}$, $\boldsymbol{n}=[n(1),n(2),\ldots,n(k)]^{\mathrm{T}}$.

Because it is difficult to calculate $\partial y(k)/\partial n(k)$, in our application, it is replaced with the average value of cell number within one second divided by N. Eqs.(8) and (9) are obtained on the basis of the time scale for the moving window, and for the total time scale, they are converted to

$$\frac{\partial J(P)}{\partial \boldsymbol{u}} = \sum_{k=(P-1)L+1}^{PL} \frac{\partial J_k(P)}{\partial u(k)}$$

$$= \sum_{k=(P-1)L+1}^{PL-1} \{-2R_q S_q(k) e_q(k) \text{sum}(Incell([1:N])) - 2R_u e_u(k)\} - 2R_u e(PL) \quad (10)$$

$$\frac{\partial J(P)}{\partial \boldsymbol{n}} = \sum_{k=(P-1)L+1}^{PL} \frac{\partial J_k(P)}{\partial n(k)} = \sum_{k=(P-1)L+1}^{PL-1} \{-2R_n e_n(k) - 2R_q S_q(k) e_q(k) \frac{\partial y(k)}{\partial n(k)}\} - 2R_n e_n(PL) \quad (11)$$

SIMULATIONS

In simulation, ADPCM VBR voice sources are considered. Each voice source is simulated using the ON/OFF binary-state model (Diagle and Langford, 1986) where a fixed number of voice cells are generated at the peak bit-rate during the ON period. No cells are generated during the OFF period. Both periods are assumed to be exponentially distributed random variable with means $1/\beta$=0.35 sec and $1/\alpha$=0.65 sec, respectively. The same simulation model as one in Habib *et al*.(1997) is used, where 150 voice sources are multiplexed into a multiplexer with finite buffer length of 50 cells in order to limit the delay to 125 μs, and the link capacity is set to 1.55 Mbps, and the sampling time is set to 0.01 s. The parameters are chosen as R_n=1, R_u=0.7, R_q=0.01, u_d=1, n_d=1, q_d=30, η=0.1, L=10.

In the proposed congestion control method, the source coding rate, the percentage of its corresponding traffic sources, and the number of cells in the multiplexer buffer are respectively shown in Fig.3, Fig.4, and Fig.5. In order to compare the proposed method with the previous methods, simulations for the same model were also carried out by the method proposed by Habib *et al*.(1997), where the coding rate for all traffic sources as the controller output signal is tuned in a body, and the source coding rating and the number of cells in the buffer are respectively shown in Fig.6 and Fig.7. Comparison of their performance indices such as mean CLR during every 60 s and voice quality and buffer utilization are shown in Fig.8, Fig.9 and Fig.10. It is clear that with the proposed congestion control methods, the cell loss rate of the multiplexer buffer decrease and voice quality related to the coding rate is guaranteed.

CONCLUSION

The obvious difference of the congestion control method proposed in this paper from other methods is that the controller output signals include not only the source coding rate but also the number of the traffic sources which send the cells at the corresponding coding rate. With the addition of source percentage as a controller output signal, the control system does not adjust the source coding rate in a body as in literature (Fan and Mars, 1997; Habib *et al*., 1997; Lee and Hou, 2000; Tarraf *et al*., 1995), but adjust the source coding rate on the basis of this controller output signal. This control method not only minimizes the cell loss rate, but also guarantees the quality of information such as voice

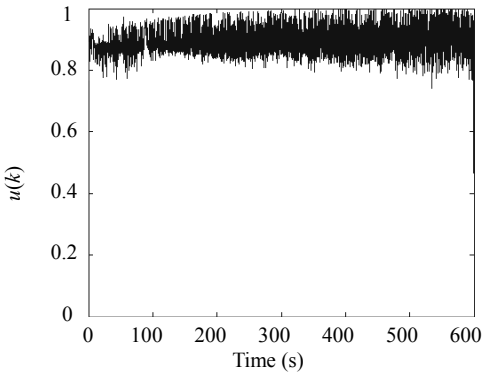

Fig.3 Source coding rate $u(k)$ with the proposed method

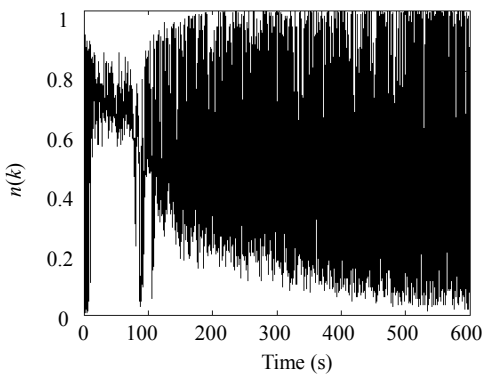

Fig.4 Percentage $n(k)$ of sources sending cells in coding rate $u(k)$

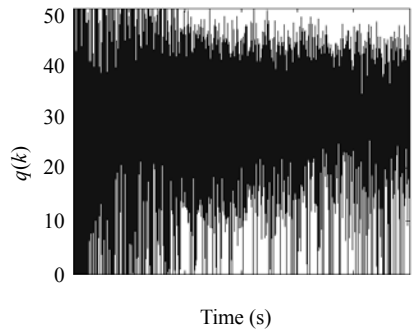

Fig.5 Number of cells in the buffer with the proposed method

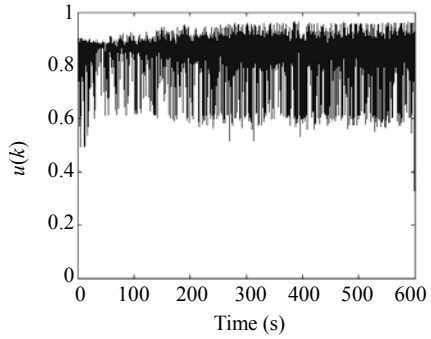

Fig.6 Source coding rate $u(k)$ with the previous methods

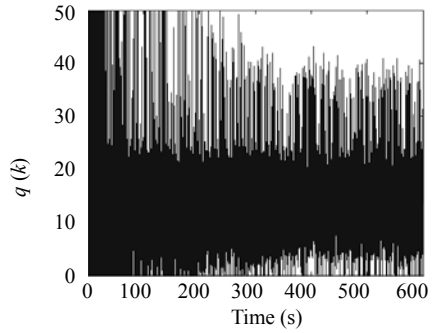

Fig.7 Number of cells in the buffer with the previous method

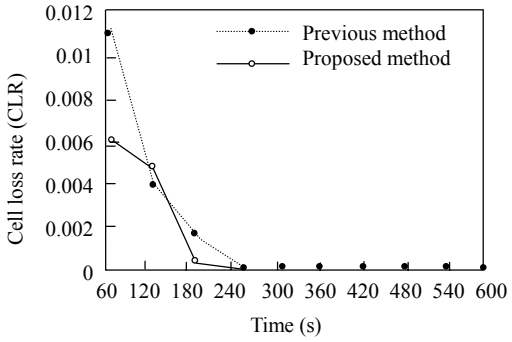

Fig.8 Comparison of mean cell loss rate

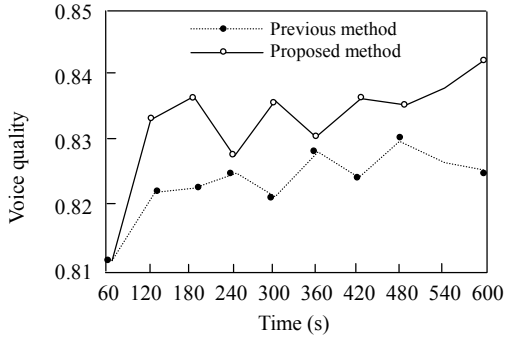

Fig.9 Comparison of voice quality

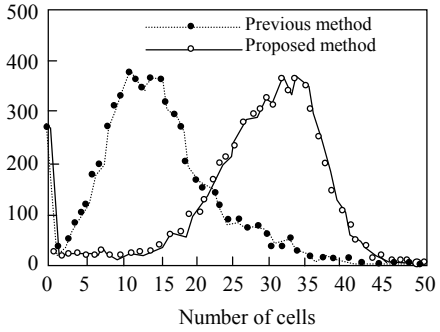

Fig.10 Comparison of histograms of the number of cells in the buffer

sources, which utilizes the ATM networks at as high a rate as possible.

References

Diagle, J., Langford, J., 1986. Models for analysis packet voice communications systems. *IEEE Journal on Selected Areas in Communications*, **4**(6):847-855.

Fan, Z., Mars, P., 1997. Access flow control scheme for ATM networks using neural-network-based traffic prediction. *IEE Proc. Commun.*, **144**(5):295-300.

Habib, I.W., Sasdawi, T.N., 1995. Access control of bursty voice traffic in ATM networks. *Computer Networks and ISDN Systems*, **27**(10):1411-1427.

Habib, I.W., Tarraf, A., Saadawi, T., 1997. A neural network controller for congestion control in ATM multipexers. *Computer Networks and ISDN Systems*, **29**(3):325-334.

Haykin, S., 1994. Neural Networks. Macmillan, New York.

Jagannathan, S., Talluri, J., 2002. Adaptive predictive congestion control of high-speed ATM networks. *IEEE Trans. Broadcasting*, **48**(2):129-139.

Lee, S.J., Hou, C.L., 2000. A neural-fuzzy system for congestion control in ATM networks. *IEEE Trans. System, Man, and Cybernetics–Part B: Cybernetics*, **30**(1):2-9.

Tarraf, A.A., Habib, W., Saadawi, T.N., 1995. Congestion Control Mechanism for ATM Networks Using Neural Networks. Proceeding of IEEE International Conference on Communications, IEEE, p.206-210.