行政院國家科學委員會補助專題研究計畫成果報告

※※※※※※※※※※※※※※※※※※※※※※※※※※

※ ※

※ **ATM 網路 ABR 訊務管理及訊務源模型之研究** ※

※ ※

※※※※※※※※※※※※※※※※※※※※※※※※※※

計畫類別：□個別型計畫　　■整合型計畫

計畫編號：NSC89-2213-E-009-107

執行期間：89 年 8 月 1 日至 89 年 7 月 31 日

計畫主持人：楊啟瑞

共同主持人：

計畫參與人員：

本成果報告包括以下應繳交之附件：

□赴國外出差或研習心得報告一份

□赴大陸地區出差或研習心得報告一份

□出席國際學術會議心得報告及發表之論文各一份

□國際合作研究計畫國外研究報告書一份

執行單位：國立交通大學資訊工程研究所

中　華　民　國　89 年　10 月　25 日

行政院國家科學委員會專題研究計畫成果報告

ATM 網路 ABR 訊務管理及訊務源模型之研究

ABR Traffic Management and Source Traffic Modelling

計畫編號：NSC89-2213-E-009-107

執行期限：89 **年** 8 **月** 1 **日至** 89 **年** 7 **月** 31 **日**

主持人：**楊啟瑞**　　**國立交通大學資訊工程研究所**

1. 中文摘要

在這個計畫內，我們針對無線ATM**網路提出了一個新的智慧型多重存取控制系統**(IMACS)**，以支援不同於**CBR**和**VBR**訊務之**ABR**和**Signaling Control (SCR) **訊務。其最大的目地是有效地滿足多種的服務品質** (QoS) **需求並能保持最大的網路產能。**IMACS**由三部份組成**：Multiple Access Controller (MACER)**、** Traffic Estimator/Predictor (TEP)**、和** Intelligent Bandwidth Allocator (IBA)**。計畫的重心是放在訊務源之評估與**ABR**訊務之預測，亦即**IMACS**之**TEP**部份。實驗結果證實採用**TEP**之設計，**IMACS**提供不同之**QoS**保證，且隨訊務源之變化得到最大之網路產能。**

關鍵字:**無線**ATM**網路、多重存取控制、頻寬分配、品質服務、衝撞解決演算法、類神經模糊技術、自我相似訊務源。**

Abstract

In the project, we have proposed a new Intelligent Multiple Access Control System (IMACS) for wireless ATM networks, supporting ABR and Signaling Control (SCR) traffic other than CBR and VBR traffic. It aims to efficiently satisfy their diverse Quality-of-Service (QoS) requirements while retaining maximal network throughput. IMACS is composed of three components: Multiple Access Controller (MACER), Traffic Estimator/Predictor (TEP), and Intelligent Bandwidth Allocator (IBA). The focus of the project is traffic estimation and prediction of ABR traffic, namely the TEP component of IMACS. Simulation results demonstrate that facilitated with TEP, IMACS offers various QoS guarantees and maximizes network throughput irrelevant to traffic variation.

Keywords: Wireless Asynchronous Transfer Mode Networks (WATM), Multiple Access Control (MAC), Bandwidth allocation, Quality-of-Service (QoS), Collision resolution algorithm, Neural-fuzzy technique, Self-similar traffic.

2. Approaches and Results

IMACS is composed of three major components (see Figure 1): Multiple Access Controller (MACER), Traffic Estimator/ Predictor (TEP), and Intelligent Bandwidth Allocator (IBA). It supports four types of traffic- CBR, VBR, ABR, and SCR. IMACS has been designed to satisfy delay guarantees for CBR/VBR traffic while offering minimal access delay for ABR and SCR traffic. Accordingly, MACER employs a reservation-based access protocol for CBR and VBR traffic making use of a fixed amount of R_C-type and R_V-type bandwidth (R_C+R_V= R) (in slots), respectively. By contrast, for SCR and ABR traffic, MACER adopts a contention-based access protocol using C_S-type and C_A-type bandwidth (C_S+C_A= C) (in slots), respectively. In particular, due to the access-delay-sensitive nature, SCR traffic is particularly governed by contention access using the DTS collision resolution algorithm parameterized by the optimal SD, denoted as DTS-*d*, if SD=*d*. The focus of the report is the TEP component of the system.

TEP is responsible for the periodic estimation of the Hurst parameter (denoted as H), and the prediction of the short-term mean and variance of ABR traffic. Specifically, H is periodically estimated based on wavelet analysis [1,2]. The short-term mean and variance for the subsequent frame are predicted by means of an on-line neural-fuzzy approach [3]. Since the prediction of the variance can be similarly applied, in the sequel we describe the estimation of H and prediction of the short-term mean number of active MT's.

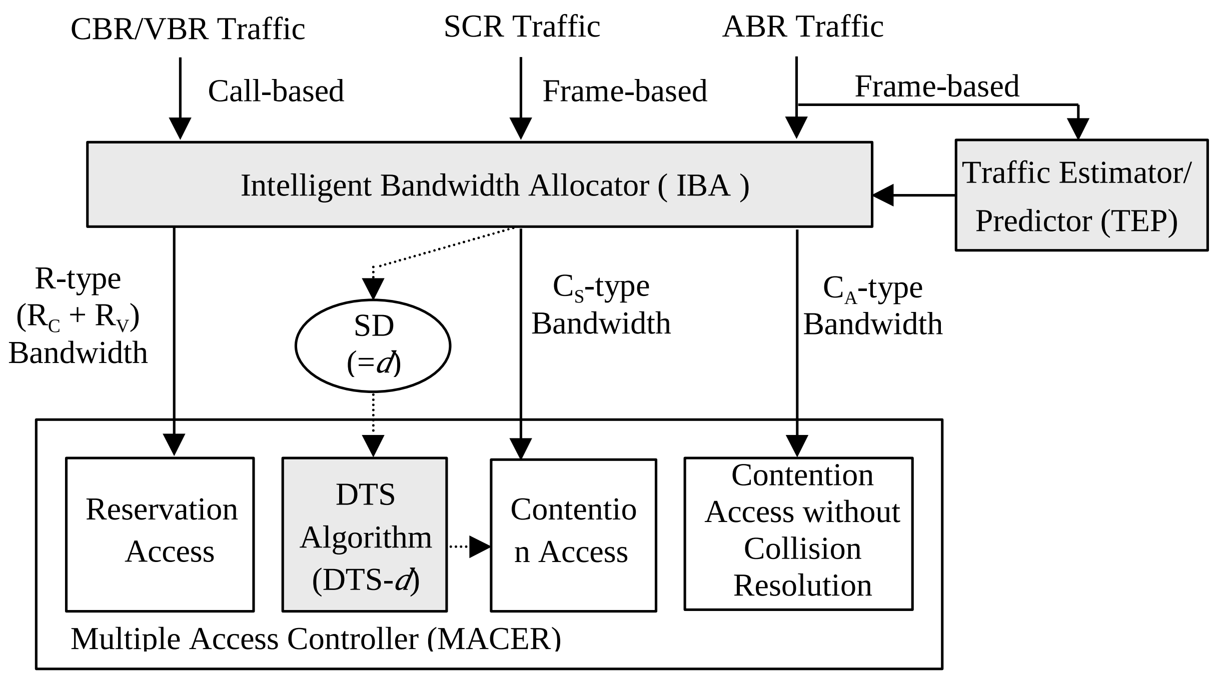

Figure 1. IMACS architecture.

NFTP performs on-line traffic prediction based on a self-constructing neural-fuzzy inference network [3]. It is involved in two phases of learning: structure and parameter learning. The structure-learning phase determines the structure of fuzzy if-then rules, and the parameter-learning phase tunes the coefficients of the rules adapting to the input traffic dynamics. Unlike existing neural-fuzzy models using sequential learning, NFTP performs the structure and parameter learning in parallel. This makes NFTP advantageous for fast on-line prediction.

NFTP is a six-layer network taking on a number of input nodes and one output node, as shown in Figure 2. Initially, there are no rules in the network other than input nodes (layer 1) and an output node (layer 6). Upon receiving on-line training data, the structure-learning process proceeds by dynamically self-constructing fuzzy if-then rules (layer 3) according to an input-output clustering-based space-partitioning algorithm [3]. Once a new rule is generated, the centers and widths of the corresponding set of Gaussian membership functions (layer 2 and layer 5) are assigned. The output of a layer 3 node corresponds to the firing strength of the corresponding fuzzy rule, which is in turn normalized in layer 4. Consequently, the predicted output value, y, is given as:

$$y = \sum_i y_i, \quad i = \text{fuzzy rule index, and}$$

$$\text{Fuzzy rule } i: \underline{\text{If}}\ x_1 \text{ is } A_{i1}\ \underline{\text{and}} \ldots \underline{\text{and}}\ \underline{\text{Then}}\ y_i = f_i m_i$$

where y_i is the contribution of fuzzy rule i to the predicted output value, x_j is the j_{th} input value, A_{ij} is the j_{th} membership function of fuzzy rule i, f_i is the normalized firing strength of fuzzy rule i, and m_i is the center of the membership function in layer 5 connected to fuzzy rule i. Meanwhile, in the parameter-learning process, the centers and widths of input membership functions (layer 2) are dynamically adjusted based on the Least Mean Squares (LMS) algorithm [3], whereas those of output membership functions (layer 5) are tuned using the Back Propagation algorithm [6].

Figure 2 illustrates an NFTP network with three inputs. This network predicts the future CNF value ($\hat{N}_4$), which corresponds to *the mean number of active MT's in the subsequent frame*, based on three input values taken from three most-recent CNF values (denoted as N_i, i=1 to 3). At the end of each frame, in addition to predicting the CNF value of the next frame, NFTP also performs the learning operation described above. This is indicated in Figure 6 by the arrowed link pointing from the CNF of Frame 4 to the NFTP output node.

We experimented on two different NFTP structures using different types of inputs, respectively, via simulation. In the first structure, called *CNF-based NFTP*, the inputs are taken directly from a set of different numbers of past CNF values (N_i), ranging from 4 to 24. In the second structure, referred to as *CNF-correlation-based NFTP*, we adopted *exponential-averaging k-lag correlation* of CNF values as inputs. Specifically, taking an example of NFTP with four inputs x_k, k=1 to 4, at the end of the i_{th} frame, x_k will be set as the k-lag correlation $\hat{C}_i$ defined as: $\hat{C}_i = {}^{a}C_i + (1- {}^{a})\hat{C}_{i-1}$, where $C_i = N_i \times N_{i-k}$, and λ is the smoothing constant ($0 < {}^{a} < 1$). With this structure, we also carried out 4 to 24 different numbers of inputs. In this simulation, we on-line predicted a set of 200 frames, using both structures of NFTP. All parameters used in the simulation are summarized in Table 1. In addition, the performance of NFTP is evaluated in terms of its prediction precision (error rate), time complexity, and space complexity. The error rate was computed as the normalized average deviation between the actual and predicated CNF values. The space complexity was given in terms of the total number of fuzzy rules generated at the end of 200-frame prediction. Notice that, since such inference network can be implemented in hardware, we thus disregarded its time complexity. Simulation results are displayed in Table 2.

We observed during the experiment that the prediction error rate using either structure is irrelevant to the Hurst parameter (H), but highly sensitive to the variance. This can be perceived by the fact that by and large, H manifests only long-term behavior, whereas variance greatly reflects short-term fluctuation. In essence, as shown in Table 2 under traffic H=0.8, the error rate greatly increases with the variance. Furthermore, compared to CNF-based NFTP, CNF-correlation-based NFTP achieves greater precision (lower error rate) and lower space complexity (less number of fuzzy rules).

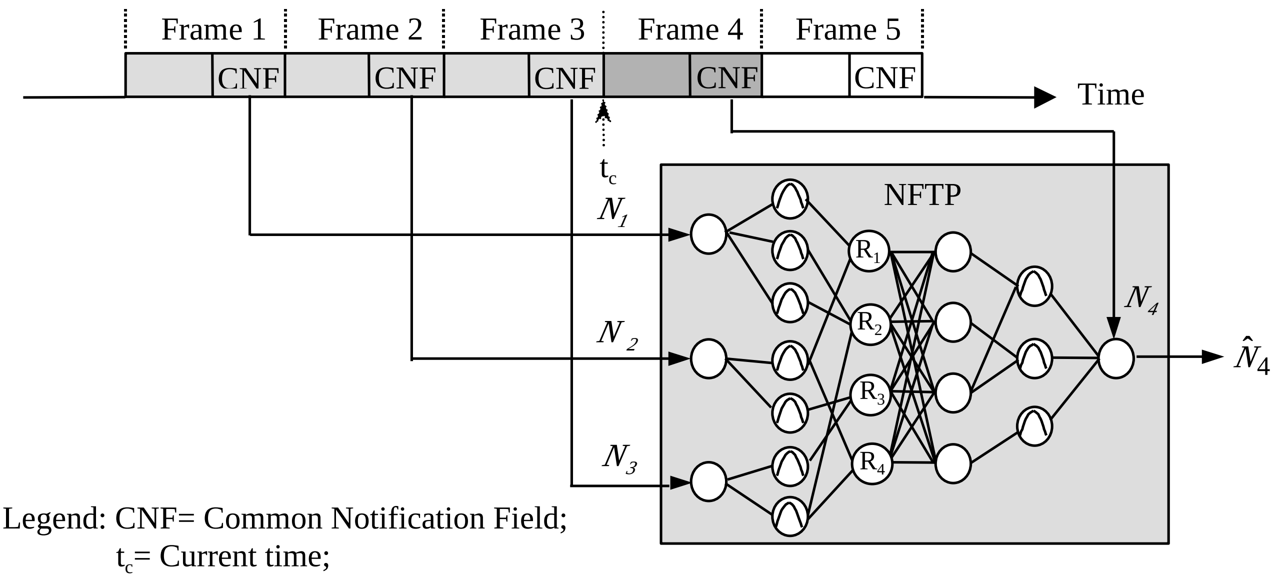

Figure 2. NFTP architecture.

We finally discovered in the table that NFTP (either structure) with 12 inputs invariably exhibits better performance under both variances. Namely, small or large numbers of inputs yield inferior performance for on-line prediction.

Table 1. NFTP parameters used in simulation

Variable	Definition	Value
$\overline{F}_{in}$	Input clustering threshold	0.1
$\overline{F}_{out}$	Output clustering threshold	0.99
°(t)	Membership function threshold	0.7
	Initial width of Gaussion function	0.25
ı	Back-propagation learning constant	0.08
a	Smoothing constant	0.5

Table 2. Performance of NFTP using two different structures

H = 0.8	CNF-based NFTP			CNF-correlation-based NFTP		
	Number of inputs	Number of fuzzy rules	Error rate	Number of inputs	Number of fuzzy rules	Error rate
Mean = 50 Variance = 20	4	27	6.1	4	6	5.5
	8	33	6.0	8	12	5.6
	12	42	5.9	12	23	5.4
	16	47	6.6	16	20	5.5
	20	54	6.7	20	28	5.3
	24	55	7.3	24	23	5.6
Mean = 50 Variance = 60	4	27	10.9	4	11	9.7
	8	33	10.7	8	20	9.5
	12	42	10.6	12	17	9.3
	16	47	11.8	16	20	9.8
	20	54	12.1	20	20	9.7
	24	55	13.1	24	23	9.9

3. Merit Review of the Project

The design and experimental results have been published in IEEE Journal of Selected Area on Communications. Moreover, we have designed several networking control systems making use of the mechanism, which has been presented in various conferences (IEEE ICC'00).

4. References

[1] P. Abry, and D. Veitch, "Wavelet Analysis of Long-Range Dependent Traffic," *IEEE Trans. Inform. Theory*, vol. 44, no. 1, Jan. 1998, pp. 2-15.

[2] S. Giordano, S. Miduri, M. Pagano, F. Russo, and S. Tartarelli, "A Wavelet-based Approach to the Estimation of the Hurst Parameter for Self-similar Data," *Proc. DSP*, 1997, pp. 479-482.

[3] C. Jung, and C. Lin, "An On-line Self-Constructing Neural Fuzzy Inference Network and Its Applications," *IEEE Trans. Fuzzy Systems*, vol. 6, no. 1, Feb. 1998, pp. 12-32.

[4] J. Beran, "Statistical Methods for Data with Long-Range Dependence," *Statistical Science*, 7(4), 1992, pp. 404-427.

[5] J. Beran, R. Sherman, M. Taqqa, and W. Willinger, "Long-Range Dependence in Variable Bit Rate Video Traffic," *IEEE Trans. Comm.*, vol. 43, no. 2/3/4, Feb./Mar./Apr. 1995, pp. 1566-1579.

[6] M. Yuang, P. Tien, and S. Liang, "Intelligent Video Smoother for Multimedia Communications," *IEEE JSAC*, vol. 15, no. 2, Feb. 1997, pp. 136-146.

[7] V. Paxson, "Fast, Approxmate Synthesis of Fractional Gaussian Noise for Generating Self-Similar Network Traffic," *Proc. ACM/SIGCOMM*, 1997, pp. 5-18.

[8] R. Addie, M. Zukerman, and T. Neame, "Broadband Traffic Modeling: Simple Solutions to Hard Problems," *IEEE Comm. Magazine*, vol. 36, no. 8, Aug. 1998, pp. 88-95.

[9] R. Addie, D. Platt, and M. Zukerman, "Performance of a Pi Persistence Protocol Subject to Correlated Gaussian Traffic," *Proc. IEEE INFOCOM*, 1996, pp. 263-270.