A Novel Neural Network Traffic Descriptor for ATM Networks

Ahmed A. Tarraf, Ibrahim W. Habib, Tarek N. Saadawi

City College of New York. Electrical Engineering Department

Abstract *Asynchronous transfer Mode (ATM) Broadband networks support a wide range of multimedia traffic (e.g. voice, video, image, and data). Accurate characterization of the multimedia traffic is essential, in ATM networks, in order to develop a robust set of traffic descriptors. Such set is required, by the Usage Parameter Control (UPC) algorithm, for traffic enforcement (policing). In this paper, we present a novel approach to characterize and model the multimedia traffic using Neural Networks (NNs). A backpropagation neural network is used to characterize and predict the statistical variations of the packet arrival process resulting from the superposition of N packetized video sources and M packetized voice sources. The accuracy of the results were verified by matching the Index of Dispersion for Counts (IDC), the variance, and the autocorrelation of the arrival process to those of the NN output. The reported results show that the NNs can be successfully utilized to characterize the complex non-renewal process with extreme accuracy.*

I. Introduction

Broadband Integrated Service Digital Networks (B-ISDN) are designed to provide multimedia traffic services (e.g. data, voice, video) over the same transport and switching systems. The Asynchronous Transfer Mode (ATM) technique has been recommended by the CCITT as the transport vehicle for B-ISDN [1]. In ATM-based networks, the multimedia information is packetized and transported in small size packets called "cells". (The cell size is 53 bytes, 5 of which is the header)

Variable Bit-Rate (VBR) video and voice sources produce multimedia traffic that possess high bit-rate fluctuations over relatively short time periods. This traffic is highly bursty and correlated in the sense that its interarrival time has a squared coefficient of variation C2 (which is the ratio of the variance to the square of the mean value) higher than that of the simple uncorrelated Poisson process. For example, it was shown in [2] that the value of C2 is 18.1 for the packet arrival process due to a single voice source, which is very large compared to that of a Poisson process which has C2 = 1. It was also observed that over short time intervals, the superposition process of M voice sources does look like a Poisson process, however over longer time intervals, the cumulative effect of positive correlations among the successive packet interarrivals cause the process to substantially deviate from the Poisson process.

In general, the traffic can be characterized by a complex non-renewal process [3]. It is essential, in ATM networks, to provide efficient admission control and traffic enforcement techniques in order to avoid serious congestion problems. These techniques require specific knowledge of the statistical-behavior of the input traffic declared via its traffic descriptors [4],[5]. Parameters such as peak bit-rate, average bit-rate, and burst length(which is the duration of the peak bit-rate) are often used as a simple set of parameters that can be used to characterize the multimedia traffic [5],[6]. More complicated second-order time domain parameters (e.g. IDC, IDI, ...etc.) are also used to fully capture the burstiness and the correlations properties of the arrival stochastic process especially those of VBR video and voice sources [7]. Mathematical models such as Continuous Time Markov Chain (CTMC), Semi Markov Modulated Process (SMMP), and Markov Modulated Poisson Process (MMPP) [8] are used to characterize and model the traffic with many approximations that limit their efficacious. Traffic characterization using simple parameters often ignores most of its important correlations and burstiness properties. Hence, leading to the definition of incorrect UPC parameters, that could seriously advert the network performance [9].

The performance of the UPC algorithm can be improved by using more sophisticated parameters that police the probability distribution function (PDF) of bit-rate of the source [10],[11]. Unfortunately in most cases, this probability distribution function can not be described by a known mathematical model. For example, the Gabarit policing mechanism [11] approximates the (PDF) by a mathematically well defined distribution (e.g. a Gaussian distribution). However, due to the complexity of its implementation, it is unfeasible to police the Gaussian envelope continuously over all points. Hence, a stair-case

0-8186-3670-X/93 $03.00 © 1993 IEEE

shape function is used to approximate the Gaussian envelope leading to errors in the estimation of UPC parameters.

In this paper, we adopt a novel approach to this problem by using NNs. NNs can predict the bit-rate variations of a complex stochastic process using simple parameters such as the instantaneous bit-rate which is measured by the number of packet arrivals within a certain time-period. The NN approach would, not only, best characterize (the bit-rate variations) of the multimedia arrival process but also predict these variations over a fixed time interval in the future. Hence, it can be used to dynamically allocate the bandwidth, to each call, and enforce the traffic subsequently. We believe that it is the only possible solution to actually avoid congestion in ATM networks before it occurs. The traffic prediction is achieved as the NN learns the actual (PDF) of the traffic. The results show that, with a proper NN architecture, the traffic can be accurately characterized and consequently described with simple parameters. The advantages of the NN approach over other computational algorithms (e.g. [7]-[9]) is the simplicity of implementing a UPC function that can characterize and predict the bit-rate variations of the traffic and hence police it. The motivation behind our selection of NNs is that it is very effective in learning, and hence predicting, non-linear complex functions, thus making it an ideal tool to employ in ATM networks.

Neural networks were proposed to solve some control problems [12],[13]. Applications to the communications networks control were reported in [14]-[17]. However NNs have not been used in the context of multimedia traffic modeling and characterization. Section II describes the NN architecture and the proposed model, section III reports simulation results and section IV contains numerical results whereas the conclusions are given in section V.

II. Neural Network Model For Traffic Characterization And Prediction

The main objective of this paper is to explore how a three-layered backpropagation NN can be used to characterize and predict non-renewal type processes. NNs based on backpropagation algorithm, can learn a nonlinear relation between many variables. These networks have been used in a number of deterministic and stochástic problems [18], [19]. It has been found that these networks perform well in most cases with accurate results.

Before we start the analysis of the NN model for traffic prediction , we briefly describe the video arrival process and the voice arrival process as follows:

Video Arrival Process: In this paper, the video traffic is generated by simulating a variable bit-rate (VBR) video source which comprise mainly head and shoulder video sequences types without scene changes. The picture sequences, produced by the video source, comprise a lot of redundant information, hence a suitable compression technique is employed to remove this redundancy, while maintaining a constant picture quality. A number of coding algorithms are employed to code the video signal such as interframe differential pulse code motion (DPCM), intraframe DPCM, conditional replenshiment (CR), motion compensation discrete cosine transfer (MC-DCT) [20]-[22]. In this paper, a simulation of a (VBR) source employing scene without abrupt movement is used.

The change in the coding bit rate process of a VBR video source is described by a number of mathematical models [23],[24], however, the burstiness of the (VBR) video source depends on the compression algorithm and the nature of the video scene. For a scene without abrupt movement (head and shoulders video type), it was proved in [23] that the bit-rate has a bell shaped stationary probability density and has exhibit significant correlations for an interval of several frames. Also, this burstiness depends on the time scale to evaluate the coded information variation. In this paper, the bit rate encoded information is evaluated frame by frame. Rate variation caused by the video signal line scanning is assumed to be smoothed out before packet assembly.

In this paper the simulation model used to generate the (VBR) video coded traffic is a continuous-state discrete-time stochastic process. A first order Autoregressive (AR) markov process X(n) that takes into consideration the autocorrelation of the sequence is used. The definition of the AR process is as follows [23]:

$$X(n) = aX(n) + bW(n) \qquad \textbf{(1)}$$

Where X(n) is the bit rate during the n th frame, W(n) is a sequence of independent Gaussian random variables where a and b are constants.

Voice Arrival Process: This process possesses correlations among the number of packets arrivals in adjacent time intervals. This complexity is due to the bursty nature of the packet arrival process from single voice source [8]. The ON/OFF periodic process is used to model the voice source, where both ON and OFF time periods are assumed to be exponentially distributed random variables with means $1/\alpha$ and $1/\beta$, respectively. During the ON period a fixed number of packets is generated, each of duration T.

Backpropagation networks have basically three layers, the input layer, hidden layer(s), and an output layer. Each layer contains a number of processing elements (PE), and is fully connected to the next layer. The input data vector is presented via the input layer which fans out the input data without making calculations. The data flows along the connections toward the hidden layer(s) and the output layer. The final result of this operation is that the input data vector is transformed (mapped) into some corresponding output vector at the output layer. Each PE in a hidden or output layer has a connection from each PE in the preceding layer. Associated with each of these connections is an adaptive weight. The output of a PE in a hidden or output layer is calculated by applying an activation function to the weighted sum of the input to that PE. Various activation functions such as S-shaped functions (sigmoid) and bump function (Gaussian), [19] are available.

During the learning phase of the network, the actual output data vector is compared to the desired output data vector, and the errors between these two vectors are calculated. The error values are then used to calculate the new wights for all output and hidden layers PEs and thereby reduce the error in the network output. This process is repeated until the mapping from the input vector to the output vector has been trained to the desired accuracy. The idea is to find a set of weight values that result in maximum accuracy and minimum error. The error criterion used by backpropagation networks is the Mean Squared Error (MSE) [25].

The role of the NN, in this application, is to capture the unknown complex relation between the past and future values of the traffic. In other word, the NN is employed as an adaptive predictor that learns the stochastic properties of the traffic. Figure (1) illustrates the basic idea in training a NN to act as a predictor. The packet arrival process, from the source(s), is represented by the data vector [H(i+m)] which is the NN target output vector. [H(i+m)] provides the NN with the bit-rate information from which the predictions will be made. The NN predicts the bit-rate variations by exploiting the inherent correlations that exist among the arrivals in the packet arrival process. For training purposes, the input data vector [(H(i)] to the NN, is the delayed value of the data vector [H(i+m)]. The NN, then tries to match the target output data vector [H(i+m)] with its predicted output data vector [$\hat{H}$(i+m)] by adjusting its weights. It, then, follows that when the input to the NN bypasses the delay unit, the output vector [$\hat{H}$(i+m)] is a prediction of the values the traffic will have in the future. The delay unit, shown in figure (1), delays its input [H(i+m)] for m time steps. Assuming that the NN requires a negligible amount of time to compute its output from its input, then the NN, after training, provides estimates for the values of traffic [$\hat{H}$(i+m)] m steps in the future. This approach to adaptive prediction rests on the assumption of a parameterized class of models for functional relationship between the current and past values of the traffic and its later values, or equivalently between earlier values of the traffic and its current values.

Figure (2) Shows the backpropagation NN structures used in this paper. The offered traffic to the NN is the multimedia traffic resulting from superposition of N video sources and M voice sources. The NN model used is

$$[\hat{H}(i+m)] = NN_f\{[H(i)], [W]\} \qquad \textbf{(2)}$$

NN_f denotes neural network transfer function, where [W] presents the weight matrices of the hidden and output layers. The vector [H(i)] can be used to present the instantaneous values of bit-rate over the past measurement period (T_m) up to the present instant i. Alternatively, the same vector [H(i)] can be used to represent the count process N(0,t) which measures the number of packet arrivals in time (0,t). In this paper, the count process N(0,t) is used for traffic descripdor in ATM networks due to its robustness against the delay variability of the packets interarrival times. Thus, the vector [H(i)] consists of m samples of bit-rate process or m samples of the count process. These samples are obtained by sampling the arrival process at every sampling period (T_s). [$\hat{H}$(i+m)] is the output vector from the NN, presenting the expected traffic over the next measurement period (T_m), see figure (3). It then follows that,

$$T_m = m * T_s \qquad \textbf{(3)}$$

The Neural Network input traffic pattern [H(i)] is expressed as

$$[H(i)] = \begin{bmatrix} h(i) \\ h(i-1) \\ \cdot \\ \cdot \\ \cdot \\ h(i-m-1) \end{bmatrix} \qquad \textbf{(4)}$$

Where h(i) is the value of h(t) at the sampling instant i expressed in packets/sec. The target traffic pattern for the next m measurement intervals [H(i+m)] is expressed as

$$[H(i+m)] = \begin{bmatrix} h(i+m) \\ h(i+m-1) \\ \cdot \\ \cdot \\ \cdot \\ h(i+1) \end{bmatrix} \qquad \textbf{(5)}$$

Where m represents the number PEs in the NN input and

output layers.
The measurement period T_m, and the sampling period T_s have a direct effect on the NN structure and complexity. The NN can characterize the traffic over an arbitrary length measurement interval T_m. However, increasing T_m while maintaining a small number of samples, within it, will give a poor prediction. Because, in this case, the number of samples in the input traffic pattern (equal to m) will not be sufficient to capture the fluctuations of the arrival process. On the other hand, increasing T_m and decreasing T_s (i.e., increasing number of samples) will give an excellent prediction, at the expense of a massive increase in the number of PEs in the NN. Increasing the number of PEs of the NN leads to a prolonged training time of the NN (the training time of the backpropagation is proportional to the number of PEs). Needless to mention, the complexity of the NN physical realization will be magnified by several orders of magnitude. So firstly, T_m should be selected to give a reasonable prediction window. For example, T_m is chosen to match the renewal period of 1 sec. for the voice traffic, whereas it could be set to one frame length for the video traffic. Secondly, T_s should be selected such that the input traffic vector [H(i)] would reveal the bit-rate fluctuations of the arrival process during the measurement interval. In the mean time, a reasonable number of PEs in the selected NN architecture is maintained.
In order to best select the sampling interval T_s, typical N video sources are simulated and their aggregate arrival process is observed at every T_s. We selected a (VBR) video source produces 30 frame/sec. Several experiments have been performed using different values for N and T_s. The power spectral density [26] for the sampled traffic was calculated for each experiment as shown in figure (4) and (5). It was found that the maximum frequency component of the autocorrelation function of the arrival process to be 15 Hz. In this paper, the sampling interval T_s, for the video traffic, was selected to be 1/30 sec. (1/number of frame per second). This result is expected since we choose a video source produces 30 frame/sec, and the coding information is taken to be constant during the frame. This means that observing the video traffic at every 1/30 sec will guarantee that the obtained sampled version of the video arrival process captures all correlations contained in the actual video traffic.
For the voice traffic it was shown in [27] that 10 msec. sampling interval is sufficient to capture all variations of the voice arrival process. For the multimedia traffic resulting from multiplexing of N video sources and M voice sources, it is clear that sampling interval of 1/30 sec will not be sufficient to capture the voice traffic fluctuations, however, taking the sampling interval T_s=10 msec makes sure that the instantaneous variations for both video traffic and voice traffic will be captured.

The proposed model was verified by matching the statistical characteristics of the arrival process to those of the proposed NN model. The arrival process is a correlated non-renewal process and can be characterized by several parameters such as the mean arrival rate, the variance of the number of packet arrivals, the index of dispersion for intervals (IDI), the index of dispersion for counts (IDC), the autocorrelation R(n) of the number of packet arrivals and the third moment of the number of packet arrivals. The variance of the number of packet arrivals in an interval t V(t), the IDC, and the R(n) were used to measure how well the output process, from the NN, matched the actual arrival process. These parameters were chosen since they could best capture the increase in the variance of the arrival process over the sum of consecutive intervals [5].
Let the count process N(0,t) denote the number of packet arrivals in interval (0,t). Let $M_i(t)$ be the ith moment of N(0,t), it then follows that:

$$M_i(t) = E[N^i(0,t)] \tag{6}$$

where the variance of the number of arrivals in (0,t), V(t) is given by

$$V(t) = M_2(t) - M_1^2(t) \tag{7}$$

The index of dispersion for counts satisfies

$$I(t) = \frac{V(t)}{M_1(t)} \tag{8}$$

The autocorrelation R(n) is defined as follows:

$$R(n) = \frac{E[(x(m)-A)(x(n+m)-A]}{Var[x(m)]} \tag{9}$$

where x(m) is the number of packets arrivals during the m th interval, E[.] is the expectation, var[.] is the variance, and A is the mean of x(m). A detailed calculation of V(t), I(t), and R(n) is mentioned in the next section. In performing this verification, the NN is trained to predict the count process over the next measurement interval. The traffic descriptor used in this,case is the number of packets arrivals in a consecutive intervals of length T_s.

III. Simulation Results

Extensive simulations were performed to obtain the NN data set, for both training and production phases, and also to assess the performance of various NNs architectures. The packet arrival process, resulting from superposition of N packetized video sources and M packetized voice

sources, was simulated on a Sun Sparc station 330 using the C language.
In the simulation of the video source, the first order autoregressive mentioned in section II is used. The parameter used in this model are [23]: a=0.8781, b=0.1108, the mean of W(n) is 4.3 Mbps, and the variance of W(n) is unity.
In the simulation of the voice source, the packet generation process of each voice source is modeled exactly as in [27]. The parameter used in this model are: mean active duration $1/\alpha$=350 msec, mean silence duration $1/\beta$=650 msec, fixed packetization period T=16 msec. As mentioned in section II, the video arrival process is sampled every T_s=1/30 sec while the multimedia arrival process are sampled every T_s=10 msec. To generate the NN data set, the simulation model were run for 30 min operating time and about 100 million packets were generated. In prediction of the video traffic, the sampling period T_s was chosen to be 1/30 sec. Whereas in prediction of the multimedia traffic we used T_s=10 msec.
Two simulation runs were performed. In the first one, [H(i)] and [H(i+m)] were generated by sampling the arrival stream, at every T_s, using the bit-rate as a traffic descriptor. Whereas in the second one, [H(i)] and [H(i+m)] were generated by sampling the count process N(0,t) at every T_s. Also, during the second run, the variance of the number of arrivals V(t), the IDC, and R(n) were calculated by dividing the simulation time into adjacent intervals each of length KT_s where (K=1,2,....). Let N_{KL} be the number of packet arrivals in the Lth interval. Then the variance $V(KT_s)$ is:

$$V(KT_s) = n_K^{-1}\sum_{L=1}^{n_k} N_{KL}^2 - \left(n_K^{-1}\sum_{L=1}^{n_k} N_{KL}\right)^2 \quad \textbf{(10)}$$

Where n_K is an integer division of the simulation time by K [2]. Whereas, the IDC is given by

$$I(KT_s) = \frac{V(KT_s)}{\left(n_k^{-1}\sum_{L=1}^{n_k} N_{KL}\right)} \quad \textbf{(11)}$$

The autocorrelation R(n) was calculated by forming a vector (V) such that the elements of this vector present the number of packets arrivals during the consecutive intervals (each of length T_s). The XCORR() and COV() functions of the MATLAB signal processing tools [28] were used to calculate R(n) as follows:

$$R(n) = \frac{XCORR(V,'biased')}{COV(V)} \quad \textbf{(12)}$$

The NNs were simulated using HNC Inc. EXPLORENET 3001 package [29]. Several experiments were performed using various backpropagation NNs architectures. During the work-course of this paper, two NNs architecture were used, they are called NN-1 and NN-2. NN-1 has 1 PE in the input layer, one slab (layer) in the hidden layer of 5 PEs, and 1 PE in the output layer, referred to as (1,{5},1), and NN-2 has 1 PE in the input layer, two slabs in the hidden layer (the first slab has 10 PEs, and the second slab has 5 PEs), and 1 PE on the output layer. NN-2 is referred to as (1,{10,5},1). The activation functions of these NNs were selected to be sigmoid where the steepness parameter (a factor determines the non-linearity of the sigmoid function) was selected by trial and error method to obtain good results (in the sense of V(t), IDC, and R(n)).

IV. Numerical Results

In this section, we demonstrate the validity of the proposed NN model by comparing the numerical results obtained from this model to those obtained from the simulation model. Three experiments were performed to produce the results.
Experiment (1):
In this experiment, we used N=1, M=0 (one video source and no voice sources). NN-1 was used to predict the video count arrival process over the next frame interval (m=1, T_s =1/30 sec). Figures (6), (7), and (8) show that the variance, the index of dispersion, and autcorrelation of the predicted video traffic, output from NN-1, match those of the actual traffic with small error. It is clear that NN-1 has captured the stochastic behaviors of the arrival process.
Experiment (2):
In this experiment, we used N=10, M=0 (ten video sources and no voice sources). NN-2 was used to predict the homogenous superposition video count arrival process over the next frame interval (m=1, T_s =1/30 sec). Figure (9) shows that the predicted traffic, output from NN-2 has the same statistical characteristics as those of the actual input traffic. It is clear that NN-2 performs better than NN-1 because it has higher number of PEs in its hidden layers.
Experiment (3):
In this experiment, we used N=1, M=1 (one video source and one voice source). NN-2 was used to predict the multimedia (heterogenous superposition process) count arrival process over the next sampling period (m=1, T_s =10 msec). Figure (10) shows that NN-2 characterized and predicted the multimedia traffic with extreme accuracy. It

is interesting to observe that NN-2 performance in this experiment is better than that in experiment (2), where it was exposed to video traffic only. This can be explained as follows:

The statistics of the multimedia traffic, resulted from superposition of one video source and one voice source, are dominated by the statistics of the single video source traffic as shown in figure (11). Where it shows the autocorrelation function of the multimedia traffic compared with those of the single video source traffic and single voice source traffic. It is clear that autocorrelation function of the multimedia traffic is very close to that of the single video source traffic. Hence, when we attempt to predict heterogenous multimedia traffic composite of one video and one voice, we actually attempt to predict the video arrival process, and since in this experiment the prediction window size, (m=1 with T_s =10 msec), was smaller than that for experiment (2), (m=1, T_s =1/30 sec), NN-2 performed better in this case.

V.Conclusions

Based upon the above experiments, we conclude that a suitable NN architecture can be chosen to characterize a specific type of traffic. After completing the training phase of the NN, it learns the PDF of the offered traffic. Hence, the NN can be used as an effective traffic descriptor. It describes the traffic by its actual PDF (instead of the approximated simple parameters such as the peak and mean bit-rates). In ATM networks, traffic management techniques require traffic parameters that can capture the various traffic characteristics. Adaptability to changes in the traffic characteristics is also important for robustness. The proposed model using the NNs is suitable for implementing an effective ATM traffic descriptor (UPC), since it can adaptively predict the traffic by learning the relationship between the past and the future traffic variations. The potential of the proposed model is demonstrated by its efficacious to predict the packet arrival process of the superposition of N video sources and M voice sources. The results show that NNs can be trained to learn the PDF of the arrival process hence it can function as an adaptive predictor.

References

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

[2] K. Siriram, and W. Whitt "Characterizing superposition arrival processes in packet multiplexers for voice and data,"IEEE JSAC, Vol. SAC-4. NO. 6, Sept 1986.

[3] H. Heffes, and D. Lucantoni, "A Markov Modulated characterization packetized voice and data traffic and related statistical multiplexer performance," IEEE JSAC, Vol. SAC-4, NO. 6, Sept.1986.

[4] H. Yamada, and S. Sumita "A traffic measurement method and its application for cell loss probability estimation in ATM networks," IEEE JSAC, Vol 9, NO. 3, April 1991.

[5] I. Habib, and T. Saadawi "Multimedia Traffic characteristics in broadband networks," IEEE Comm. Mag. July 1992.

[6] K. Siriram, and D. Lucantoni "Traffic smoothing effects of bit dropping in a packet voice multiplexer," IEEE Trans. Comm, Vol. 37, No. 7, July 1989.

[7] H. Satio, and H. Yamada "Analysis of statistical multiplexing in ATM transport network," IEEE JSAC Vol. 9, NO. 3, April 1991.

[8] J Diagle, and J Langford "Models for analysis packet voice communications systems," IEEE JSAC, Vol. SAC-4, NO. 6, Sept. 1986.

[9] A. Baiocchi et al "Loss performance analysis of an ATM multiplexer loaded with high-speed on-off sources," IEEE JSAC, Vol. 9, NO. 3, April 1991.

[10] E. Rathgeb "Modeling and performance comparison of policing mechanisms for ATM networks," IEEE JSAC, Vol. 9, No. 3, April 1991.

[11] 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.

[12] Special section on neural networks, IEEE control sys. Mag., Apr. 1988.

[13] Bronoko soucek "Neural and concurrent real-time systems," the six generation, John Wiley & sons, Inc., 1989.

[14] T. Takahashi and Hiramatsu, " Integrated ATM traffic control by distributed neural networks.", in Proc. ISS'90, Stockholm, Sweden, May 1990, vol. III, PP. 54-65.

[15] A. Hiramatsu, "ATM communications network control by neural networks," IEEE Trans. neural networks, vol 1, PP. 122-130, 1990.

[16] A. Hiramatsu, "Integration of ATM call admission control and link capacity control by distributed neural networks," IEEE JSAC, Vol. 9, No. 7, Sept. 1991.

[17] B. Aazhang et al "Neural networks for multiuser detection in code-division multiple access communications," IEEE Trans. Comm, Vol. 40, NO. 7, July 1992.

[18] R. Lippmann "An introduction to computing with neural nets," IEEE Assp Mag., April 1987.

[19] R. Nielsen "Neurocomputing," Addison wesly publishing comp., 1989.

[20] R. Grunenfelder et al "Characterization of video codecs as autoregressive moving average processes and related queuing system performance," IEEE JSAC, Vol 9, No. 3, april 1991.

[21] M. Nomura, T. Fujii, and N. Ohta "Basic characteristics of variable rate video coding in ATM environment," IEEE JSAC, Vol. 7, No. 5, June 1989.
[22] H. Soon et al "Statistics od video signals for viewphone-type pictures," IEEE JSAC, Vol 7, No. 5, June 1989.
[23] B. Maglaris et al "Performance models of statistical multiplexing in packet video communications," IEEE Trans. Comm, Vol. 36, No. 7, July 1988.
[24] P. Sen et al "Models for packet switching of variable-bit-rate video sources," IEEE JSAC, Vol. 7, No. 5, June 1989.
[25] D. E. Rumelhart, J. L. McClellend, and PDP Research Group, "Parallel Distributed Processing," Vol I. Cambridge, MA: M.I.T. Press, 1986.
[26] J. Proakis, and G. Manolakis "Introduction to digital signal processing," Macmillan publishing comp., 1990.
[27] A. Tarraf, I. Habib, and T. saadawi "Characterization of packetized voice traffic in ATM networks using neural networks," Globecom'93.
[28] J. Little and L. Shure "Signal Processing Toolbox, for use with MATLAB," The MathWork, Inc, August 1988.
[29] HNC Inc. "HNC ExploreNET release 2.0 operating manual," 1991.

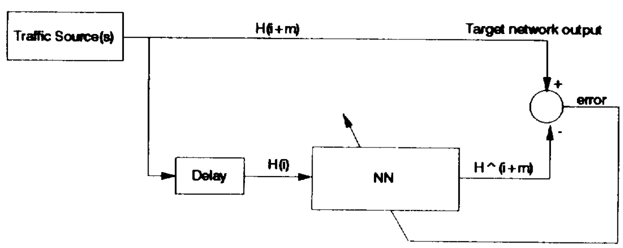

Fig.1 Using Neural Network for adaptive prediction

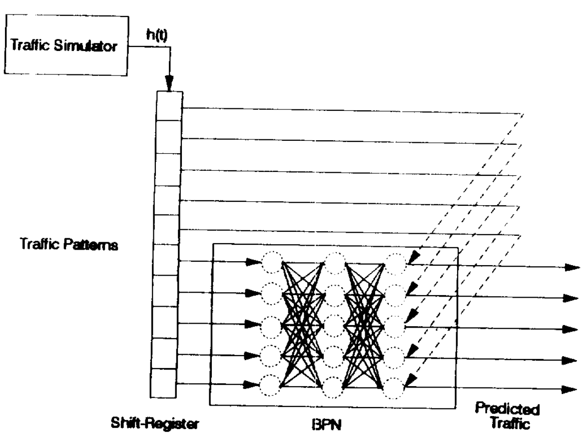

Fig. 2 Neural Network for traffic prediction.

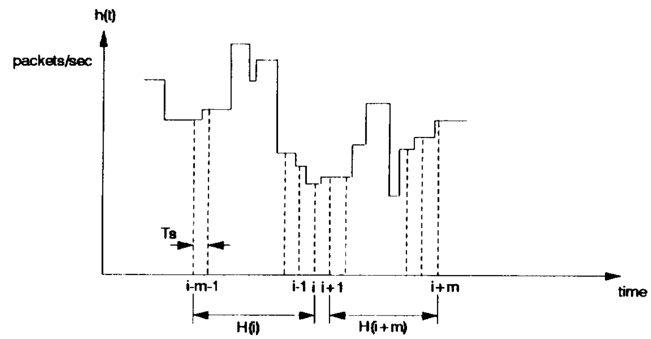

Fig 3 The sampled superposition arrival process.

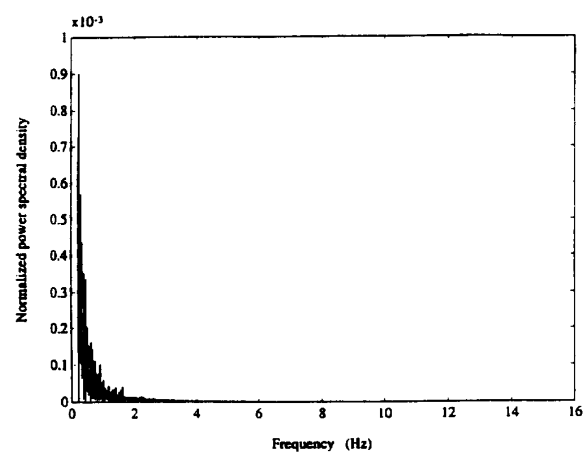

Fig. 4 The power spectral density of the video arrival process (N=10, Ts=1/30 sec)

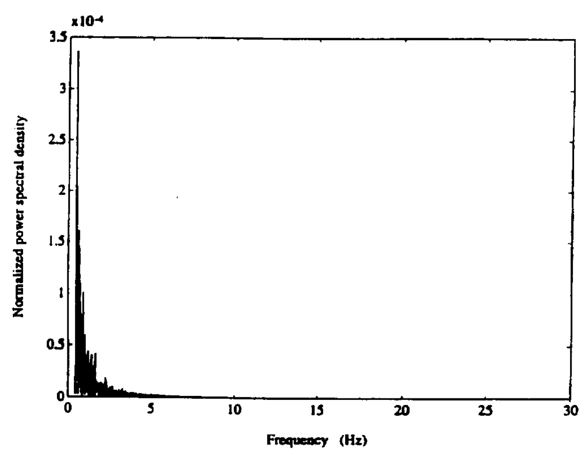

Fig. 5 The power spectral density of the video arrival process (N=10, Ts=1/60 sec)

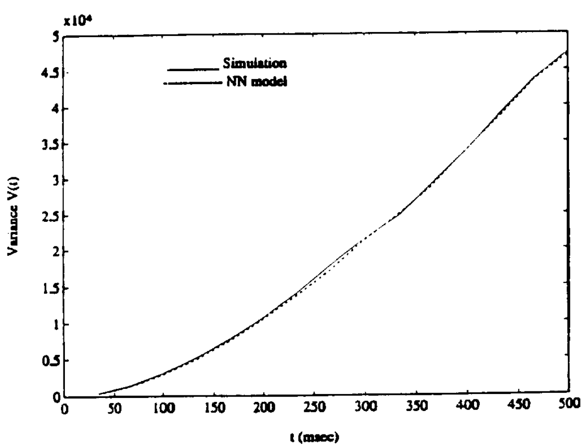

Fig. 6 Vraiance of the number of arrivals in (0,t) (N=1) (NN-1)

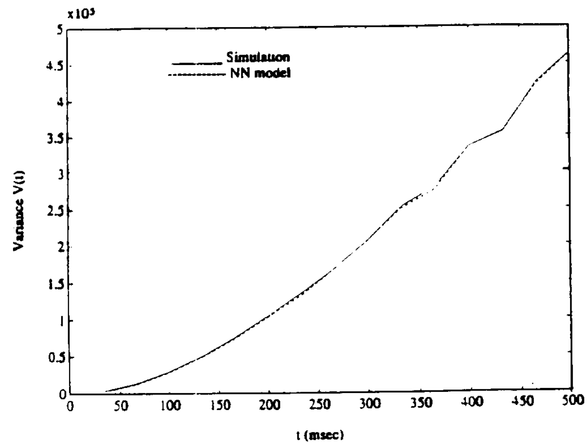

Fig. 9 Variannce of the number of arrivals in (0,t) (NN=10) (NN-2)

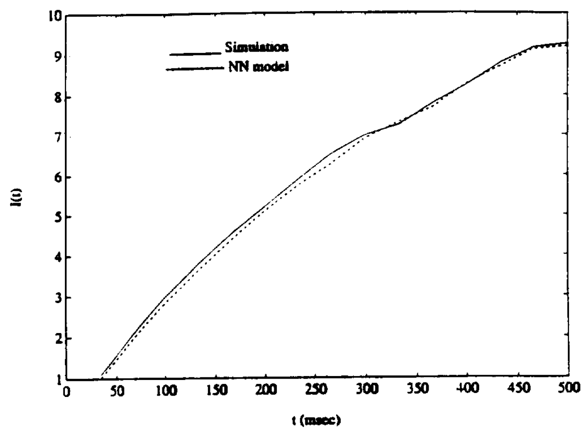

Fig. 7 Index of dispersion of number of arrivals in (0,t) (N=1) (NN-1)

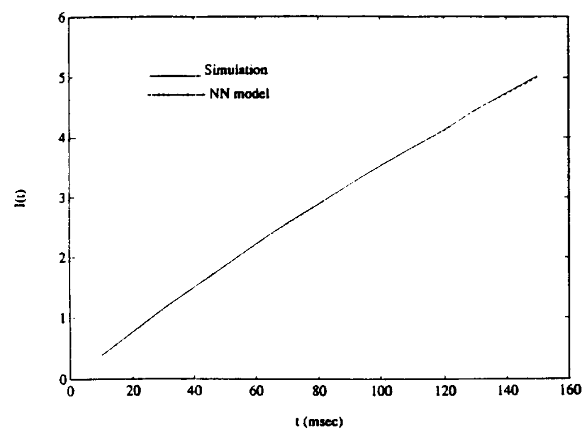

Fig. 10 Index of dispersion of number of arrivals in (0,t) (N=1, M=1) (NN-2)

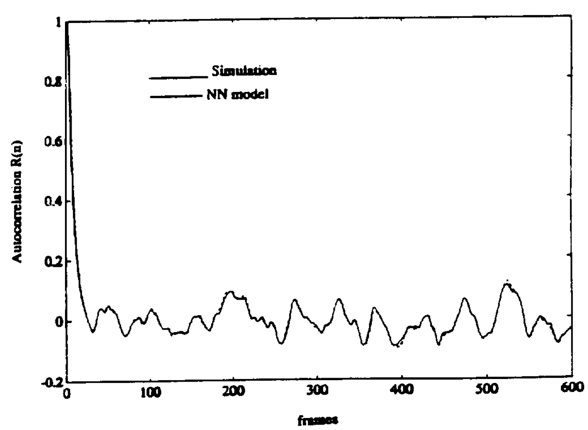

Fig. 8 Autocorrelation of the video count process (N=1) (NN-1)

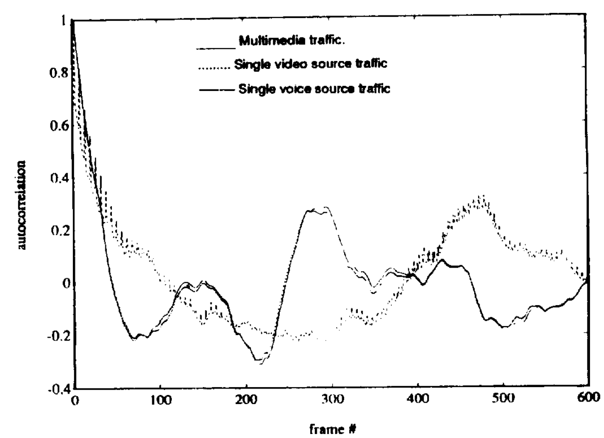

Fig. 11 Autocorrelation function.