An Artificial Intelligent Connection Admission Control Scheme for Self-Similar Traffic Using Case Based Reasoning

AFSHIN PARVARESH
Electrical Engineering Department
Science and Research Branch of
Azad University
P.O.Box: 14515-443
TEHRAN, IRAN

MOHAMMED I. KALANTARI
Electrical Engineering and Telecommunications
Department
Khaje-Nassir-Toosi University
P.O.Box: 16315-1355
TEHRAN, IRAN

Abstract : In recent high speed networks such as ATM, dealing with different kinds of traffics have become a promissing goal. Due to the dynamic nature and burstiness of aggregation of these various traffics, which is shown to be self similar, network congestion would occur more probably. Thus congestion control mechanisms (e.g. connection admission control) should be applied to preserve the quality of service of traffics passing through the network. In this paper we present a connection admission control(CAC) algorithm for self-similar traffics based on case based reasoning (CBR) which is almost a new concept in applied artificial intelligence. CBR is used to correct the estimation of the number of admitted sources to a link to achieve more link utilization than conventional CAC schemes which is known to underestimate the number of admitted sources. A case library is prepared and then the best match for a new case is retrieved from the library and reused through some experiment-based rules to obtain the best solution for the introducing case.

Key-Words: Connection Admission Control, Self-Similar Traffic, Case Based Reasoning, Quality of Service, link Utilization, Artificial Intelligence.

1 Introduction

Self-similar processes are emerging as a realistic mathematical characterization of the statistical behavior of the traffic corresponding to LAN[1], MAN[2], WAN[3], D-channel signaling in ISDN networks [4], compressed audio [5] and VBR coded video sequences [6][7][8]. Selfsimilar processes are characterized by a hyperbolic decay of their autocorrelation function that cannot be directly obtained using traditional markov processes such as poisson processes.
The presence of these different kinds of traffic and their bursty characteristics necessitates using connection admission control (CAC) mechanisms to prevent network congestion. The goal of any CAC scheme is to provide the quality of service (QOS) parameters of traffics while gaining the best usage of network resources at the same time.
Several conventional CAC techniques for high speed networks have been proposed. In the peak rate allocation, QOS is always guaranteed if the aggregate bit rate never exceeds the system capacity. However, it leads to low utilization of the network resources. An equivalent capacity (effective bandwidth) method was proposed to estimate the required bandwidth for individual or aggregate connections with desired QOS [9] [10]. A call admission scheme by inferring the upper bound of cell loss probability from the traffic parameters specified by users was studied in [11].
To evaluate the cell loss probability (as a QOS parameter) associated to each link we have to study the performance of a queuing system loaded by the composition of independent self similar traffic processes which is reffered to as a fractal queuing system. Classical queuing theory, based on Markovian approach, dedicates particular attention to poisson traffic sources since their combination is a poisson process as well. Assuming self similar input processes it is necessary to use different analytic approach [12].
A proposed diffusion approximation for the number of arrivals in the interval (0,t], considering self similar fractional gaussian noise processes, as suggested in [12], suggests:

$$A(t) \cong m.t + \sqrt{a.m}.Z(t) \qquad (1)$$

where m is the mean cell rate , a is the peakedness factor which is defined as the ratio of variance to mean of number of cells in a unit time interval and Z(t) is a normalized Fractional Brownian Motion with zero mean and variance equal to $|t|^{2H}$.

According to [12], indicating with V(t), the virtual waiting time process of the system we obtain

$$V(t) = \sup_{s \le t}((A(t) - A(s) - C.(t-s))$$

Where C is the link capacity. V(t) can represent the number of cells in the queue assuming a buffer of infinite size, by means of this process it is possible to approximate the loss probability for a queue of finite size K evaluating the probability

$$\varepsilon = \Pr(V(t) > K) \qquad (2)$$

In this sense we can obtain an upper bound for the loss probability of the finite size queue. According to the evaluations in [12] the lower bound of the complementary distribution function of the virtual waiting time process for N identical sources which is the case of this paper is :

$$\varepsilon \approx e^{-\frac{(C/N-m)^{2H}}{2am((1-H)^{1-H}.H^{H})^{2}}(BL/N)^{2-2H}} \qquad (3)$$

This relation can be used to obtain a measure of number of admitted identical sources in a CAC system with a guaranteed cell loss ratio below ε.

Studies show that, due to dynamic and bursty behavior of self similar traffics of actual networks, classical CAC schemes such as the one mentioned earlier, are not able to solve the problem in a closed form so they have to deal with some approximations which implies an overestimation of loss ratio to guarantee the desired quality of service, so lower the number of admitted sources and consequently the link utilization . A number of researchers used artificial intelligence (AI) techniques in order to overcome some of the drawbacks of the conventional methods of CAC. They are shown to deal with ill-defined problems, which is the case in traffic engineering where the traffic distributions are not well defined. In recent years, neural networks have been widely employed to deal with the traffic control problems in high speed multimedia networks [13][14][15][16]. A major feature of the neural network is the self-learning capability, which can be utilized to characterize the relationship between input traffic and system performance. Also fuzzy logic algorithms have been used to estimate the probability of cell loss ratio [17] [18]. Fuzzy logic has the ability to overcome the uncertainty resides in the computed CLRs.

In this paper we have proposed a CAC scheme using the concept of Case Based Reasoning (CBR) which is a new area in AI domain [19]. CBR is a methodology of solving new problems by adapting the solutions of previous similar problems. It is to solve a new problem by remembering a previous similar situation and by reusing information and knowledge of that situation.

In the following a call admission control scheme based on case base reasoning is presented. The rest of the paper is organized as follows: in section 2 our proposed CAC method is discussed, at first in subsections 2.1 and 2.2 the self similar traffics, which are used as traffic sources in our simulations, and CBR system which is used as an AI concept in our CAC method to improve CAC performance, are introduced respectively. Then in section 2.3 the details of our method description is brought. In section 3 the numerical results are presented and section 4 summarizes the paper.

2 The Proposed CAC scheme

This section discusses a CAC scheme using an almost new AI concept which is case based reasoning for self similar traffics.

2.1 SS traffic model

The mathematical definition of self similarity [20][1] is as follows. Let $X = (x_t : t = 1,2,3,...)$ be a covariance stationary stochastic process with mean $\mu = E[x_t]$, variance $\sigma^2 = E[(x_t - \mu)^2]$ and an autocorrelation function

$$r(k) = \frac{E[(x_t - \mu)(x_{t+k} - \mu)]}{E[(x_t - \mu)^2]} \qquad (4)$$

Only depending on k=0,1,2,… . Assume an autocorrelation function of the form $r(k) \sim ak^{-\beta}$ as $k \to \infty$ with $0 < \beta < 2$ and a is a constant. Now let $X^m = (x_k^m : k = 1,2,3,...)$ be the averaged series over nonoverlapping blocks of size m which is

$$x_k^m = \frac{1}{m}\sum_{i=km-m+1}^{km} x_i$$

With k=1,2,3,… and m=1,2,3,…. For each m, X^m is a covariance stationary process. X is called exactly second order self similar with self similarity parameter $H = 1-\beta/2$ if for all m=1,2,3,… $\mathrm{var}(X^m) = \sigma^2 m^{-\beta}$ and $r^m(k) = r(k)$. Here X is called selfsimilar when the aggregated processes X^m become indistinguishable from their root X in the sense of their autocorrelation function.

Many self similar traffic generation methods have been proposed so far such as Pareto-Tailed Sojourn Time Traffic [21], Chaotic Map Based modeling [22], and Random Midpoint Displacement (RMD) algorithm [23] which is used for FBM process generation in a given time interval. If the trajectory of FBM Z(t) is to be computed in [0,T], then we start by setting Z(0)=0 and choosing Z(T) from a gaussian distribution with mean zero and variance T^{2H}. Next, Z(T/2) (T/2 is the center of the interval [0,T]) is calculated as (Z(0)+Z(T))/2 plus an offset. The offset is a guassian random variable with a standard deviation given by T^{2H} times the initial scaling factor $s_1 = 2^{-H} s_0$ where $s_0 = \sqrt{1-2^{2H-2}}$. We then reduce the scaling factor by $1/2^H$, and the two intervals from 0 to T/2 and from T/2 to T are further subdivided from their centers, each into two equal intervals,and so on.

RMD method is used in studies of this paper because it was shown to be fast, simple, efficient and adequate for qualitative studies [23]. The cell arrival process of traffic A(t) can be written as equation (1), using Z(t) generated by RMD criteria. Thus self similar traffic sources with three parameters (m, a, H) is used in this study.

2.2 CBR

Case based reasoning is a problem solving paradigm that in many respects is fundamentally different from other major AI approaches. Instead of relying solely on general knowledge of a problem domain , or making associations along generalized relationships between problem descriptors and conclusions, CBR is able to utilize the specific knowledge of previously experienced, concrete problem situations (cases). A new problem is solved by finding a similar past case, and using it in the new problem situation. A second important difference is that CBR also is an approach to incremental, sustained learning, since a new experience is retained each time a problem has been solved, making it immediately available for future problems. The CBR field has grown rapidly over the past few years, as seen by its increased share of papers at major conferences, available commercial tools, and successful applications in daily use.

At the highest level of generality, a general CBR cycle may be described by the following four processes [24]:

1. RETRIEVE the most similar case or cases
2. REUSE the information and knowledge in that case to solve the problem
3. REVISE the proposed solution
4. RETAIN the parts of this experience likely to be useful for future problem solving

A new problem is solved by retrieving one or more previously experienced cases, reusing the case in one way or another, revising the solution based on reusing a previous case, and retaining the new experience by incorporating it into the existing knowledge-base (case-base).

2.3 Description of the CAC model

We proposed a CAC scheme based on CBR which decides to admit a number of identical sources to an ATM multiplexer with finite buffer length that sends the multiplexed traffic to an output link with capacity C. The goal of the CAC system is to raise the number of admitted sources while holding the Cell Loss Ratio (represents QOS in this study) below an objective value.

An ATM multiplexer which is characterized by it's output buffer length BL, and output link capacity C, is simulated to serve self similar traffic traces generated by RMD algorithm.

Every CBR system needs a case base, which is a library for storing past problems and solutions to be used for new problems. A case stored in the case base is defined as a selected combination of input and output parameters of the system. Here we have chosen three source parameters (m, a, H) (of a past problem), the number of admitted sources (N) (as the solution of past problem) and the resultant link parameters (the condition of the problem after applying the solution) which are cell loss ratio (CLR) and link utilization factor (UF), as case features.

Cases are defined based on the three source parameters. To construct the case base a range is defined for each of the source parameters and each range is then subdivided into six intervals. For each combination of centers of these intervals for three source parameters, one case is defined in the case base (implies $6^3 = 216$ cases). Thus the cases are categorized and then indexed for future references according to their source parameters and then the other parameters mentioned above are added to each case. The solution in each case (the number of admitted sources) is first initially normalized by a value based on the mean rate of that case (e.g. 0.8*C/mean_rate). Then during the operation of the CAC system it will gradually be updated to more appropriate value through a learning algorithm.

After creating a case base as mentioned earlier the system will proceed as follows: when a new problem arises, it means that the CAC system have to admit a number of identical sources with specified parameters, at first the indices of the parameters are computed which are used to retrieve the relevant case (category) similar to the new problem from the case base. Then the solution of the retrieved case must be reused for the new case which is done using some rules obtained by investigating many cases including the case base. These rules state that the number of admitted sources retrieved from a similar case (a past solution) might need some adjustment in order to be reused, depending on how far is the new case from the retrieved one.

Rules are constructed through knowledge achieved by investigating many cases to see how differences in values of source parameters affect the number of admitted sources . There are many rules applied in our scheme which a sample one is illustrated in relation (5):

$$\text{If } \frac{|m_2 - m_1|}{m_1} < 0.01 \text{ and } \frac{|a_2 - a_1|}{a_1} < 0.02 \text{ and } \frac{H_2 - H_1}{H_1} > 0.04 \text{ then } n_2 = 0.85 \times n_1 \qquad (5)$$

Where m is source mean rate, a is variance factor of the source, H is source Hurst parameter and indices 2 and 1 denote new and retrieved cases respectively.

To have an impression of how well our CAC system works, the performance of our method is compared to a CAC system based on the discussion of section 1. Solving equation (3) with substitution of each source parameters and server parameters (link capacity and buffer length) using numerical methods yields the maximum number of sources that the method proposed to be admitted while guaranteeing $CLR_{objective} < \varepsilon$. This number of admitted sources and resultant cell loss ratio and link utilization factor can be compared to the corresponding values from our proposed method.

A learning mechanism is used to increase the overall performance of the system. The main idea behind this algorithm lies in the fact that as long as a retrieved solution from case base which is reused to admit a number of sources, keeps the CLR in the permissible region, the number of admitted sources proposed in the case increases to raise link utilization. This action will be applied to a case in case base until it causes a failure (causes a CLR greater than the objective CLR). At this moment it decreases and never increases again. In order to implement this we use a failure file where the cases which cause QOS failure is registered there, so they can be distinguished from other cases.

3 Numerical results

As mentioned earlier a three parameters self similar traffic generator based on FBM traces is used. Using RMD for generation of these FBM traces we build traffic sources each with 4096 samples.

Figure 1 shows two sample traces which differ only in their H parameters which is shown to be a burstiness factor for the traffic trace as it can clearly be seen from the figures. The other two parameters of the traces are $m = 2300$ and $a = 300$ and are identical for both traces.

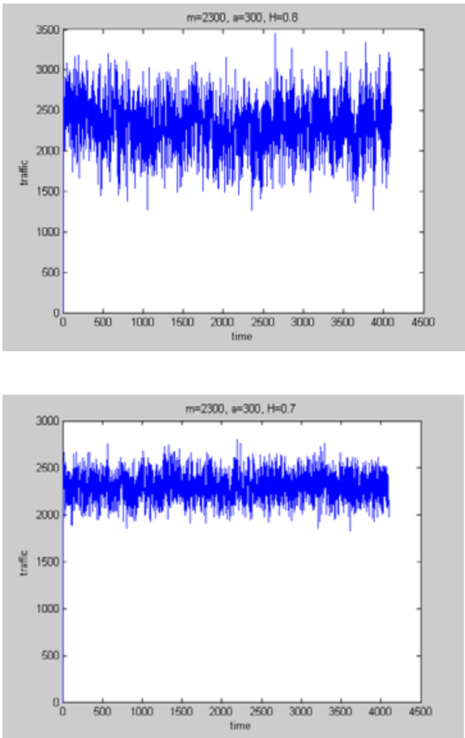

Fig.1 sample traces of self similar traffic

The traffic sources are fed to an ATM server with parameters C=2500 cell/sec and BL=300 cells which are held at constant values during our simulations.

In order to proceed with the CAC system the first step is to construct a case base as discussed in section 2 and to assign to each case an index, so later case retrieving would be much more faster. Then the CAC procedure continues with generating the three parameters of a source from a uniform random number generator. Each parameter is chosen randomly from a predefined interval, which is [10,130] cells/sec for m, [5,105] cells.sec for a and [0.55,0.98] for H. Each interval is subdivided to six subintervals to categorize and index the sources based on their parameter values. The indices of these parameters are obtained and used for retrieving a relevant case to new problem from the case base. The proposed number of admitted sources is extracted from the case and is adjusted as a solution to the new problem through the rules mentioned in section 2. This refined number of admitted sources would be the output of the CAC system which means that according to the CAC system decision a maximum of this number of specified sources (N) could be admitted to guarantee the CLR below the desired value which is chosen 10^{-4} in this work. Then to check out the performance of the CAC, a self similar traffic generator software is called to generate N traffic traces with the specified parameters each of 4096 sample length and these are fed to an ATM multiplexer/server and cell loss ratio and utilization factor in output link is obtained.

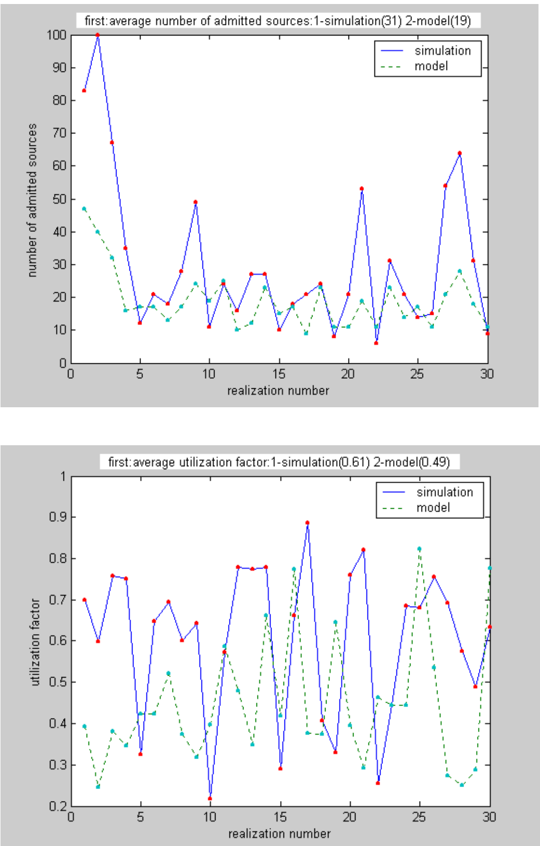

Fig.2 Comparison of the proposed CAC and an analytical method for first 30 realizations

For comparison the number of admitted sources from an analytic scheme is also computed using equation (3) and compared to our proposed system. The results are shown in figures 2-5.

In figure 2 the number of admitted sources and link utilization factor for our proposed method and the analytical model mentioned before are compared in the first 30 realization after case base initialization. As it can be seen from the figures our scheme admit more sources than the classic one which is a confirmation to the fact that the conventional CAC methods have to be more conservative to ensure the QOS being in the permissible range, thus resulting a lower link utilization factor and a waste of link capacity.

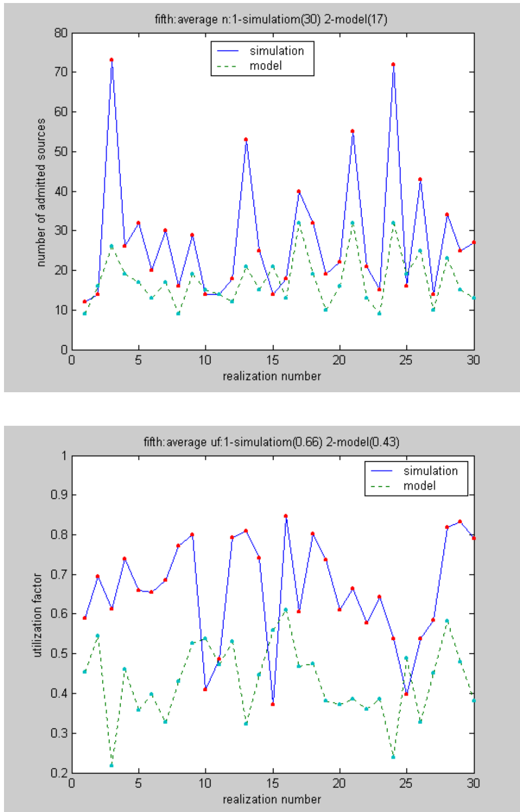

Fig.3 Comparison of the proposed CAC and an analytical method for the fifth 30 realizations

The average number of admitted sources and link utilization factor for the 30 realizations are written at the top of the figures. In order to have a better perspective of the preferences of our scheme and the effect of learning in system performance we have defined preference indices as:

$$P_N = \frac{N_{AI} - N_{model}}{N_{model}} \quad , \quad P_{uf} = \frac{uf_{AI} - uf_{model}}{uf_{model}} \qquad (6)$$

Wherein P_N and P_{uf} are preferences indices of our method in number of admitted sources and link utilization factor respectively and indices AI and model indicate our artificial intelligent method and the introduced analytical model. The higher these preference indices are, the better is the performance of our scheme. Computing these indices for the first 30 realizations yields $P_N = 0.63$ and $P_{uf} = 0.25$.

Continuing the realizations while learning is taking place result a better performance as shown in figures 3 and 4 which show the previous information for fifth and tenth 30 realizations respectively.

As it can be seen in figure 3 we have $P_N = 0.76$ and $P_{uf} = 0.53$ which compared to these indices for the first 30 realization an improvement in the system performance is obtained through the learning algorithm. The improvement continues for the next realizations which it can be seen in figure 4 for the tenth 30 realization wherein the indices are $P_N = 0.95$ and $P_{uf} = 0.62$.

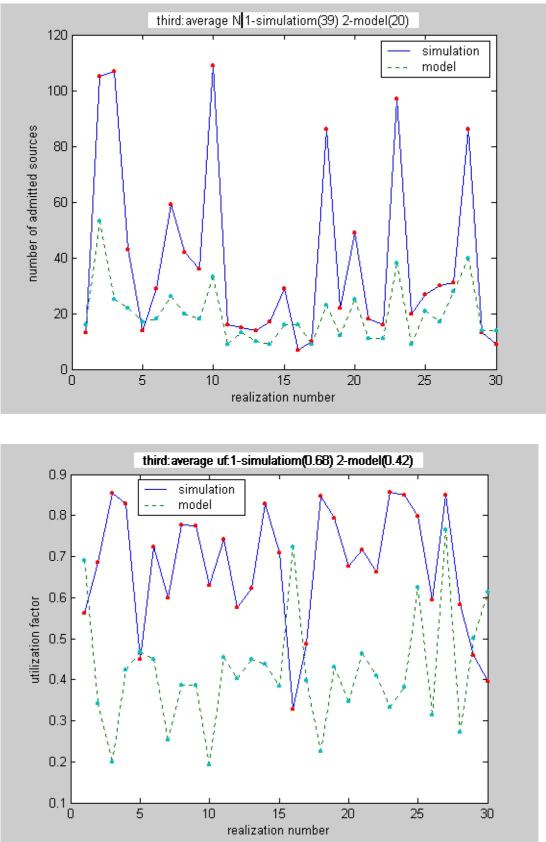

Fig.4 Comparison of the proposed CAC and an analytical method for tenth 30 realizations

The effect of learning on the performance improvement can be summarized by plotting the preference indices with respect to the number of 30 realizations, which is shown in figure 5.

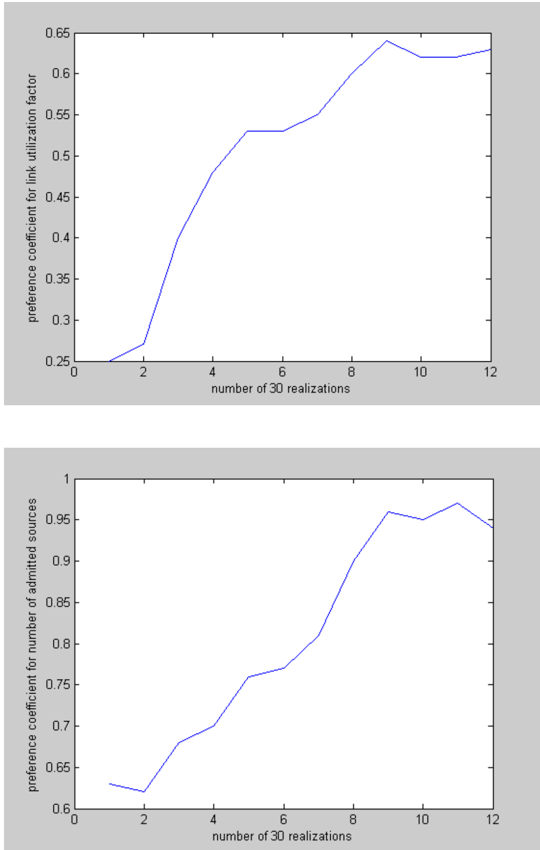

Figure 5 CAC system learning curve

This figure shows that by introducing about 300 new cases to the system it reaches almost to a stable state of performance, which maintains nearly constant thereafter.

4 Conclusions

In this paper we proposed an artificial intelligent CAC system which deals with self similar traffic sources and uses case based reasoning as the intelligent core of the system. The goal of this CAC scheme is to admit as more sources as possible thus maintaining the link utilization factor at a higher level and meanwhile guaranteeing QOS parameter CLR, below a predefined value. We predefined a case base, which is updated during the CAC operation through a learning algorithm. An improvement in system performance was observed using the learning criteria. The results were compared with a classic analytical scheme and better results were gained.

References:

[1] W. E. Leland, M. S. Taqqu, W. Willinger and D. V. Wilson, On the Self-Similar Nature of Ethernet Traffic (extended version), *IEEE Transaction on Networking,* Vol.2, No.1, February 1994, pp.1-15.

[2] M. Cinotti, E.Dalle Mese, S. Giordano, F. Russo, Long Range Dependence in Ethernet Traffic Offered to Interconnected DQDB MANs, *Proc. Of the IEEE ICCS '94 Singapore,* 14-18 Nov. 1994.

[3] S. M. Klivansky, A. Mukherjee, C. Song, On Long Range Dependence in NSFNET Traffic, *Report Georgia Tech GIT-CC-94-61*.

[4] K. Meier-Hellstern, P. Wirth, Y-L. Yan and D. Hoeflin, Traffic Models for ISDN Users: Office Automation Applications, *Proc. ITC 13 Copenhagen*, 1991.

[5] Zafer Sahinoglu and Sirin Tekinay, On Multimedia Networks: Self-Similar Traffic and Network Performance, *IEEE Communications magazine*, Jan 1999, pp.48-52.

[6] M. W. Garrett, W. Willinger, Analysis, Modeling and Generation of Self-Similar VBR Video Traffic, *ACM Sigcomm '94,* Sept. 1994.

[7] J. Beran, R. Sherman, M. S. Taqqu, W. Willinger, Long Range Dependence in Variable Bit Rate Video Traffic, *IEEE Trans. on Communications,* Vol.43, No.2/3/4, 1995, part III pp.1566-1579.

[8] G. R. Garroppo, S. Giordano, M. Pagano, F. Russo, Self Similar Source Modeling of VBR Packet Video Traffic, *7th International Workshop on packet video*, 18-19 March 1996, Brisbane, Australia.

[9] Guerin, R. Ahmadi and H. Naghshineh, Equivalent Capacity and its Application to Bandwidth Allocation in High-Speed Networks, *IEEE J. Sel. Areas Commun*., Vol.9, No. 7, 1991, pp.968-981.

[10] A. I. Elwalid, D. Mitra, Effective Bandwidth of General Markovian Traffic Sources and Admission Control of High Speed Networks, *IEEE/ACM Trans. Netw*., Vol. 1, No. 3, 1993, pp.329-343.

[11] H. Saito, Call Admission Control in an ATM Network Using Upper Bound of Cell Loss Probability, *IEEE Trans. Commun*., Vol. 40, No. 9, 1992, pp.1512-1521.

[12] I. Norros, Studies for a Model for Connectionless Traffic, Based on Fractional Brownian Motion, *Conference on applied probability in engineering, computer and communication sciences*, Paris, June 1993.

[13] A. Hiramatsu, ATM Communications Network Control by Neural Networks, *IEEE Trans. Neural Netw*., Vol. 1, No. 1, 1990, pp.122-130.

[14] R. G. Cheng and C. I. Chang, Neural Network Connection Admission Control for ATM Networks, *IEE Proc., Commun*., Vol.144, No. 2, 1997, pp.93-98.

[15] R. G. Cheng, C. J. Chang and L. I. Lin, A QOS Provisioning Neural Fuzzy Connection Admission Controller for Multimedia High Speed Networks, *IEEE/ACM Trans. Netw*., Vol. 7, No. 1, 1999, pp.111-121.

[16] H. J. Chang, L. F. Lin, S. Y. Lin and R. G. Cheng, Power Spectrum Based Neural Network Connection Admission Control for Multimedia Networks, *IEE Proc. Comm*., Vol. 149, No. 2, April 2002.

[17] V. Catania, G. Ficili, S. Palazzo and D. Panno, Using Fuzzy Logic in ATM Source Traffic Control: Lessons and Perspectives, *IEEE Communications Magazine*, Vol.34, No. 11, Nov. 1996, pp.70-74.

[18] Q. Ren and G. Ramamurthy, A Real Time Dynamic Connection Controller Based on Traffic Modeling, Measurement and Fuzzy Logic Control, *IEEE J. Sel. Areas Commun*., Vol. 18, No. 2, Feb. 2000, pp.184-196.

[19] S. Slade, Case-Based Reasoning: A Research Paradigm, *AI magazine*, Vol. 12, No. 1, Spring 1991, pp.42-55.

[20] Benoit B. Mandelbrot, *The Fractal Geometry of Nature*, Freeman Newyork, 1983.

[21] V. Paxon and S. Floyd, Wide Area Traffic: The Failure of Poisson Modeling, *IEEE/ACM Trans. Netw*. Vol. 3, 1997, pp. 226-244.

[22] P. Pruthi and A. Eramilli , Heavy Tailed ON/OFF Source Behavior and Self Similar Traffic, *Proceeding ICC'95.*

[23] W. C. Lau, A. Eramilli, J. L. Wang and W. Willinger, Self Similar Traffic Generation: The Random Midpoint Displacement Algorithm and its Properties, *ICC'95*, pp. 466-472.

[24] A. Aamodt and E. Plaza, Case-Based Reasoning : Foundational Issues, Methodological Variations, and System, *AI Communications, the European Journal of Artificial intelligence*, 1994, pp.39-59.