ELSEVIER

Computer Communications 25 (2002) 9–20

computer communications

www.elsevier.com/locate/comcom

ABR traffic management using minimal resource allocation (neural) networks

N. Hock Soon, N. Sundararajan*, P. Saratchandran

School of Electrical and Electronic Engineering, Block S2, Nanyang Technological University, Singapore, 639735 Singapore

Received 7 July 2000; revised 3 March 2001; accepted 4 April 2001

Abstract

This paper presents an adaptive available bit rate (ABR) traffic management scheme in asynchronous transfer mode (ATM) networks using the newly developed minimal resource allocation network (MRAN). MRAN generates a minimal radial basis function (RBF) neural network by adding and pruning the hidden neurons based on the input data and is well suited for on-line adaptive control of time varying nonlinear systems. In this paper, the ATM traffic is modeled using the network simulation package OPNET. The performance of MRAN-controller is compared with the conventional ABR control scheme explicit rate indication with congestion avoidance (ERICA) for different traffic scenarios such as bursty and Variable Bit Rate (VBR) traffic. Results indicate that MRAN-controller performs better than ERICA by keeping the queue length and delay to a minimum while maintaining a higher link utilization and throughput. © 2002 Elsevier Science B.V. All rights reserved.

Keywords: ABR flow control; RBF network; ERICA; OPNET

1. Introduction

The asynchronous transfer mode (ATM) has been recommended by the ITU [1] to be the chosen transfer technique for broadband integrated services digital networks (B-ISDN) to support various classes of multimedia traffic with different bit rates, quality of service (QoS) requirements, efficient statistical multiplexing and cell switching. ATM networks offer the following five classes of service: constant bit rate (CBR), real-time variable bit rate (rt-VBR), non-real time variable bit rate (nrt-VBR), Available bit rate (ABR), and unspecifed bit rate (UBR) services. Of these, ABR and UBR are designed for data traffic, which has a bursty unpredictable behavior. The UBR service is simple in the sense that the users negotiate only their peak cell rates (PCR) when setting up the connection. If many sources send the traffic at the same time, the total traffic at a switch may exceed the output capacity causing delays, buffer overflows, and loss. The network tries to minimize the delay and loss using intelligent buffer allocation, cell drop and scheduling, but makes no guarantees to the application [2].

The ABR service has been introduced to support highly-bursty data applications. Most of these applications cannot predict their own traffic parameters but have well-defined cell loss requirement in order to avoid throughput collapse due to packet retransmission. In the ABR service [3], the source adapts its rate to the changing network conditions. Information about the state of the network like bandwidth availability, state of congestion and impending congestion is conveyed to the source through special control cells called Resource Management Cells (RM-cells). ABR provides better service for data traffic by periodically advising sources about the rate at which they should be transmitting. The switches monitor their loads, compute the available bandwidth and divide it fairly among the different active flows. This allows competing sources to get a fair share of the bandwidth and not be starved due to a small set of rogue sources. The feedback from the switches to the sources is sent in RM-cells, which are sent periodically by the sources and turned around by the destinations (see Fig. 1).

The RM-cells contain the source current cell rate (CCR), and several other fields that can be used by the switches to provide the feedback to the source. These fields are: explicit rate (ER), congestion indication (CI) flag, and no increase (NI) flag. The ER field indicates the rate that the network can support at a particular instant in time. When starting at the source, the ER field is usually set to the PCR, and the CI and NI flags are cleared. On the path, each switch reduces the ER field to the maximum rate it can support and sets CI or NI, if necessary. The RM-cells flowing from the source to the destination are called forward RM-cells (FRMs) while

* Corresponding author. Tel.: +65-790-5027; fax: +65-792-0415.
E-mail address: ensundara@ntu.edu.sg (N. Sundararajan).

0140-3664/02/$ - see front matter © 2002 Elsevier Science B.V. All rights reserved.
PII: S0140-3664(02)00331-0

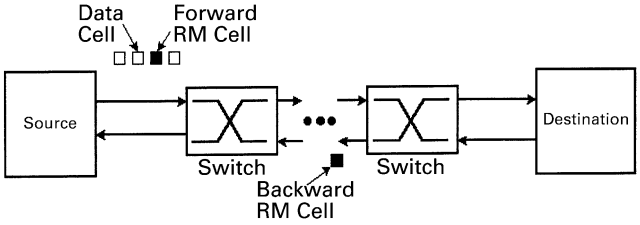

Fig. 1. RM-cell path.

those returning from the destination to the source are called backward RM-cells (BRMs). When a source receives a BRM, it computes it's allowed cell rate (ACR) using its current ACR, CI and NI flags, and the ER field of the RM-cell.

A number of rate-based ABR traffic management schemes have been developed in the literature. One of the well-developed ABR control scheme is the Explicit Rate Indication with congestion avoidance (ERICA) Scheme [4]. Under this control scheme, the source monitors its load and periodically sends control cells that contain the load information. Instead of using the queue threshold detection, each switch periodically computes its input load compared with the target utilization. The switch target utilization is determined with a margin of safety required to avoid any congestion. On receiving the control cell, the switch uses the information provided in the cell and its current load to determine an appropriate rate adjustment factor, which is inserted into the control cell.

The use of artificial neural networks (ANN) in traffic management of ATM networks has gained momentum recently [5–7]. The motivation to use neural networks in traffic management for ATM networks is to utilize their learning capabilities to adaptively control a non-linear time-varying dynamic system (here, the ATM multiplexer) without having to define an accurate analytical model of the system. The neural network learns the dynamics of the system from input/output data. Another motivation is to use the adaptive capabilities of neural networks to handle unpredictable time varying and statistical fluctuations of ATM traffic, which cannot be described by theoretical models.

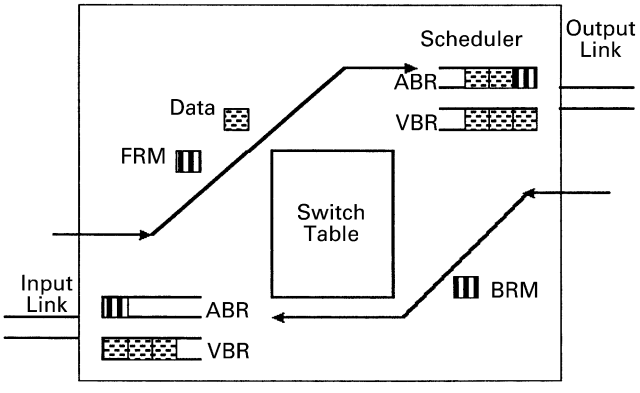

Fig. 2. ATM switch model.

Recently, an adaptive controller using a minimal radial basis function (RBF) neural network referred to as minimal resource allocation network (MRAN)has been developed in [8,9] for congestion control in ATM multiplexers. MRAN uses a sequential learning scheme for adding and pruning RBF hidden layer neurons to achieve a compact RBF neural network without sacrificing the approximation accuracy [10,11]. When no neuron is added or removed, the algorithm uses an extended kalman filter (EKF) to update the centers, widths and weights of each of the hidden neurons.

This paper presents the first application of MRAN for ABR flow control scheme in ATM networks. To test the MRAN ABR controller's performance, the network simulation package OPNET [12] is used where several traffic scenarios have been created. These scenarios include constant traffic, bursty and VBR traffic conditions. Under every scenario studied, the performance of MRAN controller is compared with that of ERICA and the results show that MRAN's performance is better than that of ERICA.

The paper is organized as follows. Section 2 presents the basic ABR flow control mechanism that includes both the conventional ERICA scheme and also the proposed MRAN ABR flow controller. Section 3 briefly describes the MRAN algorithm. Section 4 presents the detailed performance results of MRAN ABR controller and also a comparison of MRAN with the ERICA algorithm under different traffic scenarios. Conclusions from this study are summarized in Section 5.

2. ABR flow control mechanism

Fig. 2 describes the basic ABR switch model. Every service class has a separate FIFO output queue which feeds to the output link under the control of a scheduling mechanism. The RM-cells of a connection enter the switch through one port in the forward direction and exit through another port in the reverse direction. Here, we monitor the forward flow for metrics and give the feedback in the backward RM-cells for minimizing the latency in delivering feedback to the sources.

The ABR service was designed to achieve a high throughput with a control over cell loss. Further development of feedback mechanisms has led to a control in queuing delays and also provides a combination of quick response time and a stable steady state. At the same time, the switches can also compensate efficiently for errors due to variation in network loads and capacity. Typically, there is a trade-off between the link utilization and the switch queuing delay. For low utilization, the switch queue is small and the delay is small. Once utilization is very high, the queues grow. Finally, when the queue size exceeds the available buffer size, cells are dropped. In this state, though the link utilization may be high, the effective end-to-end throughput is low.

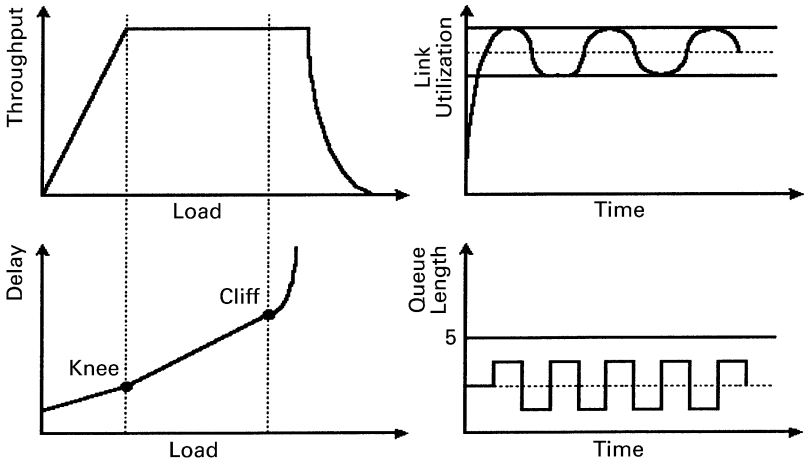

Fig. 3. Target throughtput-delay behavior of the system.

Fig. 3 shows the throughput and delay with a varying load in the network. The operating point which has a utilization close to 100 and moderate delays is called the *knee* of the delay-throughput curve. Formally, the knee is the point where the ratio of the bottleneck throughput to bottleneck response time (delay) as a function of input load is maximized. In a network which is at an ideal operating point, typical utilization graphs have a steady state with controlled oscillations close to 100 utilization, and typical queue length curves have a steady state with controlled oscillations close to zero queue length curves. For flow control schemes, the 'knee' is a good choice for an operating point. Schemes which operate (or close to) this point are called *congestion avoidance schemes*. Congestion avoidance is one notion of efficiency in the ABR service.

If the load is increased beyond the knee, the delay increases as a function of the load, but there is always a non-zero queuing delay. However, beyond a certain delay, the throughput drops again and the (end-to-end) delays rise sharply due to higher layer mechanisms like timeout and retransmission. This point is called the 'cliff' of the delay-throughput curve. The cliff is a highly unstable operating point and has large queuing delays. Operating points between the knee and the cliff may also be desirable. Such operating points keep the network at 100% utilization in steady state and maintain a 'pocket of queues' in the buffer.

2.1. Conventional ABR flow control-ERICA algorithm

The ERICA [13] algorithm was presented at the ATM Forum in February 1995. Since then, its performance has been studied in many papers [2,14,15]. A brief description of the ERICA algorithm is given below. The ERICA algorithm operates at each output port (or link) of a switch. The switch periodically monitors the load on each link and determines a load factor (z), the ABR capacity, and the number of currently active virtual connections or VCs (N). A measurement or 'averaging' interval is used for this purpose. These quantities are then used to calculate the feedback, which is indicated in the RM-cells. The measurements are made in the forward direction, whereas the feedback is given in the reverse direction. Further, the switch gives at most one new feedback per source in any averaging interval.

The load factor is calculated as the ratio of the measured input rate at the port to the target capacity of the output link.

$$z = \frac{\text{ABR input rate}}{\text{Target ABR capacity}} \tag{1}$$

where

$$\text{ABR capacity} = \text{Target utilization}(U) \times \text{Link bandwidth} \tag{2}$$

The input rate is measured over an interval called the switch averaging interval. The above steps are executed at the end of the switch averaging interval. target utilization (U) is a parameter, which is set to a fraction (close to, but less than 100%) of the available capacity. Typical values of target utilization are 0.9 and 0.95.

The load factor, z, is an indicator of the congestion level of the link. High overload values are undesirable because they indicate excessive congestion; so as the low overload values, which indicate a link under utilization. The optimal operating point is at an overload value equal to one. The goal of the switch is to maintain the network at unit overload. The fair share of each VC called FairShare, is also computed as follows:

$$\text{FairShare} = \frac{\text{Target ABR capacity}}{\text{Number of active VCs}} \tag{3}$$

The switch allows each source sending at a rate below the FairShare to rise to the FairShare value every time it sends a feedback to the source. If the source does not use all of its FairShare value, then the switch fairly allocates the remaining capacity to the sources, which can use it. For this purpose, the switch calculates the quantity:

$$\text{VCShare} = \frac{\text{Current cell rate(CCR)}}{z} \tag{4}$$

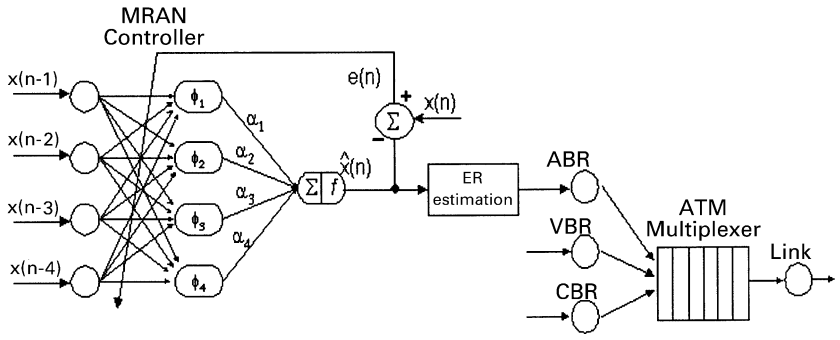

Fig. 4. Online MRAN control scheme.

If all the VCs changed their rate to their VC Share values, the switch would experience unit overload (z equals one) in the next cycle. Hence VC Share aims at bringing the system to an efficient operating point, which may not necessarily be fair. FairShare allocation aims at ensuring fairness, possibly leading to overload (inefficient operation). A combination of these two quantities is used to rapidly reach optimal operation as follows:

$$\text{ER calculated} \leftarrow \text{Max}(\text{Fair Share}, \text{VCShare}) \quad (5)$$

ER is a field in RM-cell specifying the maximum cell rate a source may use for transmission over an ABR virtual connection (VC). Sources are allowed to send at a rate of at least the FairShare value within the first round-trip. This ensures minimum fairness between sources. If the VCShare value is greater than the FairShare value, the source is allowed to send at VCShare value, so that the link is not under utilized. This step also allows an unconstrained source to proceed towards its max–min rate. The previous step is one of the key innovations of the ERICA scheme because it improves fairness at every step, even under overload conditions.

The calculated ER value cannot be greater than the ABR Capacity which has been measured earlier. Hence:

$$\text{ER calculated} \leftarrow \text{Min}(\text{ER calculated}, \text{ABR capacity}) \quad (6)$$

To ensure that the bottleneck ER reaches the source, each switch computes the minimum of the ER it has calculated as above and the ER value in the RM-cell. This value is inserted in the ER field of the RM-cell:

$$\text{ER in RM cell} \leftarrow \text{Min}(\text{ER in RM cell}, \text{ER calculated}) \quad (7)$$

This is how ERICA algorithm sets the appropriate rate of the ABR sources (Eqs. (1)–(7)) through the RM-cell.

A complete pseudo code of the ERICA scheme including all its features is provided in Ref. [2]. The performance of the ERICA algorithm depends significantly upon the way the measurements are done and its parameters are chosen. Some features of this scheme have been designed with scalability and high variance conditions to ensure the robustness of ERICA algorithm [13]. These include switch averaging intervals, reliable counting of number of active VCs and Load/Capacity averaging.

2.2. ABR flow control using MRAN

It is well known that congestion control schemes in ATM networks require specific knowledge of the statistical behavior of the input traffic via its traffic descriptors. Traffic descriptors using simple parameters will not accurately characterize very rapid changes in the bit rate variations of the traffic over short time intervals and often ignore the bursty nature of the traffic. On the other hand, those mechanisms using more sophisticated parameters are computationally expensive and impractical.

Although ERICA and its updated version has performed well for the ABR flow control, they are still not fast enough to handle sudden changes. Furthermore, the effectiveness of these algorithms greatly depends on the way of measurements and its parameters are chosen. In order to develop a more robust, fast, simple and effective ABR flow control scheme, the newly developed MRAN is studied for this application. MRAN will predict the dynamic changing multimedia traffic based on the past data. The role of MRAN is to capture the unknown complicated relationship between past and future cell arrival rate of the traffic. Fig. 4 shows the adaptive ABR flow control scheme using MRAN.

The definition of the different variables shown in Fig. 4 is given below:

$e(n)$ = Prediction error of cell arrival rate
$x(n)$ = VBR cell arrival rate at time interval n
$x(n)$ = Predicted VBR cell arrival rate at time interval n
$x(n-1) \ldots x(n-4)$ = Previous VBR cell arrival rates.

In the proposed scheme, the control signal is generated based on the real time measurement of arrival rate process. This control signal is the predicted VBR traffic. The network is designed to make a prediction of sample $x(n)$ referred to as $\hat{x}(n)$, given the past q samples $x(n-1), x(n-2), \ldots, x(n-q)$. MRAN is trained online to minimized the prediction error: $e(n) = x(n) - \hat{x}(n)$. The ABR service's ER value is estimated once in every time interval. The estimated

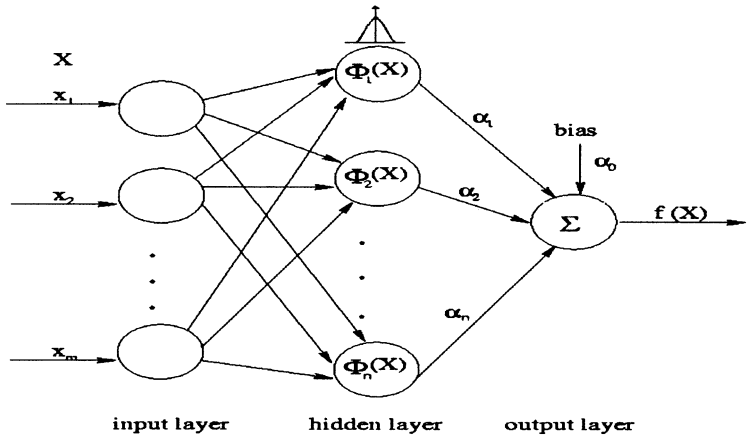

Fig. 5. RBF network model.

ER value is

$$\mathrm{ER}_p = \frac{(C_\mathrm{L} - \hat{x})}{N_\mathrm{ABR}} \tag{8}$$

where C_L is the link capacity, N_ABR is the number of active ABR sources.

Based on this prediction of VBR traffic background, fairly precise allowable cell rate (ACR) for the next time interval is obtained. The ABR source end-system will then adjust its transmission rate in such a way that fairness and effectiveness for the ABR services can be achieved while gaining a high utilization at the same time. Before a detailed performance comparison of ERICA and MRAN ABR control schemes is presented, a brief description of the MRAN algorithm is given in Section 3.

3. Minimal resource allocation network (MRAN) algorithm

MRAN is a minimal radial basis function neural network(RBFNN) which is an improvement of the resource allocation network(RAN) of Platt [16] and the RAN via Extended Kalman Filter (RANEKF) algorithm proposed by Kadirkamanathan et al. [17]. It is a sequential learning algorithm recently developed by Yingwei et al. [10,11] which combines the growth criterion of RAN with a pruning strategy to realize a minimum RAN.

Fig. 5 shows a typical RBF network, with m inputs $x, (x_1–x_m)$, and one output. The hidden layer consists of n hidden neurons ($\Phi_1–\Phi_n$), that are connected to the output by n connection weights ($\alpha_1–\alpha_n$). The output of the network is given by:

$$f(x) = \alpha_0 + \sum_{k=1}^{K} \alpha_k \Phi_k(x) \tag{9}$$

where $\Phi_k(x)$ is the response of the kth hidden neuron to the input x, and α_k is the weight connecting the kth hidden unit to the output unit. α_0 is the bias term. Here, K represents the number of hidden neurons in the network. $\Phi_k(x)$ is a Gaussian function given by,

$$\Phi_k(x) = \exp(-\|x - \mu_k\|^2/\sigma_k^2) \tag{10}$$

where μ_k is the center and σ_k is the width of the Gaussian function. $\|\ \|$ denotes the Euclidean norm.

In the MRAN algorithm, the network begins with no hidden neuron. As each input–output training data (x_n, y_n) is received, the network is built up based on certain growth criteria. The algorithm adds hidden neurons, as well as adjusts the existing network, according to the data received. The criteria that must be met before a new hidden neuron is added are:

$$\|x_n - \mu_{nr}\| > \epsilon_n \tag{11}$$

$$e_n = y_n - f(x_n) > e_\mathrm{min} \tag{12}$$

$$e_{rmsn} = \sqrt{\sum_{i=n-(M-1)}^{n} \frac{e_i^2}{M}} > e_\mathrm{min1} \tag{13}$$

where μ_{nr} is the center (of the hidden neuron) which is closest to x_n, the data that was just received. ϵ_n, e_min and e_min1 are thresholds to be selected appropriately. Eq. (11) ensures that the new neuron to be added is sufficiently far from all the existing neurons. Eq. (12) decides if the existing neurons are insufficient to obtain a network output that meets the error specification. Eq. (13) checks that the network has not met the required sum squared error specification for the past M outputs of the network. Only when all these criteria are met, is a new hidden neuron added to the network. Each new hidden neuron added to the network will have the following parameters associated with it:

$$\alpha_{K+1} = e_n,\ \mu_{K+1} = x_n,\ \sigma_{K+1} = k\|x_n - \mu_{nr}\|.$$

The overlap of the responses of the hidden neurons in the input space is determined by k, the overlap factor. When an input to the network, does not meet the criteria for a new hidden neuron to be added, the network parameters $w =$

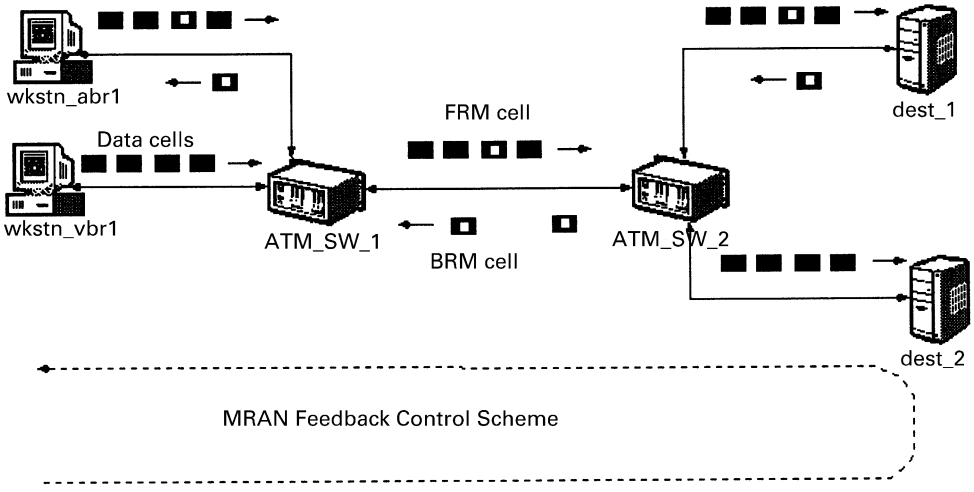

Fig. 6. ABR feedback control scheme.

$[\alpha_0, \alpha_1, \mu_1^T, \sigma_1, \ldots, \alpha_K, \mu_K^T, \sigma_K]^T$ are adapted using the EKF as follows:

$$w_n = w_{n-1} + e_n k_n \tag{14}$$

where k_n is the Kalman gain vector given by,

$$k_n = [R_n + a_n^T P_{n-1} a_n]^{-1} P_{n-1} a_n \tag{15}$$

a_n is the gradient vector and has the following form,

$$a_n = [1, \Phi(x_n), \Phi_1(x_n)(2\alpha_1/\sigma_1^2)(x_n - \mu_1)^T, \Phi_1(x_n) \times (2\alpha_1/\sigma_1^3)\|x_n - \mu_1\|^2, \ldots, \Phi(x_K), \Phi_K(x_n)(2\alpha_K/\sigma_K^2)(x_n - \mu_K)^T, \Phi_K(x_n)(2\alpha_K/\sigma_K^3)\|x_n - \mu_K\|^2]^T \tag{16}$$

R_n is the variance of the measurement noise. P_n is the error covariance matrix which is updated by,

$$P_n = [I - k_n a_n^T] P_{n-1} + QI \tag{17}$$

where Q is a scalar that determines the allowed random step in the direction of the gradient vector. If the number of parameters to be adjusted is N, P_n is a $N \times N$ positive definite symmetric matrix. When a new hidden neuron is allocated, the dimensionality of P_n increases to

$$\mathbf{P}_n = \begin{pmatrix} P_{n-1} & 0 \\ 0 & P_0 I \end{pmatrix} \tag{18}$$

and the new rows and columns are initialized by P_0. P_0 is an estimate of the uncertainty in the initial values assigned to the parameters. The dimension of the identity matrix $\mathbf{I}$ is equal to the number of new parameters introduced by the new hidden neuron.

The algorithm also incorporates a pruning strategy, which is used to prune hidden neurons that do not contribute significantly to the output of the network. This is done by observing the output of each of the hidden neurons for a period of time, and then removing the neuron that has not been contributing a significant output for that period. Consider the output, o_k of the kth hidden unit:

$$o_k = \alpha_k \exp(-\|x - \mu_k\|^2/\sigma_k^2) \tag{19}$$

If α_k or σ_k in the above equation is small, o_k might become small. Also, if $\|x - \mu_k\|$ is large, the output will be small. This would mean that the input is far away from the center of this hidden unit. To reduce inconsistency caused by using the absolute values of the outputs, their values are normalized to that of the highest output. This normalized output of each neuron is then observed for M consecutive inputs. A neuron is pruned, if the output of that neuron falls below a threshold value for M consecutive inputs. The dimensions of the EKF are then reduced to suit the reduced network.

A number of successful applications of MRAN in different areas such as non-linear system identification, function approximation and time series prediction have been reported in [10]. This is the first time MRAN is being applied to ABR traffic management.

4. Performance comparison of ERICA and MRAN schemes

The capacity of the output link is assumed to be shared between the higher priority classes (CBR, rt-VBR and nrt-VBR) and the ABR class. We bunch the higher priority classes into one conceptual class called 'VBR'. Link bandwidth is first allocated to the VBR class and the remaining bandwidth, is given to the ABR class traffic. The capacity allocated to ABR is called the ABR capacity. The problem of controlling the ABR capacity and ABR queues of the output port have been studied.

Fig. 6 illustrates a typical two-source configuration with ATM switches in the backbone.

The (wkstn_abr1) source generates a traffic to the (dest_1) while (wkstn_vbr1) source generates a traffic to

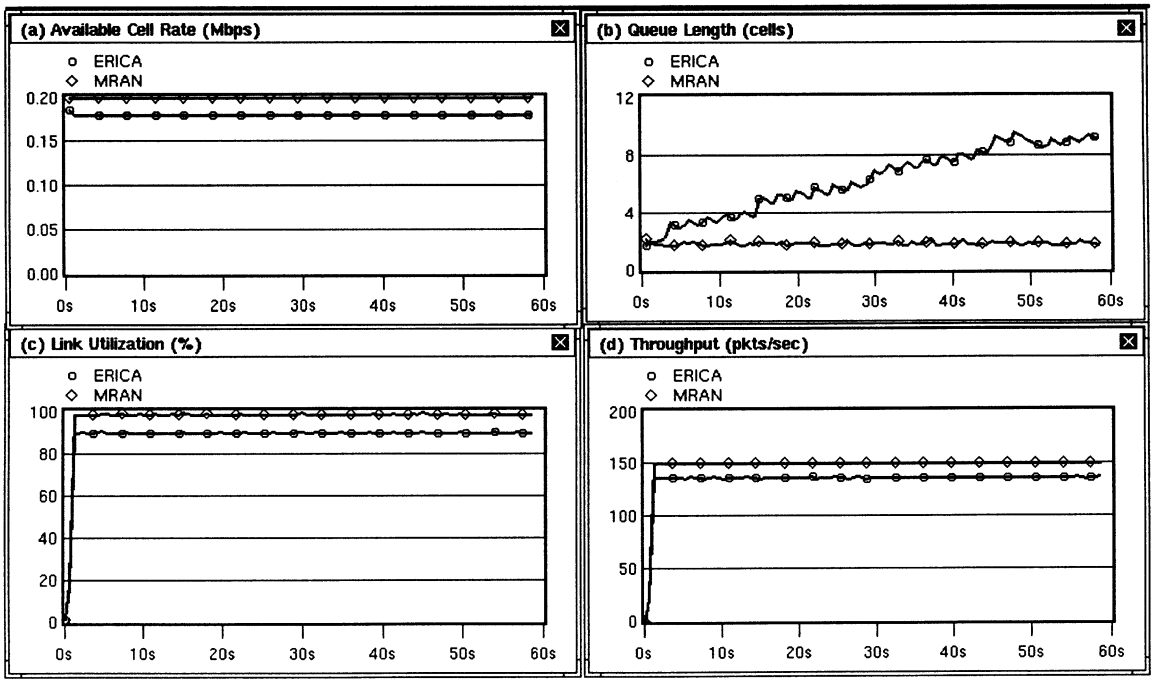

Fig. 7. Single constant traffic source.

the dest_2. It should be pointed out that even though only two sources called VBR and ABR are shown in the figure, the source VBR include all the sources generating CBR, rt-VBR and nrt-VBR traffic classes. For convenience, all these traffic sources have been bunched under a single source called VBR. VBR traffic will be served with higher priority and will send out their packets immediately if there is available bandwidth. For the ABR service, data and RM-cells are sent to the destination. The RM-cells are sent back to the source nodes by the destination nodes and take feedback information from the network to the sources. After that, the data traffic sources use these feedback information from the network to adjust their generation rates. This feedback scheme is used to avoid congestion in the network in a pro-active manner. Different traffic scenarios

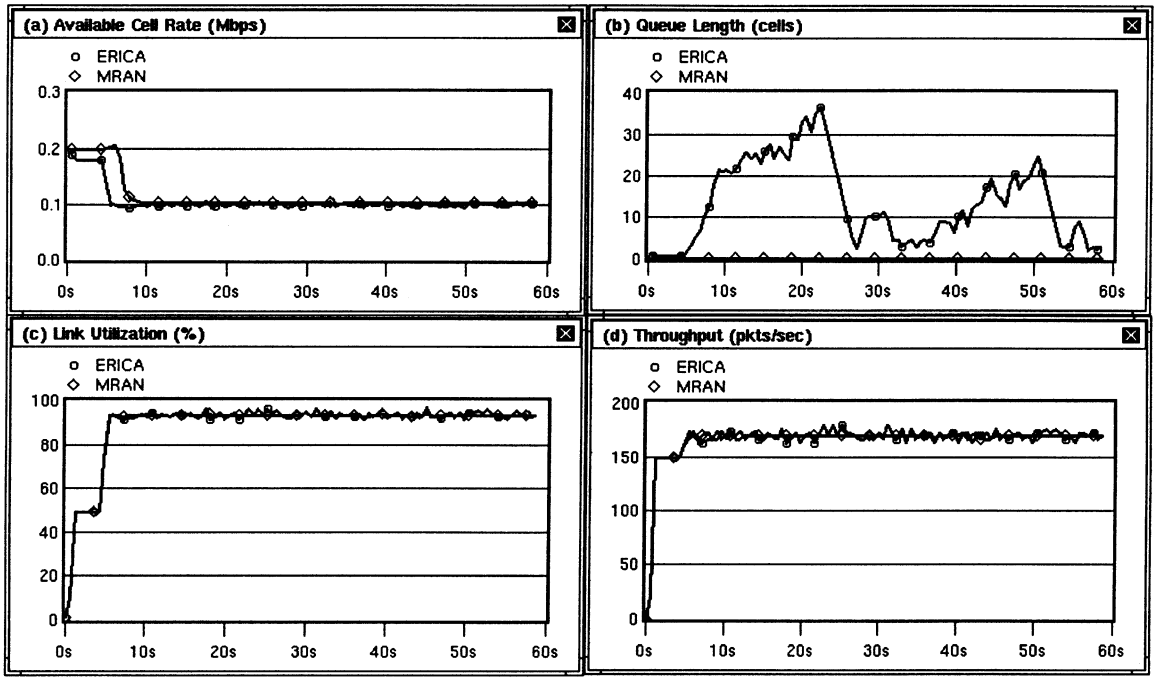

Fig. 8. Two constant traffic sources.

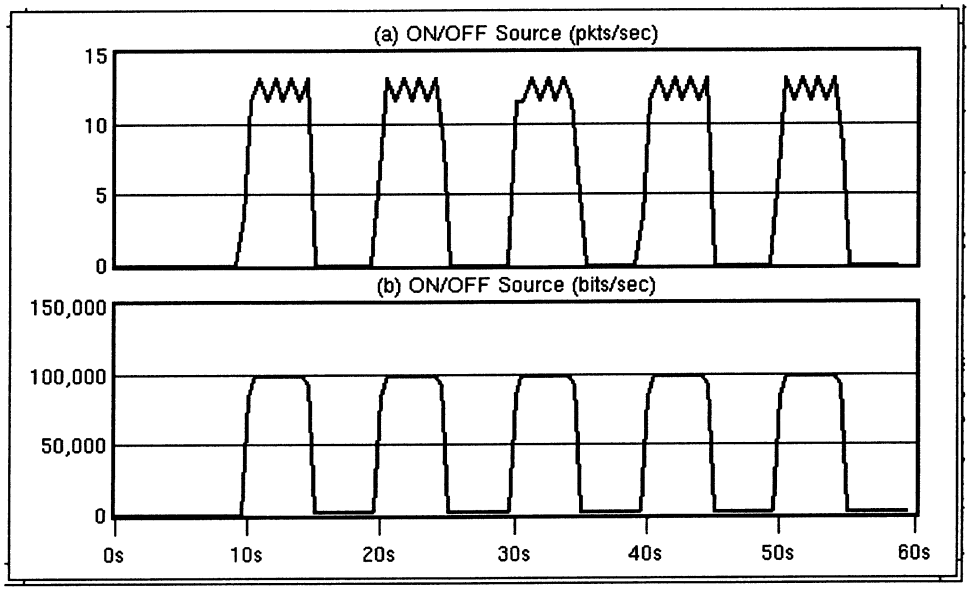

Fig. 9. ON/OFF (regular interval) traffic source.

have been created to compare the performance of ERICA scheme with the proposed MRAN flow control scheme.

The capacity allocated to ABR is called as the ABR capacity. The problem of controlling the ABR capacity and ABR queues of the output port have been studied. The results are presented in the form of the following four curves for each of the scenarios studied:

1. ACR vs. time for the ABR traffic source
2. ABR queue lengths in cells vs. time at each switch
3. Link Utilization (as a percentage) vs. time for each link
4. ATM Throughput at the destination vs. time.

The efficiency of the scheme under transient and steady state performance and its adaptation to variable capacity and various traffic sources are examined.

4.1. Efficiency

In order to verify efficient operation, a single source

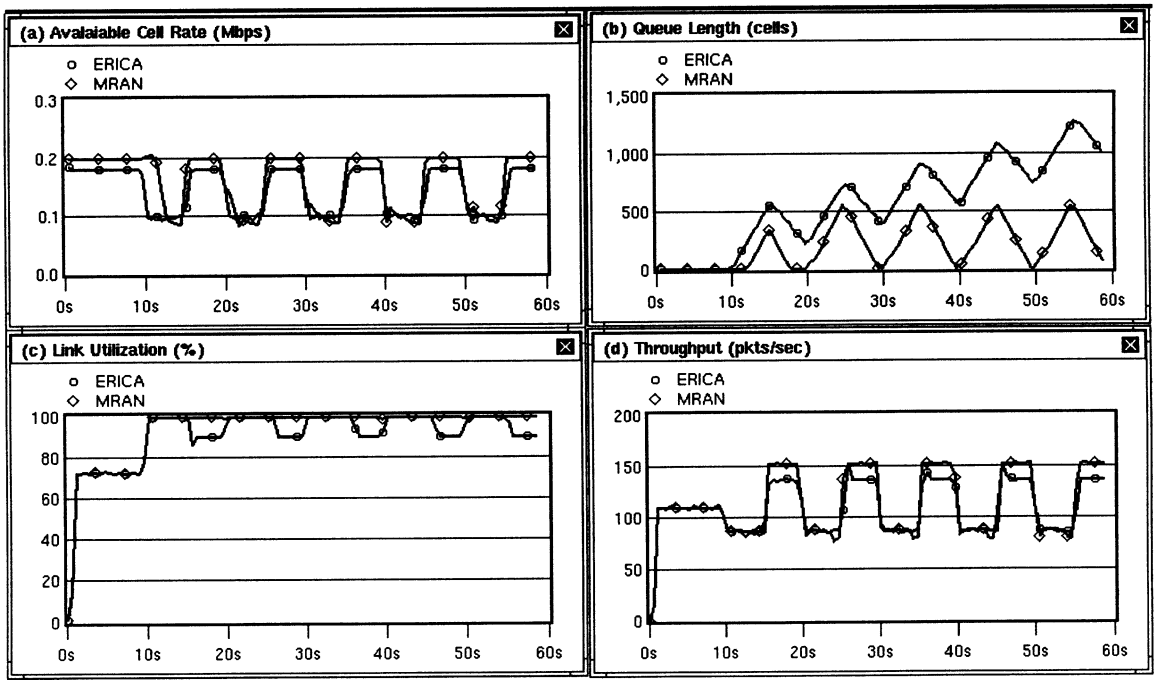

Fig. 10. 1 Constant, 1 ON/OFF (regular interval) traffic source.

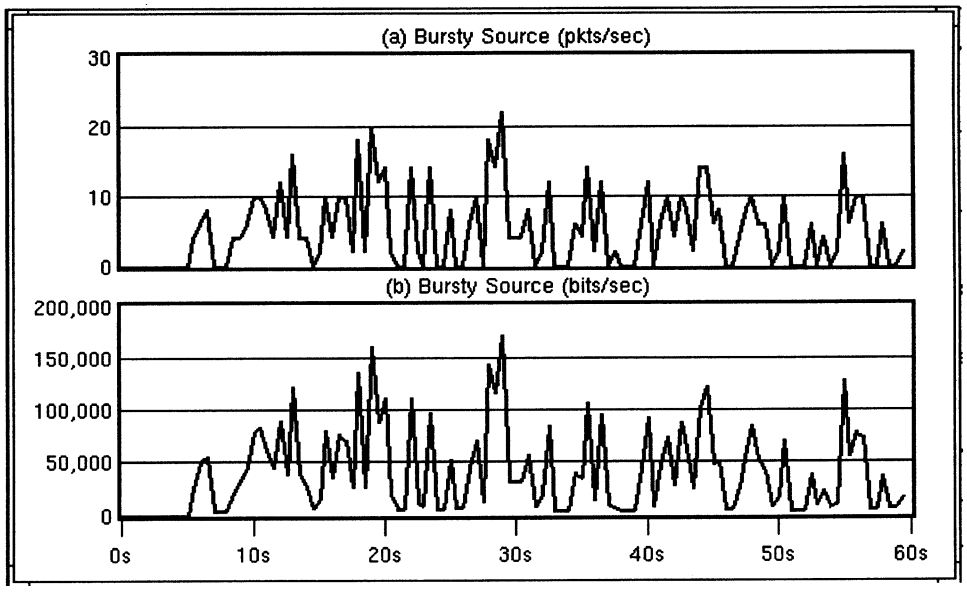

Fig. 11. Bursty traffic source.

configuration has been used. The traffic source is active over the entire simulation period. Fig. 7 illustrates that ERICA achieves the required efficiency since the source rate rises to a level which almost fully utilizes the link. There is no rate oscillation in the steady state and the utilization is at the target goal of 95. As for the MRAN scheme, the source rate rises to fully utilize the link (100 utilization) with no oscillations and minimal queues.

4.2. Minimal delay and queue lengths

After verifying both ERICA and MRAN schemes work efficiently under a single source configuration, two source configuration has been used to test for minimal delays and shorter queue lengths. Each source must converge to almost half of the link (1/2 × Target utilization), which is the max–min optimal allocation value. Fig. 8 shows that ERICA scheme is able to converge and has a good steady state performance.

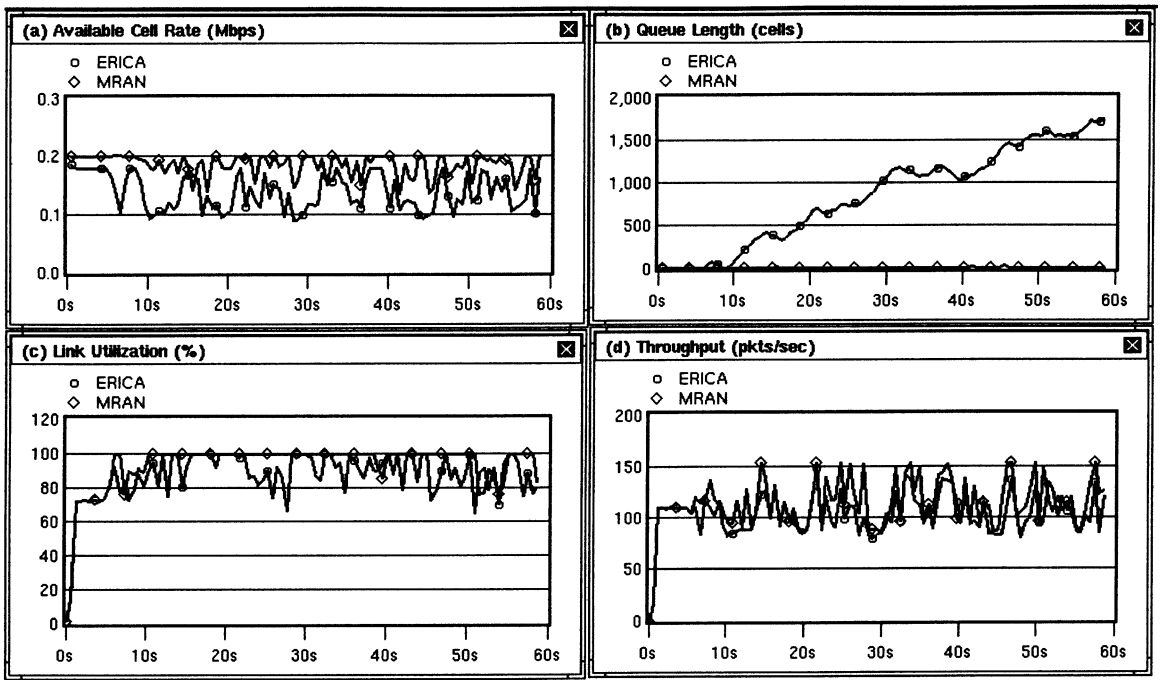

Fig. 12. 1 Constant and 1 Bursty traffic source.

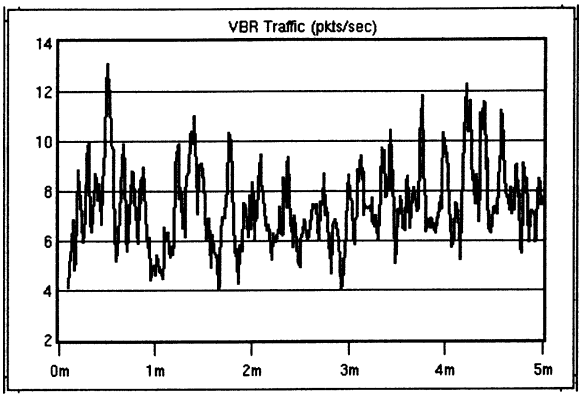

Fig. 13. VBR traffic source.

However, the queue length grows to 40 cells for a certain duration and is drained off after some time. For MRAN, the two traffic sources rapidly converge to their optimal rates and the queue lengths are controlled at about 2 cells level. In other words, MRAN reacts faster and has a smaller delay compared to the ERICA scheme.

4.3. Adaptation to bursty traffic

In ATM networks, CBR and variable bit rate (VBR) service classes have higher priority than ABR service. In the presence of VBR traffic, ABR capacity becomes a varying quantity. Here, a simple ON/OFF traffic at a regular interval of 5.0 s is simulated as shown in Fig. 9 to test the adaptability of ERICA and MRAN control schemes.

From Fig. 10, it is observed that ERICA is able to detect the change in the available ABR capacity and gives appropriate feedback to the ABR source. When the bursty source is active, the ABR source will reduce its rate. At this time, the queue length also begins to increase. The ACR will then shift back to its maximum value at the bursty traffic OFF period and this will result in the decrease of the queue length. However, it is not able to totally drain off the queues as shown in Fig. 10(b). As a result, the queue length keeps growing. The utilization is generally high; the utilization drops reflect the time taken for the feedback to reach the sources and this is called as the feedback delay (Fig. 10(c)).

On the other hand, MRAN reacts rapidly to the ABR capacity change and is able to drain off the queue length during the bursty traffic OFF period. This will keep the queue length smaller than that of ERICA and leads to a shorter delay. At the same time, MRAN is able to switch immediately for the ON/OFF burst shifting which can be seen from Fig. 10(c). MRAN is able to keep the link utilization to its maximum always without any feedback delay. The throughput of both ERICA and MRAN are almost the same which means a high utilization of the link.

The above study on bursty traffic is extended to test the varying Bursty traffic in which their ON/OFF conditions occur at irregular intervals and with varying packet arrival as shown in Fig. 11. Fig. 12(a) shows the ACR of the ABR source keeps varying in accordance to the bursty traffic load. Although ERICA is trying its best, the queue length as indicated in Fig. 12(b) still keeps on increasing. The link utilization is high most of the time except some drops due to the feedback delay. ERICA has worked with its best performance to maintain a high utilization and also maintain a slow growth in the queue length to avoid delay in such a heavy and dynamic traffic situation.

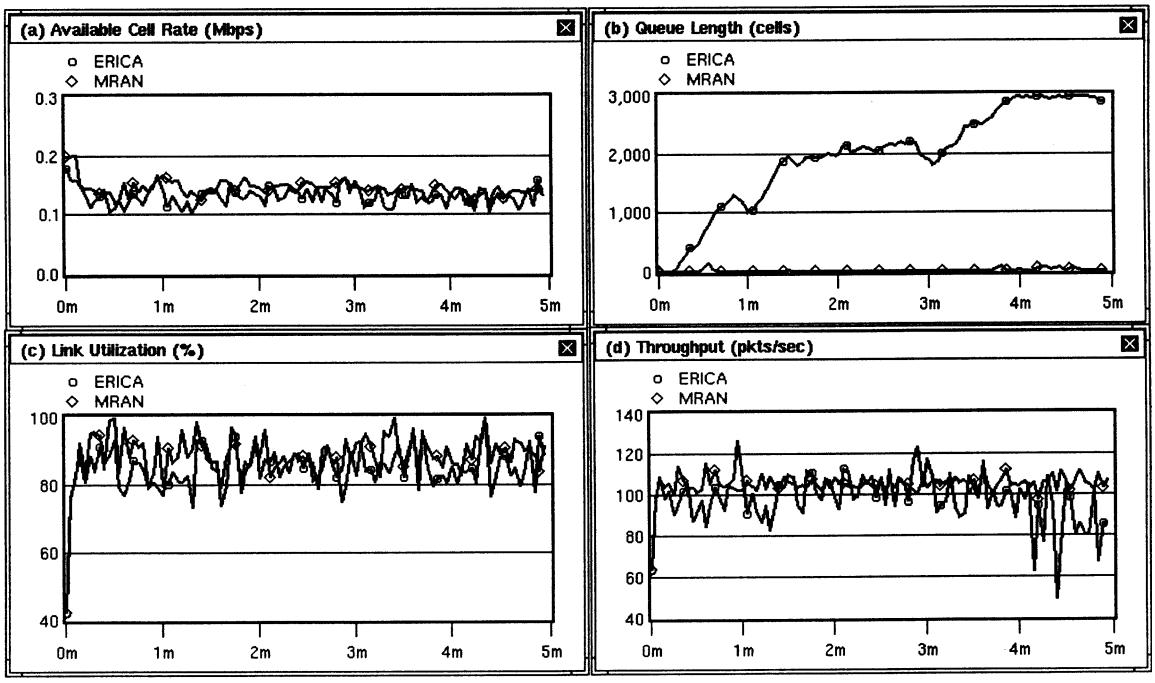

Fig. 14. 1 Constant and 1 VBR traffic source.

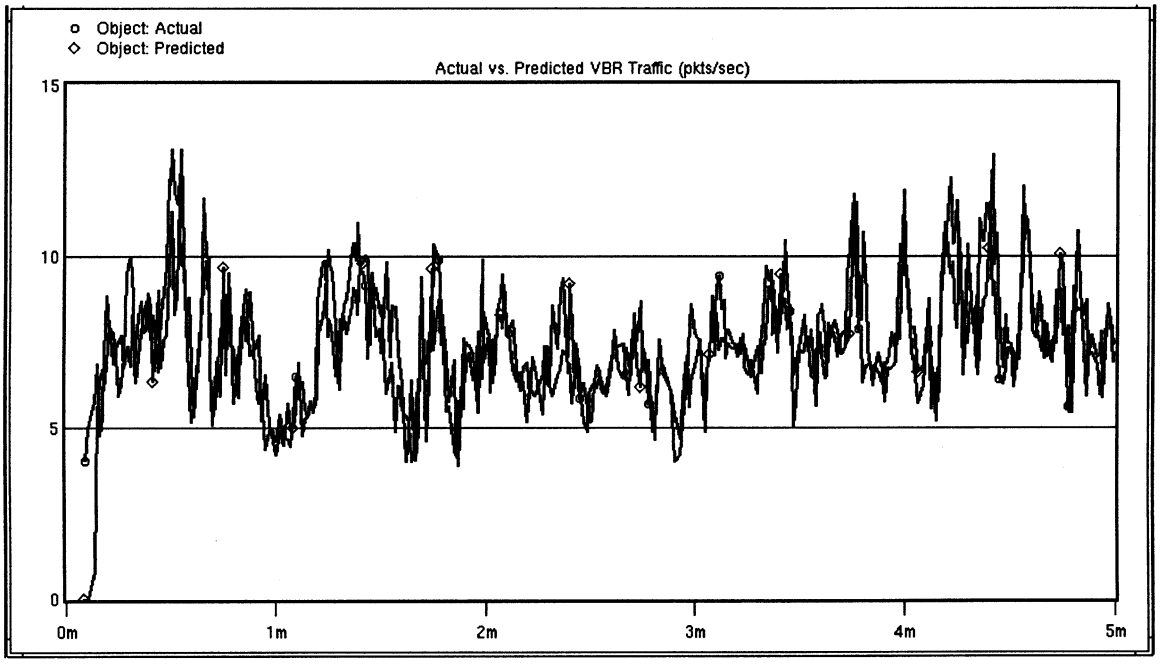

Fig. 15. MRAN performance for VBR traffic prediction.

Under MRAN flow control scheme, the ACR is always kept updated to fit with the dynamic changing bursty traffic source. The 'oscillation' in this ACR is much less compared to the ERICA case mentioned above. MRAN's ability for future prediction ensures a minimum feedback delay and provides ACR information at current interval. Consequently, the queue length as shown in Fig. 12(b) is also under control although under a high variability situation. The link utilization is at 100 most of the time which results in a high throughput.

4.4. Adaptation to VBR traffic

Finally, both ERICA and MRAN's performance under a VBR traffic condition are compared. The VBR traffic is shown in Fig. 13. This simulation has been carried out for a period of 5 min to observe the steady state performance.

As illustrated in Fig. 14(a), ERICA's ACR keeps on changing since there is a VBR traffic background. However, it seems that it is not able to handle this varying traffic scenario since the queue lengths keeps on increasing with the simulation time as shown in Fig. 14(b). This is probably caused by the traffic load which has been changed when appropriate feedback is passed back to the ABR source. Feedback delay is a critical issue for this kind of rate-based flow control scheme. Meanwhile, ERICA is still able to keep a quite high utilization with some oscillations. The overall performance of ERICA is just fair in such a case since it cannot gain from the full target utilization and minimum queue lengths.

As for MRAN, it is seen that the ACR is kept changing along the simulation time. It tries to ensure that the ABR source gets the maximum link bandwidth leftover by the VBR traffic. The performance is much better compared to ERICA scheme as observed from Fig.14. MRAN is able to keep the queue length to a minimum and obtain almost full link utilization for the whole simulation period.

It is useful to investigate the MRAN prediction accuracy for the VBR traffic. In Fig. 15, the predicted traffic is almost the same as the actual traffic. MRAN is not having any prior off-line training. It builds and adjusts its network architecture online. Thus, MRAN algorithm reacts fast enough to handle the ABR flow control in any situation.

From all the simulation results shown above, it is obvious that MRAN outperforms the existing ERICA flow control algorithm. MRAN controller is able to achieve a high link utilization with a higher throughput and maintain minimum queue lengths at the same time. Also, the prediction of dynamically changing ACR is accurate enough although MRAN just used the current and previous ACR rates.

Even though the performance of MRAN in ABR management is better than that of ERICA, MRAN algorithm contains several threshold values which have to be properly selected. Normally this is done based on trial and error and experience. Besides, the computing and memory requirements of EKF in MRAN increases with the network size. These factors may affect the use of MRAN for a real time implementation. However, the main objective in this paper has been to explore the use of MRAN for ABR traffic management without getting into the details of implementation.

5. Conclusions

In this paper, an adaptive ABR traffic management scheme using the recently developed MRAN neuro-controller is studied using OPNET simulation of ATM networks under different traffic scenarios. Both ERICA and MRAN schemes are working to give proper feedback to the ABR source so that high utilization and throughput can be obtained. At the same time, queue lengths and delay also have to be kept under control. Simulation studies show that the MRAN scheme is able to achieve significantly lower queue lengths and higher link utilization than the ERICA scheme. MRAN is able to respond faster towards traffic changes and maintain lowest queuing delay. Besides, it uses up the link utilization efficiently for all the traffic scenarios. Also, MRAN dynamically adjusts its structure to perform well even when the traffic scenario changes suddenly. It is evident that MRAN ABR flow control scheme can easily handle traffic, which is time varying and unpredictable.

References

[1] ITU-T Recommendation, I. 371 Traffic control and congestion control in B-ISDN, Helsinki, 1993.
[2] Shivkumar Kalyanaraman, Traffic management for the available bit rate (ABR) service in asynchronous transfer mode (ATM) networks. PhD Dissertation, The Ohio State University, August 1997.
[3] ATM forum technical committee traffic management working group. ATM Forum Traffic Management Specification Version 4.0, April 1996.
[4] S. Kamolphiwong, A.E. Karbowiak, H. Mehrpour, Flow control in ATM networks: a survey, Computer Communications 21 (1998) 951–968.
[5] Neurocomputing in high speed networks, IEEE Communications Magazine 33 (10) (1995).
[6] Computational and artificial intelligence in high speed networks, IEEE Journal on Selected Areas in Communication 15 (2) (1997).
[7] I. Habib, A. Tarraf, T. Saadawi, A neural network controller for congestion control in ATM multiplexers, Computer Networks and ISDN Systems 29 (3) (1997) 325–334.
[8] Ng Hock Soon, N. Sundararajan, P. Saratchandran. Neural congestion controller for ATM using OPNET (Distinguished Paper Award). In: Proceedings of the OPNETWORK '99, Washington DC, August 1999.
[9] Ng Hock Soon, N. Sundararajan, P. Saratchandran. Adaptive neural congestion controller for ATM network with heavy traffic. Proceedings of the IFIP TC 6 Fifth International Conference on BROADBAND COMMUNICATIONS '99, Hong Kong, November 1999.
[10] Y. Lu, N. Sundararajan, P. Saratchandran, Performance evaluation of a sequential minimal radial basis function (RBF) neural network learning algorithm, IEEE Transactions on Neural Networks 9 (1997) 308–318 (Vol. 9, No. 2, 1998).
[11] Y. Lu, N. Sundararajan, P. Saratchandran, A sequential learning scheme for function approximation using minimal radial basis function neural networks, Neural Computation 9 (1998) 461–478.
[12] R. Goyal, R. Jain, S. Fahmy, S. Narayanaswamy. Modeling traffic management in ATM networks with OPNET (Distinguished Paper Award). In: Proceedings of the OPNETWORK '98,Washington DC, May 1998.
[13] Shivkumar Kalyanaraman, Raj Jain, Sonia Fahmy, Rohit Goyal, Bobby Vandalore, The ERICA switch algorithm for ABR traffic management in ATM networks, IEEE/ACM Transactions on Networking 18 (2000) 87–98.
[14] A. Arulambalam, X. Chen, N. Ansari, Allocating fair rates for available bit rate service in ATM networks, IEEE Communications Magazine 34 (11) (1996) 92–100.
[15] D.H.K. Tsang, Wales Kin Fai Wong, A new rate-based switch algorithm for ABR traffic to achieve max-min fairness with analytical approximation and delay adjustment. INFOCOM '96, Proceedings of the IEEE 3 (1996) 1174–1181.
[16] J.C. Platt, A resource allocating network for function interpolation, Neural Computation 3 (1991) 213–225.
[17] Visakan Kadirkamanathan, Mahesam Niranjan, A function estimation approach to sequential learning with neural networks, Neural Computation 5 (1993) 954–975.