Connection Admission Control in ATM Networks Using Survey-Based Type-2 Fuzzy Logic Systems

Qilian Liang, Nilesh N. Karnik, *Associate Member, IEEE*, and Jerry M. Mendel, *Fellow, IEEE*

***Abstract*—This paper presents a connection admission control (CAC) method that uses a type-2 fuzzy logic system (FLS). Type-2 FLSs can handle linguistic uncertainties. The linguistic knowledge about CAC is obtained from 30 computer network experts. A methodology for representing the linguistic knowledge using type-2 membership functions and processing surveys using type-2 FLS is proposed. The type-2 FLS provides soft decision boundaries, whereas a type-1 FLS provides a hard decision boundary. The soft decision boundaries can coordinate the cell loss ratio (CLR) and bandwidth utilization, which is impossible for the hard decision boundary.**

***Index Terms*—Connection admission control, fuzzy logic systems, group decision making, linguistic uncertainties, surveys, type-2 fuzzy sets.**

I. Introduction

ASYNCHRONOUS transfer mode (ATM) is the most promising technology for supporting broadband multimedia communication services. The advantages of ATM networks are the flexibility to accommodate a diverse mixture of traffic which possess different traffic characteristics and quality of service (QoS) requirements. The ATM technique provides an attractive solution to the problem of integrating different types of services, with widely different bit-rates, through common interface and switching fabrics. It is a compromise between packet switching and circuit switching techniques.

A set of traffic control functions must be provided by the ATM network to ensure the QoS of each service and to achieve a high network utilization. The wide range of service characteristics, such as bit rates, burstiness factors, cell delay constraints (latency), cell loss tolerance (accuracy), and priority combined with the need for adaptive, and sometimes real-time services makes the use of traditional control methods very difficult.

Although ATM networks can support a wide variety of transmission rates and provide transmission efficiency by asynchronous multiplexing, a cell might be lost in ATM switches if cells are excessively fed into the networks. In order to avoid this situation, the terminals are required to declare their transmission rates as traffic parameters, e.g., peak cell rate (PCR) and sustainable cell rate (SCR), in advance of transmission. According to these declarations of transmission rates, ATM switches judge whether the required QoS can be achieved. If the QoS can be met without deteriorating those of the existing calls, then the call is admitted, otherwise it is rejected. This traffic control function for an ATM system, called connection admission control (CAC), decides whether to accept or reject a call based upon availablity of capacity required to support its QoS. Thus an estimate of the QoS is required based on monitoring traffic patterns and buffer status, which is important in determining the cell loss probability, cell delay and delay variations.

Taking into consideration factors like the source traffic descriptor, the amount of current network congestion along the path of the incoming call, and QoS requirements of the new and the pre-existing calls is a daunting task for any mathematical model. We believe that the *type of services class* is the most important class. A service is a real-time service, such as voice and video, or a nonreal-time service, such as text data; hence, we only study two descriptors, *total average input rate of real-time voice and video traffic* and *total average input rate of nonreal-time data traffic*.

Chong and Li [5] realized the CAC via a probabilistic burstiness-curve, in which each session connection is defined by the buffer space and transmission bandwidth. Zhang et al. [40] presented a uniform CAC scheme based on a Chernoff bound method that uses a simple novel traffic model requiring only a few parameters. Evans and Everitt [10] focused on the newly-developed CDMA cellular networks and proposed an effective bandwidth-based CAC method.

The decision-making nature of CAC has attracted many researchers to apply FLSs and neural networks to it, e.g., [8], [12], [13]. Fuzzy logic systems (FLSs) are known to represent and numerically manipulate linguistic rules in a natural way and for their ability to handle problems that conventional control theory cannot approach successfully because the latter relies on a valid and accurate model which does not always exist, and, FLSs have also been extensively used in system modeling, e.g., [6], [30], [38].

Chang *et al.* [3], [4] proposed a power-spectrum based neural fuzzy CAC for ATM networks. They constructed a decision hyperplane of the CAC according to the parameters of the power spectrum. They devised rules which used the following type-1 fuzzy sets in antecedents sets: *light load, medium load, and heavy load* and the following type-1 fuzzy sets in consequent sets: *straightly reject, weakly reject, weakly accept, and straightly accept*. All of these rules are based on the knowledge from a single expert; but, words can mean different things to different people [27]. Experts have diverse opinions about the meaning of linguistic labels, and they often provide different consequents for the same antecedents; so, fuzzy rules based

Manuscript received August 17, 1998; revised January 1, 2000 and May 1, 2000.

The authors are with the Signal and Image Processing Institute, Department of Electrical Engineering Systems, University of Southern California, Los Angeles, CA 90089-2564 USA (e-mail: mendel@sipi.usc.edu).

Publisher Item Identifier S 1094-6977(00)09085-4.

on just one expert are partial; they ignore the uncertainties associated with collecting rules from a group of experts.

Uehara and Hirota [33] studied the possibility distribution of cell loss as a function of the number of calls per class, by a fuzzy inference scheme based on the observed data of cell loss ratio (CLR), and obtained the upper bound of CLR. They applied fuzzy inference to estimate the possibility distribution of CLR, which was then a basis for admission control decisions. Mehrvar and Le-Ngoc [25] also proposed a CAC scheme using a type-1 FLS to estimate the level of traffic burstiness; they estimated hurst parameters and used them for CAC in an adaptive environment.

Comparing the existing FLS-based CAC approaches and other approaches, the main difference between them is that FLSs can handle expert knowledge and numerical data in a unified framework, and the FLS-based approach requires less computing complexity.

In this paper, we treat the CAC as a *group decision making problem*, where *group* means experts. In [7], a fusion operator which can combine numerical and linguistic information was introduced to deal with group decision making problems. In [24], three ways to improve the pairwise group decision making based on fuzzy preference relations were proposed, and the authors observed the disadvantages of using type-1 fuzzy sets, and proposed to extend them to type-2 fuzzy sets as their future research directions. In addition, Tanaka and Hosaka [32] observed the difficulties of obtaining appropriate MFs for efficient communication network control, which suggests that type-2 MFs will be a better way to represent the uncertainty in network.

To date, type-2 sets and FLSs have been used in decision making ([2], [37]), solving fuzzy relation equations [34], time-series forecasting [21], MPEG video traffic modeling and classification [22], function approximation [15], time-varying channel equalization [18], control of mobile robots [36], and pre-processing of data [14].

In this paper, we consider the design of a FLS that is based on rules collected by surveying a group of experts. In this situation, two types of uncertainties can arise.

1) Different experts often give different answers to the same question, which results in rules having the same antecedents, but different consequents. Consequently, answers to rule-based questions lead to uncertain consequents.
2) Because words mean different things to different people, and membership functions are associated with words (labels), if we also ask the experts about the membership function parameters (e.g., center, spread), we are likely to get different answers for these parameter values. This results in uncertain membership functions. Consequently, answers to queries about membership functions lead to uncertain antecedents and additional uncertainty about consequents.

In this paper, we show how the above two kinds of uncertainties can be handled in the framework of type-2 FLSs [16], [17].

This paper develops a survey-based CAC method using type-2 FLSs. A type-2 FLS provides a new and powerful framework to represent rule uncertainties. Rule uncertainties occur, as we just explained, due to the use of words and their assciated membership functions. For example, in a type-1 FLS-based CAC method, a typical rule might be: IF the total average input rate of real-time voice and video traffic is *a moderate amount*, AND the total average input rate of the nonreal-time data traffic is *some*, THEN the confidence of accepting the call is *a large amount*. In this case, a type-2 FLS can effectively provide a natural mechanism to represent the vagueness inherent in these *italicized* linguistic labels.

In Section II, we give an overview of the recently developed theory of type-2 FLSs. In Section III, we present a survey-based CAC using type-2 FLSs, and, in Section IV, we present our conclusions.

In this paper, A is a type-1 fuzzy set, and the membership grade of $x \in X$ in A is $\mu_A(x)$, which is a crisp number in $[0, 1]$. A type-2 fuzzy set in X is $\tilde{A}$, and the membership grade of $x \in X$ in $\tilde{A}$ is $\mu_{\tilde{A}}(x)$, which is a type-1 fuzzy set in $[0, 1]$. The elements of the domain of $\mu_{\tilde{A}}(x)$ are called *primary memberships* of x in $\tilde{A}$ and the memberships of the primary memberships in $\mu_{\tilde{A}}(x)$ are called *secondary memberships* of x in $\tilde{A}$. The latter defines the possibilities for the primary membership. $\mu_{\tilde{A}}(x)$, can be represented, for each $x \in X$, as $\mu_{\tilde{A}}(x) = \int_u f_x(u)/u$, $u \in J_x \subseteq [0, 1]$; when the secondary MFs are type-1 interval sets, we call the type-2 set an *interval type-2 set*.

II. Type-2 Fuzzy Logic Systems: A Brief Overview

In a FLS, rule uncertainties can occur due to linguistic or numerical uncertainties in the knowledge used to construct the rules. These uncertainties can be handled by using type-2 fuzzy sets. The concept of a *type-2 fuzzy set* was introduced by Zadeh [39] as an extension of the concept of an ordinary fuzzy set (henceforth called a *type-1 fuzzy set*). A type-2 fuzzy set is characterized by a fuzzy membership function, i.e., the membership value (or membership grade) for each element of this set is a fuzzy set in $[0, 1]$, unlike a type-1 set where the membership grade is a crisp number in $[0, 1]$. Such sets are useful in circumstances, where it is difficult to determine the exact membership function for a fuzzy set; hence, they are useful for incorporating uncertainties.

Fig. 1 shows an example of a type-2 set. The standard deviations of the secondary Gaussians are the same for all x. Intensity of the shading is approximately proportional to secondary membership grades. Darker areas indicate higher secondary memberships. The flat portion from about 4.5 to 5.5 appears because primary memberships cannot be greater than 1 (since primary memberships, themselves, are possible membership values, they have to be in $[0, 1]$) and so the Gaussians have to be "clipped." The domain of the membership grade corresponding to $x = 4$ is also shown. The membership grade for every point is a Gaussian type-1 set contained in $[0, 1]$, we call such a set a "Gaussian type-2 set". When the membership grade for every point is a crisp set, the domain of which is an interval contained in $[0, 1]$, we call such type-2 sets "interval type-2 sets" and their membership grades "interval type-1 sets." Interval type-2 sets are very useful when we have no other knowledge about secondary memberships. Since all the memberships in an interval type-1 set are

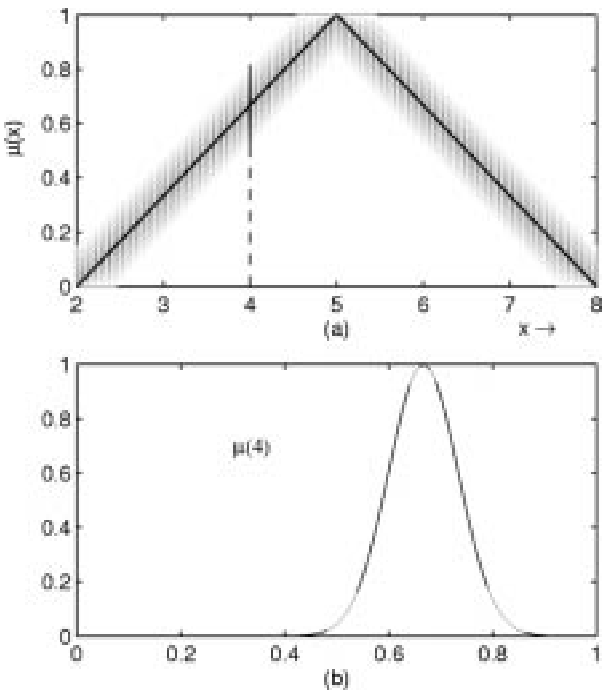

Fig. 1. (a) Pictorial representation of a Gaussian type-2 set. The secondary memberships in this type-1 fuzzy set are shown in (b), and are Gaussian. Note that this set is called a Gaussian type-2 set because all its secondary membership functions are Gaussian. The "principal" membership function (the bold line—see [15], [16] for further discussion), which is triangular in this case, can be of any shape.

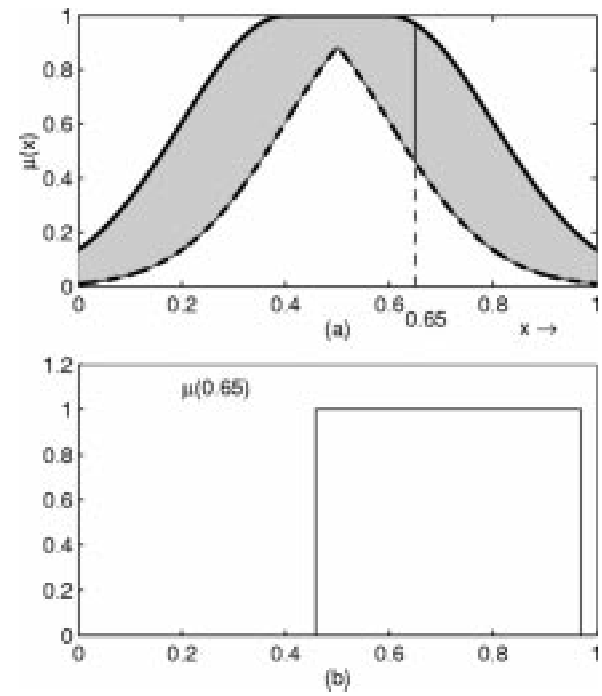

Fig. 2. (a) An interval type-2 set. Since all the secondary memberships are unity, the shading is uniform all over. The domain of the membership grade corresponding to $x = 0.65$ is also shown. The secondary memberships in this type-1 set are shown in (b), and are all equal to 1, i.e., the membership grade is an interval type-1 set.

unity, in the sequel, we represent an interval type-1 set just by its domain interval, which can be represented by its left and right end-points as $[l, r]$. The two end-points are associated with two type-1 MFs that we refer to as *upper* and *lower MFs* [21].

A. Upper and Lower MFs

For convenience in defining the upper and lower MFs of a type-2 MF, we first give the definition of *footprint of uncertainty of a type-2 MF*.

Definition 1 (Footprint of Uncertainty of a Type-2 MF): Uncertainty in the primary membership grades of a type-2 MF consists of a bounded region, that we call the *footprint of uncertainty* of a type-2 MF. It is the union of all primary membership grades.

Definition 2 (Upper and Lower MFs): An upper MF and a lower MF are two type-1 MFs which are bounds for the footprint of uncertainty of an interval type-2 MF. The upper MF is a subset which has the maximum membership grade of the footprint of uncertainty; and, the lower MF is a subset which has the minimum membership grade of the footprint of uncertainty.

For example, in Fig. 2, the upper MF is plotted using a heavy solid line, and the lower MF is plotted using a heavy dashed line. We use an overbar (underbar) to denote the upper (lower) MF. For example, the upper and lower MFs of $\mu_{\tilde{F}_k^l}(x_k)$ are $\bar{\mu}_{\tilde{F}_k^l}(x_k)$ and $\underline{\mu}_{\tilde{F}_k^l}(x_k)$, so that $\mu_{\tilde{F}_k^l}(x_k)$ can be expressed as

$$\mu_{\tilde{F}_k^l}(x_k) = \int_{w^l \in \left[\underline{\mu}_{\tilde{F}_k^l}(x_k), \bar{\mu}_{\tilde{F}_k^l}(x_k)\right]} 1/w^l. \tag{1}$$

B. Operations on Type-2 Sets

Recall that the membership grades of type-2 sets are type-1 sets; therefore, in order to perform operations like union and intersection on type-2 sets, we need to be able to perform t-conorm and t-norm operations between type-1 sets. This is done using Zadeh's Extension Principle [9], [15], [39]. A binary operation $*$ between two crisp numbers can be extended to two type-1 sets $F = \int_v f(v)/v$ and $G = \int_w g(w)/w$, as

$$F * G = \int_v \int_w [f(v) \star g(w)]/(v * w) \tag{2}$$

where $\star$ denotes the chosen t-norm. We will generally use product or minimum t-norm. For example, the extension of the t-conorm (we generally use the maximum t-conorm) operation to type-1 sets is

$$F \sqcup G = \int_v \int_w [f(v) \star g(w)]/(v \vee w). \tag{3}$$

This is called the *join* operation [28]. Similarly, the extension of the t-norm operation to type-1 sets, which is known as the *meet* operation [28], is

$$F \sqcap G = \int_v \int_w [f(v) \star g(w)]/(v \star w). \tag{4}$$

We next show an example of the *meet* operation under product t-norm, when the sets involved are interval type-1 sets.

Example 2.1: Let F and G be two interval type-1 sets with domains $[l_f, r_f]$ and $[l_g, r_g]$, respectively. Using (4), the *meet* between F and G, under product t-norm, can be obtained as

$$F \sqcap G = \int_{v \in F} \int_{w \in G} (1 \times 1)/(vw). \tag{5}$$

Observe, from (5), that 1) each term in $F \sqcap G$ is equal to the product vw for some $v \in F$ and $w \in G$, the smallest term being $l_f l_g$ and the largest $r_f r_g$ and 2) since both F and G have continuous domains, $F \sqcap G$ also has a continuous domain; consequently, $F \sqcap G$ is an interval type-1 set with domain $[l_f l_g, r_f r_g]$, i.e.,

$$F \sqcap G = \int_{u \in [l_f l_g, r_f r_g]} 1/u. \tag{6}$$

In a similar manner, the *meet*, $\sqcap_{i=1}^{n} F_i$, of n interval type-1 sets $F_1, \ldots, F_n$, having domains $[l_1, r_1], \ldots, [l_n, r_n]$, respectively, is an interval set with domain $[\prod_{i=1}^{n} l_i, \prod_{i=1}^{n} r_i]$.

[19] gives a similar result for the multiplication of fuzzy numbers. □

Fast algorithms for computing the *join* and *meet* of type-1 fuzzy sets have also been developed for the cases where the sets involved are not interval type-1 sets (see [15] and [16] for details).

Algebraic operations between type-1 sets are also defined using (2), e.g., the algebraic sum of F and G can be defined as

$$F + G = \int_v \int_w [f(v) \star g(w)]/(v + w). \tag{7}$$

Using the same reasoning as in Example 2.1, it can be shown that when F and G are interval type-1 sets with domains $[l_f, r_f]$ and $[l_g, r_g]$, respectively, their algebraic sum is also an interval type-1 set with domain $[l_f + l_g, r_f + r_g]$ (see [19] for a similar result). More generally, we have the following result for interval type-1 sets.

Theorem 1: Given n interval type-1 sets $F_1, \ldots, F_n$, with means $m_1, m_2, \ldots, m_n$ and spreads $s_1, s_2, \ldots, s_n$, their affine combination $\sum_{i=1}^{n} \alpha_i F_i + \beta$, where α_i $(i = 1, \ldots, n)$ and β are crisp constants, is also an interval type-1 set with mean $\sum_{i=1}^{n} \alpha_i m_i + \beta$, and spread $\sum_{i=1}^{n} |\alpha_i| s_i$. □

See [15], [16] for the proof of Theorem 1.

Observe, from (2) and (4), that, when using product t-norm, the product of F and G is the same as the *meet* of F and G; hence, all our earlier discussions about the *meet* operation under product t-norm apply to the multiplication of type-1 sets under product t-norm.

Using the Extension Principle, an n-ary operation $f(\theta_1, \ldots, \theta_n)$ on crisp numbers can be extended to n type-1 fuzzy sets $F_1, \ldots, F_n$ as [18]

$$\begin{aligned} &f(F_1, \ldots, F_n) \\ &\quad = \int_{\theta_1} \cdots \int_{\theta_n} \mu_{F_1}(\theta_1) \star \cdots \star \mu_{F_n}(\theta_n) / f(\theta_1, \ldots, \theta_n) \end{aligned} \tag{8}$$

where all the integrals denote logical union, and $\theta_i \in F_i$ for $i = 1, \ldots, n$.

We, next, define the concept of the "centroid" of a type-2 set using (8). This concept is required in a type-2 FLS.

Recall that the centroid of a type-1 set A, whose domain is discretized into N points is given as

$$c_A = \frac{\sum_{i=1}^{N} x_i \mu_A(x_i)}{\sum_{i=1}^{N} \mu_A(x_i)}. \tag{9}$$

Similarly, the centroid of a type-2 set $\tilde{A}$, whose domain is discretized into N points, can be defined using (8) as follows. If we let $D_i = \mu_{\tilde{A}}(x_i)$, then

$$C_{\tilde{A}} = \int_{\theta_1} \cdots \int_{\theta_N} [\mu_{D_1}(\theta_1) \star \cdots \star \mu_{D_N}(\theta_N)] \Big/ \frac{\sum_{i=1}^{N} x_i \theta_i}{\sum_{i=1}^{N} \theta_i} \tag{10}$$

where $\theta_i \in D_i$, and all the integrals denote logical union.

Equation (10) can be described in words as follows. Each point x_i of $\tilde{A}$ has a type-1 fuzzy membership grade, $D_i = \mu_{\tilde{A}}(x_i)$, associated with it. To find the centroid, we consider every possible combination $\{\theta_1, \ldots, \theta_N\}$ such that $\theta_i \in D_i$. For every such combination, we perform the type-1 centroid calculation in (9) by using θ_is in place of $\mu_{\tilde{A}}(x_i)$s; and, to each point in the centroid, we assign a membership grade equal to the t-norm of the membership grades of the θ_is in the D_is. If more than one combination of θ_is gives us the same point in the centroid, we keep the one with the largest membership grade.

Every combination, $\{\theta_1, \ldots, \theta_N\}$ $(\theta_i \in D_i)$, considered when computing $C_{\tilde{A}}$, can be thought to form the membership function of some type-1 set A' which has the same domain as $\tilde{A}$. We call A' an *embedded type-1 set* in $\tilde{A}$ [see Fig. 3(a) for two examples of embedded type-1 sets]. Every embedded type-1 set also has a weight associated with it, which is calculated as the t-norm of the secondary memberships corresponding to $\{\theta_1, \ldots, \theta_N\}$ that make up that embedded set. The type-2 set $\tilde{A}$ can, therefore, be thought of as a large collection of embedded type-1 sets, each having a weight associated with it, and its centroid $C_{\tilde{A}}$ can be thought of as a type-1 set whose elements are the centroids of all the embedded type-1 sets in $\tilde{A}$, and their memberships are the weights associated with the corresponding embedded sets. The centroid computation simplifies considerably when $\tilde{A}$ is an interval type-2 set, as we show next.

If $\tilde{A}$ is an interval type-2 set, (10) simplifies to

$$C_{\tilde{A}} = \int_{\theta_1} \cdots \int_{\theta_N} 1 \Big/ \frac{\sum_{i=1}^{N} x_i \theta_i}{\sum_{i=1}^{N} \theta_i} \tag{11}$$

where each θ_i belongs to some interval in $[0, 1]$. We present a procedure for computing (11) in the Appendix .

Example 2.2: Consider the interval type-2 set in Fig. 2, shown again in Fig. 3(a). Using the Appendix computational procedure, we find that $C_{\tilde{A}}$ is an interval type-1 set with domain $[0.398\,55, 0.601\,45]$. As explained in Appendix , only two sets of computations are needed to obtain $C_{\tilde{A}}$, one each for its left and right end-points. □

See [17], [15], [18] for more discussions about the centroid of a type-2 set.

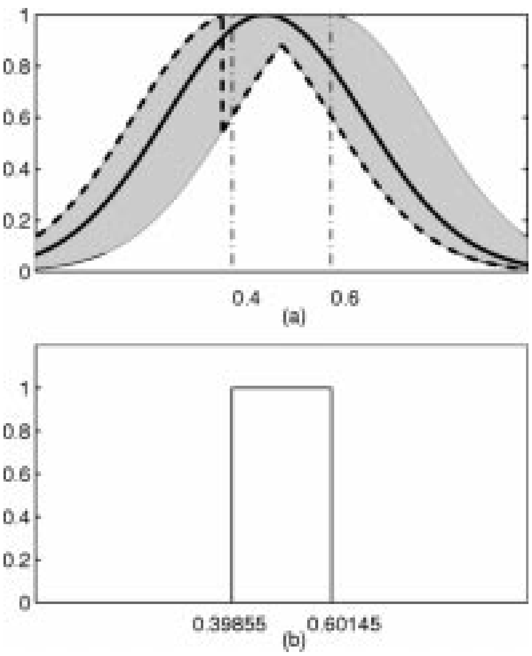

Fig. 3. (a) The interval type-2 set shown in Fig. 2. Two embedded type-1 sets are also shown, one with a thick dashed line and the other with a thick solid line. (b) Centroid of the type-2 set in Fig. (a), computed using the computational procedure described in the Appendix .

C. Type-2 Fuzzy Logic Systems

Fig. 4 shows the structure of a type-2 fuzzy logic system (FLS). It is very similar to the structure of a type-1 FLS [26]. For a type-1 FLS, the *output processing* block only contains the defuzzifier. When an input is applied to a type-1 FLS, the inference engine computes the type-1 output set corresponding to each rule. The defuzzifer then computes a crisp output from these rule output sets. For a type-2 FLS, the antecedent and/or consequent sets are type-2, so that each rule output set is type-2. "Extended" versions of type-1 defuzzification methods [obtained using (8)] yield a type-1 set from the type-2 rule output sets. We call this process *type-reduction* rather than defuzzification, and the resulting type-1 set, the *type-reduced set*. The defuzzifier in the type-2 FLS can, then, defuzzify the type-reduced set to obtain a crisp output for the type-2 FLS. Output processing is depicted pictorially in Fig. 5. The fuzzifier maps the crisp input into a fuzzy set. This fuzzy set can, in general, be a type-2 set; however, in this paper, we consider only *singleton* fuzzification, for which the input fuzzy set has only a single point of nonzero membership.

To see the difference between a type-1 FLS and type-2 FLS, we first review a type-1 FLS.

1) Type-1 FLS: Consider a p-input 1-output type-1 FLS, using singleton fuzzification, *center-of-sets* defuzzification [15], [17], [31] and "IF-THEN" rules of the form [23]

$$R^l\text{:IF } x_1 \text{ is } F_1^l \text{ and } x_2 \text{ is } F_2^l \text{ and } \cdots \text{ and } x_p \text{ is } F_p^l, \quad \text{THEN } y \text{ is } G^l. \tag{12}$$

Assuming singleton fuzzification, when an input $\mathbf{x}' = \{x_1', \ldots, x_p'\}$ is applied, the degree of firing corresponding to

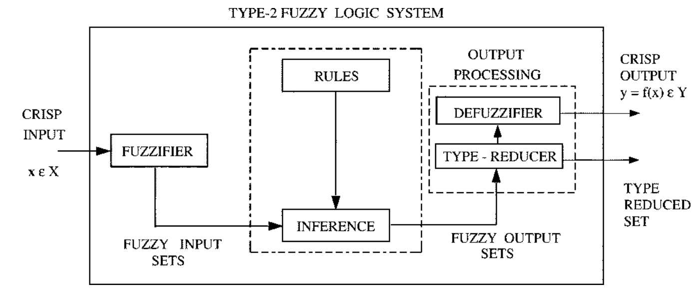

Fig. 4. Structure of a type-2 FLS. In order to emphasize the importance of the type-reduced set, we have shown two outputs for the type-2 FLS, the type-reduced set and the crisp defuzzified value.

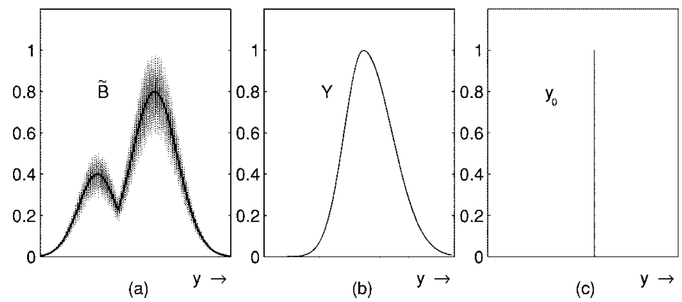

Fig. 5. Pictorial representation of the *output processing* in a type-2 FLS. For an applied input $\mathbf{x}$, the type-reducer first combines the individual rule output sets in some manner to obtain (a) combined output set $\tilde{B}$, and then, from $\tilde{B}$ (b) creates a type-1 set, Y, which we call the type-reduced set. (c) Defuzzifier produces a crisp output, y_0 from the type-reduced set.

the lth rule is computed as

$$\mu_{F_1^l}(x_1') \star \mu_{F_2^l}(x_2') \star \cdots \star \mu_{F_p^l}(x_p') = \mathcal{T}_{i=1}^{p} \mu_{F_i^l}(x_i') \tag{13}$$

where $*$ and $\mathcal{T}$ both indicate the chosen t-norm. There are many kinds of defuzzifiers. In this paper, we focus, for illustrative purposes, on the center-of-sets defuzzifier [31]. It computes a crisp output for the FLS by first computing the centroid, c_{G^l}, of every consequent set G^l, and, then computing a weighted average of these centroids. The weight corresponding to the lth rule consequent centroid is the degree of firing associated with the lth rule, $\mathcal{T}_{i=1}^{p} \mu_{F_i^l}(x_i')$ [see (13)], so that

$$y_{\cos}(\mathbf{x}') = \frac{\sum_{l=1}^{M} c_{G^l} \mathcal{T}_{i=1}^{p} \mu_{F_i^l}(x_i')}{\sum_{l=1}^{M} \mathcal{T}_{i=1}^{p} \mu_{F_i^l}(x_i')} \tag{14}$$

where M is the number of rules in the FLS.

2) Type-2 FLS: Now, consider a p-input 1-output type-2 FLS, using singleton fuzzification, *center-of-sets* type-reduction [15], [17] and rules of the form

$$R^l\text{:IF } x_1 \text{ is } \tilde{F}_1^l \text{ and } x_2 \text{ is } \tilde{F}_2^l \text{ and } \cdots \text{ and } x_p \text{ is } \tilde{F}_p^l, \quad \text{THEN } y \text{ is } \tilde{G}^l. \tag{15}$$

Note that, although we have shown all the antecedent and consequent sets to be type-2 in (15), it need not necessarily be so. Even if only one of the antecedents or the consequent is type-2, the FLS is type-2. When an input $\mathbf{x}' = \{x_1', \ldots, x_p'\}$ is applied, the inference engine computes the degree of firing of each rule by performing the *meet* operation (4) between the antecedent membership grades of each rule. The degree of firing corresponding to the lth rule is [see (13)]

$$\mu_{\tilde{F}_1^l}(x'_1) \sqcap \mu_{\tilde{F}_2^l}(x'_2) \sqcap \cdots \sqcap \mu_{\tilde{F}_p^l}(x'_p) = \sqcap_{i=1}^{p} \mu_{\tilde{F}_i^l}(x'_i). \quad (16)$$

The center-of-sets type-reducer, then, requires the centroid of each rule consequent. Centroid computations are done as explained in Section II-B. Once all the consequent centroids are computed, the center-of-sets type-reduced set is computed by using the extended version of (14) as follows [15], [17]:

$$Y_{\cos}(\mathbf{x}') = \int_{y^1} \cdots \int_{y^M} \int_{f_1} \cdots \int_{f_M} \mathcal{T}_{l=1}^{M} \mu_{C^l}(y^l) \star \mathcal{T}_{l=1}^{M} \mu_{F^l}(f_l) \Bigg/ \frac{\sum_{l=1}^{M} y_l f_l}{\sum_{l=1}^{M} f_l} \quad (17)$$

where $\mathcal{T}$ and $\star$ indicate the chosen t-norm; $y^l \in C^l = C_{\tilde{G}^l}$, the centroid of the lth consequent set; and, $f_l \in F^l = \sqcap_{i=1}^{p} \mu_{\tilde{F}_i^l}(x'_i)$, the degree of firing associated with the lth consequent set, for $l = 1, \ldots, M$.

A crisp output for the FLS is obtained by finding the centroid of $Y_{\cos}(\mathbf{x}')$. This is the defuzzifier step shown in Fig. 4.

The type-reduced set of a type-2 FLS shows the possible variation in the crisp output of the FLS due to uncertain natures of the antecedents and/or consequents. It establishes a band of values around a crisp output value in much the same way that a confidence interval establishes a band about a point estimate when stochastic uncertainty is present; but, it does this for linguistic uncertainties.

A general type-2 FLS has a high computation complexity, but things simplify a lot when the secondary membership degree is an interval. In this paper, we use an interval type-2 FLS for CAC.

The theory and design of an interval type-2 FLS is given in [21]. Here we briefly summarize the results for computing the firing interval when a singleton fuzzifier is used.

Theorem 2: In an interval type-2 FLS with meet under minimum or product t-norm, the firing interval $F^l \triangleq [\underline{f}^l, \bar{f}^l]$ of the lth rule is

$$\underline{f}^l = \underline{\mu}_{\tilde{F}_1^l}(x_1) \star \cdots \star \underline{\mu}_{\tilde{F}_p^l}(x_p) \quad (18)$$

and

$$\bar{f}^l = \bar{\mu}_{\tilde{F}_1^l}(x_1) \star \cdots \star \bar{\mu}_{\tilde{F}_p^l}(x_p). \quad (19)$$

The proof of this Theorem is given in [21].

For an interval type-2 FLS, (17) reduces to

$$Y_{\cos}(\mathbf{x}') = \int_{y^1} \cdots \int_{y^M} \int_{f_1} \cdots \int_{f_M} 1 \Bigg/ \frac{\sum_{l=1}^{M} y^l f_l}{\sum_{l=1}^{M} f_l} = [y_l, y_r] \quad (20)$$

where $y^l \in C^l$ (centroid of the consequent set), and $f_l \in [\underline{f}^l, \bar{f}^l]$. These computations are not difficult. y_l and y_r can be computed using the procedure presented in the Appendix .

III. Survey-Based CAC Using Type-2 FLSs

We apply type-2 FLSs to CAC for ATM networks, in which the type-2 rules are based on a survey regarding the CAC as determined by the input traffic. We chose a type-2 FLS for CAC to give ATM network designers more room to accomodate their own thoughts and preferences, and let their decisions be more flexible, since requirements about cell loss ratio (CLR) and bandwidth untilization cannot be mutually satisfied. Lower CLR and higher bandwidth utilization are the desired performance of CAC; but, for a fixed bandwidth allocation scheme, lower CLR means less bandwith utilization, and higher bandwidth untilization means higher CLR. A type-1 FLS provides a single decision boundary for CAC, which means a compromise decision has to be made with respect to CLR and bandwidth untilization. On the other hand, as we demonstrate below, a type-2 FLS provides a region bounded by two decision boundaries, so that designers are free to choose a decision boundary to meet their preferences, e.g., higher bandwidth utilization.

Designing a survey-based type-2 FLS includes collecting the knowledge, setting the rules, choosing and defining antecedent and consequent membership functions, choosing type-reduction, and extracting decision boundaries.

A. Extracting the Knowledge for CAC

In ATM networks, input traffic is often classified into two classes. Class 1 is real-time voice and video traffic, and Class 2 is nonreal-time data traffic. We, therefore, used two antecedents for our FLS-based CAC—the total average input rate of real-time voice and video traffic, and the total average input rate of nonreal-time data traffic. The linguistic variables used to represent the input rate of traffic were divided into five levels: *none to very little* (NVL), *some* (S), *a moderate amount* (MOA), *a large amount* (LA), and *a maximum amount* (MAA). The consequent—the confidence of accepting the call—was also divided into these same five levels. We used rules obtained from the knowledge of many network experts (30 USC electrical engineering Ph.D. students who have studied EE555 *broadband network architecture*). We surveyed these experts using questions such as:

> IF the total average input rate of real-time voice and video traffic is *a moderate amount*, and the total average input rate of the nonreal-time data traffic is *a large amount*, THEN the confidence of accepting the call is ————.

These experts were requested to choose a consequent using one of the five linguistic variables. Different experts gave different answers to the questions in the survey.

As pointed out in [27], "words mean different things to different people," and in [24], "the decision makers may have the same preferences to a particular alternative, e.g., highly preferred but with different degrees," so, we created two different kinds of surveys for the network experts. The first survey asked the experts to locate each linguistic label in the interval $[0, 10]$ domain. We randomized the five labels, as shown in Table I, so that they are uncorrelated. For each linguistic label, we got 30 intervals from the 30 experts, and we then computed the mean and std of the label's two end-points. We summarize the survey results in Table II and Fig. 6. Observe, in Fig. 6, that overlap between *some* and *none to very little* and between *a moderate amount* and *a large amount* only occurs due to consideration of uncertainties.

The second survey is the CAC technical survey. Table III summarizes the questions used in this survey (the questions were

TABLE I
SURVEY TABLE: RANDOMIZED LABELS

Linguistic label	Start	End
a maximum amount		
some		
none to very little		
a large amount		
a moderate amount		

TABLE II
PROCESSED SURVEY RESULTS: ORDERED LABELS

No.	Range Label	Means		Standard Deviations	
		Start (a)	End (b)	Start (σ_a)	End (σ_b)
1	none to very little	0	1.9850	0	0.8104
2	some	2.5433	5.2500	0.9066	1.3693
3	a moderate amount	3.6433	6.4567	0.8842	0.8557
4	a large amount	6.4833	8.7500	0.7484	0.5981
5	a maximum amount	8.5500	10	0.7468	0

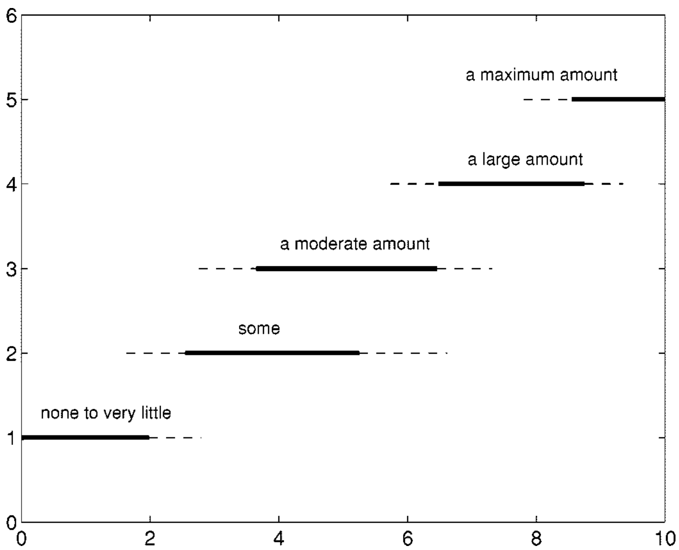

Fig. 6. All five labels, their intervals and uncertainty bands (dashed lines).

randomized in the actual survey, but are shown in their natural order for the convenience of the reader). Thirty respondents completed the survey, and their results are shown in Table IV.

B. Representing the Linguistic Labels Using MFs

We used trapezoidal MFs to represent *none to very little* and *a maximum amount*, and triangle MFs to represent *some*, *a moderate amount*, and *a large amount* (see Fig. 7).

For the linguistic labels *some*, *a moderate amount*, and *a large amount*, the mean values of their interval end-points are a and b, and, the standard deviation (std) of their left end-point is σ_a, and, the std of their right end-point is σ_b. The three break points of a triangle type-1 MF were then located at $(a-\sigma_a, 0)$, $((a+b)/2, 1)$, and $(b+\sigma_b, 0)$. For the linguistic labels *none to very little* and *a maximum amount*, the four break points in their trapzoidal type-1 MFs were located at $(a-\sigma_a, 0)$, $(a, 1)$, $(b, 1)$, and $(b+\sigma_b, 0)$. We show these type-1 MFs in Fig. 7 as heavy dashed lines.

There are uncertainties associated with the break points (i.e., $a-\sigma_a$ and $b+\sigma_b$) of triangle MFs, e.g., why not use $a-0.5\sigma_a$ or $a-2\sigma_a$ instead of $a-\sigma_a$. These uncertainties cannot be captured using type-1 fuzzy sets; however, they can be by using

TABLE III
QUESTIONS FOR CAC IN ATM NETWORKS. ANTECEDENT 1 IS *THE TOTAL AVERAGE INPUT RATE OF REAL-TIME VOICE AND VIDEO TRAFFIC*, ANTECEDENT 2 IS *THE TOTAL AVERAGE INPUT RATE OF NON-REAL-TIME DATA TRAFFIC*, AND CONSEQUENT IS *THE CONFIDENCE OF ACCEPTING THE CALL*. THE EXPERTS WERE ASKED TO FILL IN THE BLANK FOR THE CONSEQUENT USING ONE OF FIVE LINGUISTIC LABELS. THEY WERE GIVEN A RANDOMIZED VERSION OF THESE 25 QUESTIONS

Question Number	Antecedent 1	Antecedent 2	Consequent
1	*none to very little*	*none to very little*	
2	*none to very little*	*some*	
3	*none to very little*	*a moderate amount*	
4	*none to very little*	*a large amount*	
5	*none to very little*	*a maximum amount*	
6	*some*	*none to very little*	
7	*some*	*some*	
8	*some*	*a moderate amount*	
9	*some*	*a large amount*	
10	*some*	*a maximum amount*	
11	*a moderate amount*	*none to very little*	
12	*a moderate amount*	*some*	
13	*a moderate amount*	*a moderate amount*	
14	*a moderate amount*	*a large amount*	
15	*a moderate amount*	*a maximum amount*	
16	*a large amount*	*none to very little*	
17	*a large amount*	*some*	
18	*a large amount*	*a moderate amount*	
19	*a large amount*	*a large amount*	
20	*a large amount*	*a maximum amount*	
21	*a maximum amount*	*none to very little*	
22	*a maximum amount*	*some*	
23	*a maximum amount*	*a moderate amount*	
24	*a maximum amount*	*a large amount*	
25	*a maximum amount*	*a maximum amount*	

TABLE IV
HISTOGRAMS OF EXPERT RESPONSES ABOUT CAC. 30 NETWORK EXPERTS ANSWERED THE QUESTIONS. NVL STANDS FOR *NONE TO VERY LITTLE*; S STANDS FOR *SOME*, MOA STANDS FOR *A MODERATE AMOUNT*, LA STANDS FOR *A LARGE AMOUNT*, AND MAA STANDS FOR *A MAXIMUM AMOUNT*. THE ENTRIES IN THE SECOND—SIXTH COLUMNS CORRESPOND TO THE WEIGHTS $w_1^l, \ldots, w_5^l$, RESPECTIVELY

Rule Number (l)	NVL	S	MOA	LA	MAA	c_{avg}^l	C_{avg}^l
1	0	1	0	0	29	8.9331	[8.7411, 9.1224]
2	0	1	2	5	22	8.4066	[8.2010, 8.6101]
3	0	0	1	6	23	8.6591	[8.4622, 8.8539]
4	0	0	5	15	10	7.6572	[7.4346, 7.8790]
5	0	1	13	10	6	6.6616	[6.4112, 6.9120]
6	0	1	6	13	10	7.4558	[7.2258, 7.6850]
7	0	1	11	10	8	6.9324	[6.6889, 7.1755]
8	0	2	14	12	2	6.2556	[5.9926, 6.5190]
9	0	4	11	15	0	6.1715	[5.9022, 6.4415]
10	1	9	11	9	0	5.3726	[5.0784, 5.6680]
11	0	0	14	12	4	6.5923	[6.3422, 6.8425]
12	0	1	12	15	2	6.5412	[6.2877, 6.7950]
13	0	5	12	13	0	5.9701	[5.6934, 6.2475]
14	2	7	13	8	0	5.2258	[4.9383, 5.5143]
15	5	18	7	0	0	3.8033	[3.4748, 4.1335]
16	0	9	9	12	0	5.7539	[5.4632, 6.0456]
17	3	11	6	10	0	5.1336	[4.8417, 5.4267]
18	2	8	15	5	0	4.9401	[4.6432, 5.2383]
19	8	13	7	3	0	4.0030	[3.6951, 4.3122]
20	8	19	3	0	0	3.3844	[3.0620, 3.7086]
21	1	15	9	5	0	4.8379	[4.5172, 5.1601]
22	6	12	7	5	0	4.2936	[3.9968, 4.5918]
23	8	18	4	0	0	3.4174	[3.0979, 3.7386]
24	12	16	2	0	0	2.9688	[2.6672, 3.2720]
25	26	3	1	0	0	1.5967	[1.3756, 1.8186]

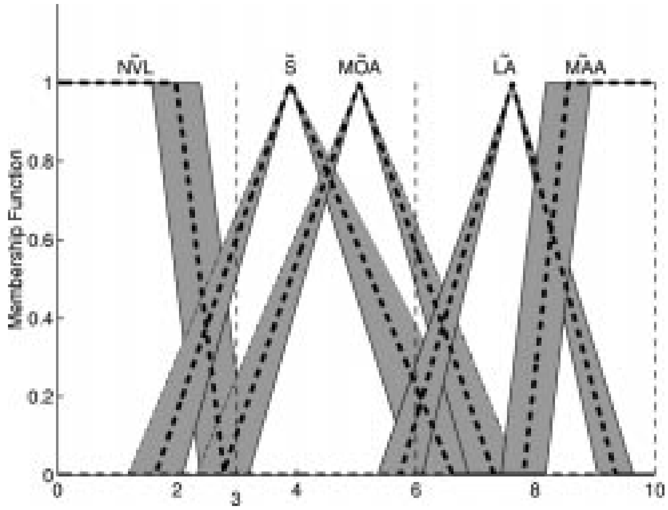

Fig. 7. Type-2 MFs used to represent the five linguistic labels. The footprints of uncertainty are shown shaded, and the heavy dashed lines denote the type-1 MFs used to represent the five linguistic labels.

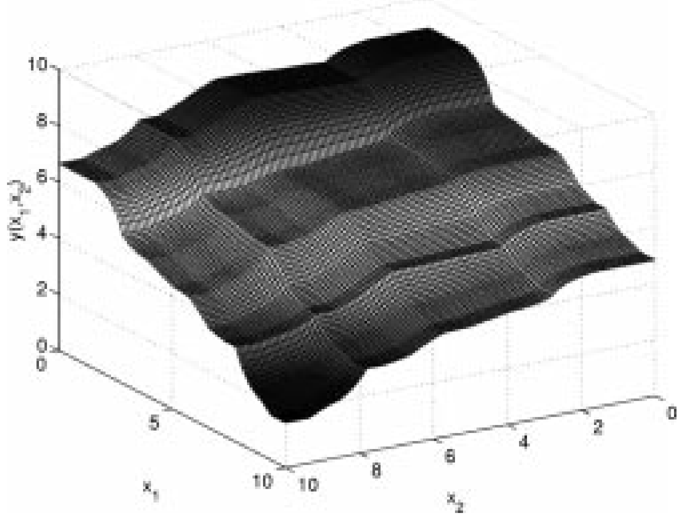

Fig. 8. Type-1 FLS for CAC. The confidence of accepting the call, $y(x_1, x_2)$, versus x_1 and x_2.

type-2 fuzzy sets. In type-2 MFs, the footprints of uncertainty are obtained by specifying upper and lower MFs for each fuzzy set. Let ρ denote *a fraction of uncertainty*, i.e., $0 \le \rho \le 1$. Then we construct the footprints of uncertainty as follows.

- For the triangle MFs with uncertain break points, the break points of the upper MF are $(a - (1+\rho)\sigma_a, 0)$, $((a+b)/2, 1)$, and $(b + (1+\rho)\sigma_b, 0)$, and the break points in lower MFs is $(a - (1-\rho)\sigma_a, 0)$, $((a+b)/2, 1)$, and $(b + (1-\rho)\sigma_b, 0)$.
- For the trapzoidal MFs with uncertain break points, the break points in upper MF are $(a - (1+\rho)\sigma_a, 0)$, $(a - \rho\sigma_a, 1)$, $(b+\rho\sigma_b, 1)$, and $(b+(1+\rho)\sigma_b, 0)$, and the break points in lower MFs is $(a - (1-\rho)\sigma_a, 0)$, $(a+\rho\sigma_a, 1)$, $(b - \rho\sigma_b, 1)$, and $(b + (1-\rho)\sigma_b, 0)$.

Fig. 7 depicts the footprint of uncertainty for $\rho = 0.5$. In this paper, we use $\rho = 0.5$ to illustrate our design of a type-2 FLS and CAC decision boundaries.

C. Survey Processing Using Type-1 FLS

In our approach to forming a type-1 rule base, we chose a single consequent for each rule (for discussions on other ways to use the consequent data, see [15]). To do this, we averaged the centroids of all the responses for each rule and used this average in place of the rule consequent centroid. Doing this leads to rules that have the following form:

> R^l: IF *the total average input rate of real-time voice and video traffic* (x_1) is F^i and *the total average input rate of the nonreal-time data traffic* (x_2) is F^j, THEN the confidence of accepting the call (y) is c^l_{avg}

where

$$c^l_{\text{avg}} = \frac{\sum_{i=1}^{5} w_i^l c^i}{\sum_{i=1}^{5} w_i^l} \tag{21}$$

in which w_i^l is the number of people choosing linguistic label i for the consequent of rule l $(i = 1, \ldots, 5; l = 1, \ldots, 25)$ (see Table IV); and, c^i is the centroid of the ith consequent set $(i = 1, \ldots, 5)$. The centroids of the five type-1 sets depicted in Fig. 7 are $c^1 = 1.1811$, $c^2 = 4.0507$, $c^3 = 5.0402$, $c^4 = 7.5666$, and $c^5 = 9.1015$. To illustrate the use of (21), note, for example, that

$$c^6_{\text{avg}} = \frac{c^2 + 6c^3 + 13c^4 + 10c^5}{1+6+13+10} = 7.4558. \tag{22}$$

All 25 c^l_{avg} values are listed in Table IV.

For every input (x_1, x_2), the output is computed using (14), with c^l_{avg} replacing c_{G^l}. For example (see Fig. 7), when the input is $\mathbf{x} = [3, 6]$, two subsets (the second and third) are fired for $x_1 = 3$, and their firing degrees are 0.6032 and 0.1052, respectively, and, three subsets (the second, third, and fourth) are fired for $x_2 = 6$, and their firing degrees are 0.2275, 0.5801, and 0.1409. Consequently (as determined from Table III), rules 7, 8, 9, 12, 13, and 14 are fired, with firing degrees 0.1372 (0.6032×0.2275), 0.3499, 0.0850 0.0239, 0.0610, and 0.0148, respectively. The defuzzified output is [using (14)] $y(3, 6) = 6.3447$.

By repeating these calculations for $\forall x_i \in [0, 10]$, we obtain a hypersurface $y(x_1, x_2)$, as plotted in Fig. 8. Observe that $y(x_1, x_2)$ is a monotonically decreasing function, because with the increase of *the total average input rate of real-time voice and video traffic* or *the total average input rate of nonreal-time data traffic*, *the confidence of accepting the call* decreases.

D. Survey Processing Using Type-2 FLS

Here, we adopt a similar strategy to the one in Section III-C. Because all membership functions are now type-2, we create a type-2 FLS, which has rules of the form

> R^l: IF *the total average input rate of real-time voice and video traffic* (x_1) is $\tilde{F}^i$ and IF *the total average input rate of the nonreal-time data traffic* (x_2) is $\tilde{F}^j$, THEN the confidence of accepting the call (y) is C^l_{avg}.

where

$$C^l_{\text{avg}} = \frac{\sum_{i=1}^{5} w_i^l C^i}{\sum_{i=1}^{5} w_i^l} \tag{23}$$

$\sum$ denotes algebraic sum, C^i is the centroid of $\tilde{F}^i$ $(i = 1,\ldots,5)$, and w_i^l is the weight associated with the ith consequent for the lth rule (see Table IV). The centroids of the five type-2 sets [computed using (11)] are $C^1 = [0.9808, 1.3820], C^2 = [3.6717, 4.4320], C^3 = [4.7501, 5.3314]$, $C^4 = [7.3418, 7.7915]$, and $C^5 = [8.9159, 9.2842]$. Since each C^l is an interval type-1 set, (23) can be computed using Theorem 1. For example, for rule 9, from Table IV the number of people choosing the five linguistic labels are 0, 4, 11, 15, 0. So we calculate $C_{\rm avg}^9$ as

$$\begin{aligned} C_{\rm avg}^9 &= \frac{4\times C^2 + 11\times C^3 + 15\times C^4}{4+11+15} \\ &= [5.9022, 6.4415]. \end{aligned} \tag{24}$$

All $C_{\rm avg}^l$ are listed in Table IV.

We show the calculation of the type-reduced set for $\mathbf{x} = [3,6]$. For $x_1 = 3$ (see Fig. 7), three subsets are fired ($\tilde{F}_1 = \tilde{\rm NVL}$, $\tilde{F}_2 = \tilde{S}$, and $\tilde{F}_3 = \tilde{\rm MOA}$), and their firing degrees are $[0, 0.2475], [0.5037, 0.6695]$, and $[0, 0.2499]$, respectively; for $x_2 = 6$, three subsets are also fired ($\tilde{F}_2$, $\tilde{F}_3$, and $\tilde{F}_4 = \tilde{\rm LA}$), and their firing degrees are $[0, 0.3827], [0.4822, 0.6469]$, and $[0, 0.2834]$. Consequently, nine rules are fired whose antecedent pairs are

$$(\tilde{F}_1,\tilde{F}_2),(\tilde{F}_1,\tilde{F}_3),(\tilde{F}_1,\tilde{F}_4),(\tilde{F}_2,\tilde{F}_2),(\tilde{F}_2,\tilde{F}_3),$$
$$(\tilde{F}_2,\tilde{F}_4),(\tilde{F}_3,\tilde{F}_2),(\tilde{F}_3,\tilde{F}_3),\ \text{and}\ (\tilde{F}_3,\tilde{F}_4)$$

i.e., rules 2, 3, 4, 7, 8, 9, 12, 13, 14 are fired (from Table III), with firing intervals computed using (18) and (19). For example, the firing interval $F^8 = [\underline{f}^8, \bar{f}^8]$, for R^8 is

$$\begin{aligned} \underline{f}^8 &= \underline{\mu}_{\tilde{F}_2^8}(x_1) \star \underline{\mu}_{\tilde{F}_3^8}(x_2) \\ &= 0.5037\times 0.4822 = 0.2429 \end{aligned} \tag{25}$$

and

$$\begin{aligned} \bar{f}^8 &= \bar{\mu}_{\tilde{F}_2^8}(x_1) \star \bar{\mu}_{\tilde{F}_3^8}(x_2) \\ &= 0.6695\times 0.6469 = 0.4331. \end{aligned} \tag{26}$$

Similarly, we can compute the other firing intervals: $F^2 = [0, 0.0947]$, $F^3 = [0, 0.1601]$, $F^4 = [0, 0.0701]$, $F^7 = [0, 0.2562]$, $F^9 = [0, 0.1897]$, $F^{12} = [0, 0.0956]$, $F^{13} = [0, 0.1617]$, and $F^{14} = [0, 0.0708]$.

The type-reduced output, obtained using the procedure given in the Appendix , is

$$\begin{aligned} Y(3,6) &= [y_l(3,6), y_r(3,6)] \\ &= \int_{c_2}\cdots\int_{c_{14}}\int_{f_2}\cdots\int_{f_{14}} 1 \Bigg/ \frac{\sum_{i=2,\ldots,14} c_i f_i}{\sum_{i=2,\ldots,14} f_i} \\ &= [5.7338, 7.6942] \end{aligned} \tag{27}$$

where $c_i \in C_{\rm avg}^i$, and $f_i \in F^i$ $(i = 2,3,4,7,8,9,12,13,14)$.

Similarly, we can compute $Y(x_1,x_2)$ for any value (x_1,x_2) in the measurement domain, to obtain the region $Y(x_1,x_2)$ between the two hypersurfaces $y_l(x_1,x_2)$ and $y_r(x_1,x_2)$ in the measurement domain $x_1 \in [0,10]$ and $x_2 \in [0,10]$ (Fig. 9). Note that $Y(x_1,x_2) \subset [0,10]$ for any $[x_1,x_2]$. Observe, from Fig. 9, that $y_l(x_1,x_2)$ and $y_r(x_1,x_2)$ are monotonically decreasing functions.

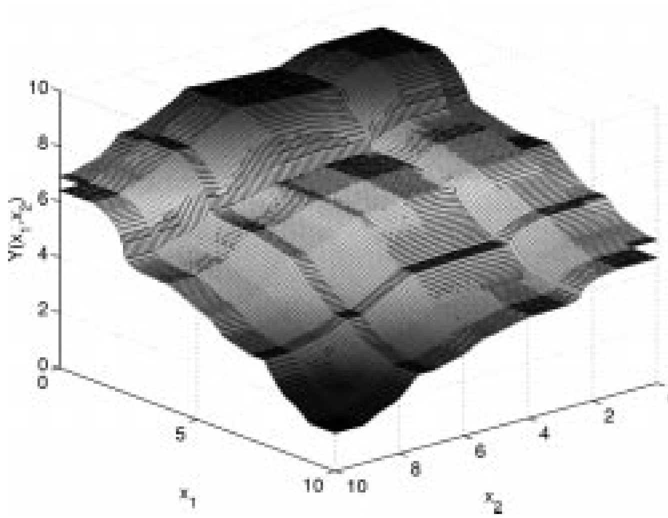

Fig. 9. Type-2 FLS for CAC. The confidence interval of accepting the call, $Y(x_1,x_2)$, versus x_1 and x_2.

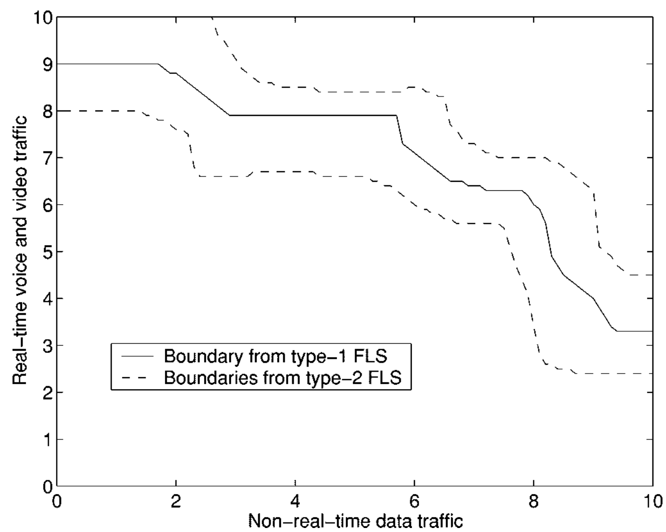

Fig. 10. Decision boundary generated by the type-1 FLS, and decision boundaries generated by the type-2 FLS.

E. CAC Decision Boundaries

CAC is a binary decision problem—accept or reject, so *the confidence of accepting the call* $+$ *the confidence of rejecting the call* $= 10$. A call will be accepted if *the confidence of accepting the call* > 5. It is then very straightforward to obtain the decision boundary for *the confidence of accepting the call* $= 5$.

For the type-1 FLS, the decision boundary is $y(x_1,x_2) = 5$, as plotted in Fig. 10 (solid line). When a call occurs with input rate of traffic (x_1,x_2) below the decision boundary, then it will be accepted; similarly, if it occurs above the decision boundary, it will be rejected. As we see, the decision boundary generated from type-1 FLS is a hard-threshold.

The decision boundaries for the upper and lower hypersurfaces of the type-2 FLS output can be represented as $y_l(x_1,x_2) = 5$ and $y_r(x_1,x_2) = 5$, which are plotted in Fig. 10 using dashed lines. If a designer wishes to increase the bandwidth utilization, he can choose the upper decision boundary. When a call occurs with input rate of traffic (x_1,x_2) below the upper decision boundary, then it will be accepted. If, on the other hand, the designer wishes to decrease the cell-loss-ratio (CLR), he can choose the lower decision boundary. If the designer wishes to achieve a compromise in performance between bandwidth utilization and CLR, he can choose any decision boundary between the upper and lower decision boundaries. We see, therefore, that a type-2 FLS provides a soft decision boundary that depends on the preference of the designer; it can be any decision boundary between the lower and upper decision boundaries.

The lower and upper decision boundaries plotted in Fig. 10 are for the footprint of uncertainty in Fig. 7 ($\rho = 50\%$). By varying ρ, we could get families of intervals, each labeled with different ρ. A designer could make a choice by "thinking" in terms of a level of uncertainty. Note that $\rho = 0$ corresponds to the type-1 FLS case.

IV. Conclusions and Future Works

The type-2 FLS-based CAC method has the following features.

1) it combines the input rate of real-time voice and video traffic and nonreal-time data traffic in the decision of connection admission;
2) it combines the experiences from lots of experts, so that an acceptable decision boundary can be obtained;
3) it provides an interval decision, so that a soft-decision can be made based on a design tradeoff between cell loss ratio and bandwidth utilization.

Recently, it has been observed that video/voice/data traffic have self-similarity [11], [20], [35]. According to Stallings [29], "Self-similarity is such an important concept that, in a way, it is surprising that only recently has it been applied to data communications traffic analysis." Further, "Since 1993, a number of studies reported in the literature have documented that the pattern of data traffic is well modeled by self-similar processes in a wide variety of real-world networking situations." Such self-similarity is quite common in both natural and human-made phenomena [29] such as the distribution of earthquakes, ocean waves, fluctuation of the stock market. These kinds of time-series have been successfully forecasted using the Box–Jenkins method [1]. We have used type-2 FLSs for time-series forecasting [21]. Video/voice/data traffic is like a time-series; so, we can also use a type-2 FLS to forecast the future input rate of traffic for dynamic bandwidth allocation and CAC.

We have demonstrated how a type-2 FLS can be used in decision making; so, combining all the control problems in ATM networks, such as policing, rate control, buffer management, traffic prediction into a type-2 fuzzy logic coordinating system is also a very promising research area.

Appendix
Computation of Weighted Average of Interval Type-1 Sets

Consider the weighted average

$$y(z_1, \ldots, z_M, w_1, \ldots, w_M) = \frac{\sum_{l=1}^{M} w_l z_l}{\sum_{l=1}^{M} w_l} \tag{28}$$

where $z_l \in \Re$ and $w_l \in [0, 1]$ for $l = 1, \ldots, M$. If each z_l is replaced by an interval type-1 set $Z_l \subset \Re$ and each w_l is replaced by an interval type-1 set $W_l \subset [0, 1]$, then the extension of (28), according to (8), is

$$\begin{aligned} &Y(Z_1, \ldots, Z_M, W_1, \ldots, W_M) \\ &\quad = \int_{z_1} \cdots \int_{z_M} \int_{w_1} \cdots \int_{w_M} 1 \bigg/ \frac{\sum_{l=1}^{M} w_l z_l}{\sum_{l=1}^{M} w_l} \\ &\quad = [y_l, y_r] \end{aligned} \tag{29}$$

where $w_l \in W_l$ and $z_l \in Z_l$ for $l = 1, \ldots, M$, and all the integrals denote logical union.

In [15] and [17] we developed a computational procedure to compute the weighted average Y, which itself is an interval type-1 set. We restate the procedure here. Note that, since Y is an interval type-1 set, only two sets of computations are required, one for each end point of the domain interval of Y, y_l and y_r.

Let $Z_l = [m_l - s_l, m_l + s_l]$ and $W_l = [h_l - \Delta_l, h_l + \Delta_l]$ for $l = 1, \ldots, M$. y_r can be obtained by following the iterative procedure. We set $z_l = m_l + s_l$ $(l = 1, \ldots, M)$, and, without loss of generality, assume that the z_ls are arranged in ascending order, i.e., $z_1 \le z_2 \le \cdots \le z_M$.

1) Set $w_l = h_l$ for $l = 1, \ldots, M$, and compute $S = y(m_1 + s_1, \ldots, m_M + s_M, h_1, \ldots, h_M)$ using (28).
2) Find k $(1 \le k \le M - 1)$ such that $z_k \le S \le z_{k+1}$.
3) Set $w_l = h_l - \Delta_l$ for $l \le k$ and $w_l = h_l + \Delta_l$ for $l \ge k + 1$, and compute $S' = y(m_1 + s_1, \ldots, m_M + s_M, h_1 - \Delta_1, \ldots, h_k - \Delta_k, h_{k+1} + \Delta_{k+1}, \ldots, h_M + \Delta_M)$ using (28).
4) Check if $S' = S$. If yes, stop. S' is the right end-point of Y. If no, go to step 5.
5) Set S equal to S'. Go to step 2.

The left end-point of the domain interval of $Y(Z_1, \ldots, Z_M, W_1, \ldots, W_M)$, y_l, can be obtained by using a procedure similar to the one described above. Only two changes need to be made: 1) we must set $z_l = m_l - s_l$ for $l = 1, \ldots, M$ and 2) in Step 3, we must set $w_l = h_l + \Delta_l$ for $l \le k$ and $w_l = h_l - \Delta_l$ for $l \ge k + 1$, to compute the weighted average $S' = y(m_1 - s_1, \ldots, m_M - s_M, h_1 + \Delta_1, \ldots, h_k + \Delta_k, h_{k+1} - \Delta_{k+1}, \ldots, h_M - \Delta_M)$.

This computational procedure can be used to compute the type-reduced set of an interval type-2 FLS [see (20)], as well as to compute the centroid of an interval type-2 set [see (11)]. In the latter case, all the z_ls are crisp, so that, we set $s_l = 0$ for $l = 1, \ldots, M$.

Acknowledgment

The authors would like to thank the reviewers of the first version of this paper for their very helpful suggestions, most of which we incorporated into this version.

References

[1] G. Box, G. Jenkins, and G. Reinsel, *Time Series Analysis*, 3rd ed. Englewood Cliffs, NJ: Prentice-Hall, 1994.

[2] J. L. Chaneau, M. Gunaratne, and A. G. Altschaeffl, "An application of type-2 sets to decision making in engineering," *Analysis of Fuzzy Information—Vol. II: Artificial Intelligence and Decision Systems*, 1987.

[3] C.-J. Chang, C.-H. Lin, D.-S. Guan, and R.-G. Cheng, "A power-spectrum based neural fuzzy connection admission mechanism for ATM networks," in *IEEE Proc. ICC*, Monterey, CA, 1997, pp. 1709–1713.

[4] ——, "Design of a power-spectrum-based ATM connection admission controller for multimedia communications," *IEEE Trans. Ind. Electron.*, vol. 45, pp. 52–59, Feb. 1998.

[5] S. Chong and S.-Q. Li, "Probabilistic burstiness-curve-based connection control for real-time multimedia services in ATM networks," *IEEE J. Select. Areas Commun.*, vol. 15, pp. 1072–1086, Aug. 1997.

[6] M. Delago and A. Gonzalez, "An inductive learning procedure to identify fuzzy systems," *Fuzzy Sets Syst.*, vol. 55, no. 2, pp. 121–133, 1993.

[7] M. Delgado, F. Herrera, E. Herrera-Viedma, and L. Martinez, "Combining numerical and linguistic information in group decision making," *J. Inform. Sci.*, vol. 107, pp. 177–194, 1998.

[8] C. Douligeris and G. Develekos, "Neuro-fuzzy control in ATM networks," *IEEE Commun. Mag.*, pp. 154–163, May 1997.

[9] D. Dubois and H. Prade, *Fuzzy Sets and Systems: Theory and Applications*. New York: Academic, 1980.

[10] J. S. Evans and D. Everitt, "Effective bandwidth-based admission control for multiservice CDMA cellular networks," *IEEE Trans. Veh. Technol.*, vol. 48, pp. 36–46, Jan. 1999.

[11] M. W. Garrett and W. Willinger, "Analysis, modeling and generation of self-similar VBR video traffic," in *Proc. SIGCOMM'94*, London, U.K., Aug. 1994, pp. 269–280.

[12] I. W. Habib, "Applications of neurocomputing in traffic management of ATM networks," *Proc. IEEE*, vol. 84, pp. 1430–1441, Oct. 1996.

[13] "Special issue on computational and artificial intelligence in high speed networks," *IEEE J. Select. Areas Commun., Special Issue Artif. Intell. High-Speed Networks*, vol. 15, no. 2, Feb. 1997.

[14] R. I. John, P. R. Innocent, and M. R. Barnes, "Type 2 fuzzy sets and neuro-fuzzy clustering of radiographic tibia images," in *Proc. 1998 IEEE Int. Conf. Fuzzy Systems*, Anchorage, AK, May 1998, pp. 1373–1376.

[15] N. N. Karnik and J. M. Mendel. An Introduction to type-2 fuzzy logic systems. [Online]. Available: http://sipi.usc.edu/~mendel/report

[16] ——, "Operations on type-2 fuzzy sets," *Fuzzy Sets Syst.*, submitted for publication.

[17] ——, "Centroid of a type-2 fuzzy set," *Inf. Sci.*, submitted for publication.

[18] N. N. Karnik, J. M. Mendel, and Q. Liang, "Type-2 fuzzy logic systems," *IEEE Trans. Fuzzy Syst.*, vol. 7, pp. 643–658, Dec. 1999.

[19] A. Kaufmann and M. M. Gupta, *Introduction to Fuzzy Arithmetic: Theory and Applications*. New York: Van Nostrand Reinhold, 1991.

[20] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, "On the self-similar nature of ethernet traffic," *IEEE Trans. Networking*, vol. 2, pp. 1–15, Feb. 1994.

[21] Q. Liang and J. M. Mendel, "Interval type-2 fuzzy logic systems: Theory and design," *IEEE Trans. Fuzzy Syst.*, submitted for publication.

[22] ——, "MPEG VBR video traffic modeling and classification using fuzzy techniques," *IEEE Trans. Fuzzy Syst.*, submitted for publication.

[23] E. H. Mamdani, "Applications of fuzzy logic to approximate reasoning using linguistic systems," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-7, pp. 1182–1191, Dec. 1977.

[24] E. H. Marimin, M. Umano, I. Hatono, and H. Tamura, "Linguistic labels for expressing fuzzy preference relations in fuzzy group decision making," *IEEE Trans. Syst., Man, Cybern. B*, vol. 28, pp. 205–218, Apr. 1998.

[25] H. R. Mehrvar and T. Le-Ngoc, "Fuzzy logic in estimation of traffic burstiness for admission control in broadband networks," in *Proc. IEEE ICC*, Monterey, CA, 1997, pp. 1090–1094.

[26] J. M. Mendel, "Fuzzy logic systems for engineering: A tutorial," *Proc. IEEE*, vol. 83, pp. 345–377, Mar. 1995.

[27] ——, "Computing with words when words can mean different things to different people," in *Proc. Int. ICSC Congr. Computational Intelligence: Methods Applications, 3rd Annu. Symp. Fuzzy Logic Applications*, Rochester, NY, June 22–25, 1999.

[28] M. Mizumoto and K. Tanaka, "Some properties of fuzzy sets of type-2," *Inf. Contr.*, vol. 31, pp. 312–340, 1976.

[29] W. Stallings, *High-Speed Networks: TCP/IP and ATM Design Principles*. Upper Saddle River, NJ: Prentice-Hall, 1998.

[30] M. Sugeno and K. Tanaka, "Successive identification of a fuzzy model and its applications to prediction of a complex system," *Fuzzy Sets Syst.*, vol. 42, no. 3, pp. 315–334, 1991.

[31] M. Sugeno and T. Yasukawa, "A fuzzy-logic-based approach to qualitative modeling," *IEEE Trans. Fuzzy Syst.*, vol. 1, pp. 7–31, Feb. 1993.

[32] Y. Tanaka and S. Hosaka, "Fuzzy control of telecommunications networks using learning technique," *Elect. Commun. Jpn.*, pt. I, vol. 76, pp. 41–51, Dec. 1993.

[33] K. Uehara and K. Hirota, "Fuzzy connection admission control for ATM networks based on possibility distribution of cell loss ratio," *IEEE J. Select. Areas Commun.*, vol. 15, pp. 179–190, Feb. 1997.

[34] M. Wagenknecht and K. Hartmann, "Application of fuzzy sets of type 2 to the solution of fuzzy equation systems," in *Fuzzy Sets Syst.*, vol. 25, 1988, pp. 183–190.

[35] W. Willinger, M. S. Taqqu, R. Sherman, and D. V. Wilson, "Self-similarity through high-variability: Statistical analysis of ethernet LAN traffic at the source level," *IEEE Trans. Networking*, vol. 5, pp. 71–86, Feb. 1997.

[36] K. C. Wu, "Fuzzy interval control of mobile robots," *Comput. Elect. Eng.*, vol. 22, no. 3, pp. 211–229, 1996.

[37] R. R. Yager, "Fuzzy subsets of type II in decisions," *J. Cybern.*, vol. 10, pp. 137–159, 1980.

[38] Y. Yoshinari, W. Pedrycz, and K. Hirota, "Construction of fuzzy models through clustering techniques," *Fuzzy Sets Syst.*, vol. 54, pp. 157–165, 1993.

[39] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning—1," *Inf. Sci.*, vol. 8, pp. 199–249, 1975.

[40] Z.-L. Zhang, J. Kurose, J. D. Salehi, and D. Towsley, "Smoothing, statistical multiplexing, and call admission control for stored video," *IEEE J. Select. Areas Commun.*, vol. 15, pp. 1148–1166, Aug. 1997.

Qilian Liang received the B.S. degree from Wuhan University, Wuhan, China, in 1993, the M.S. degree from Beijing University of Posts and Telecommunications, Beijing, China, in 1996, and the Ph.D. degree from University of Southern California (USC), Los Angeles, in May 2000, all in electrical engineering.

He was a Research Assistant with the Department of Electrical Engineering Systems, USC, from September 1997 to May 2000. His research interests include wireless communications, broadband networks, fuzzy logic systems, and video traffic classification.

Nilesh N. Karnik (S'99–A'99) was born in Thane, India, in 1971. He received the B.E. degree in electronics engineering from Victoria Jubilee Technical Institute, University of Mumbai, India, in 1993, and the M.S. and Ph.D. degrees, both the electrical engineering, from the University of Southern California, Los Angeles, in 1997 and 1998, respectively.

He is currently with the Applied Technology Group, Tata Infotech, Ltd., Mumbai. His interests include fuzzy systems and knowledge based systems in general, and their applications to scheduling and optimization problems, modeling, and data mining.

Jerry M. Mendel (S'59–M'61–SM'72–F'78) received the Ph.D. degree in electrical engineering from the Polytechnic Institute, Brooklyn, NY.

He has been with the University of Southern California, Los Angeles, since 1974. He is currently Professor of Electrical Engineering and Associate Director for Education of the Integrated Media Systems Center. He has published more than 380 technical papers and is author and/or editor of seven books. His present research interests include type-2 fuzzy logic systems and their applications to a wide range of problems.

Dr. Mendel is a Distinguished Member of the IEEE Control Systems Society. He was President of the IEEE Control Systems Society in 1986. Among his awards are the 1983 Best Transactions Paper Award of the IEEE Geoscience and Remote Sensing Society, the 1992 Signal Processing Society Paper Award, a 1984 IEEE Centennial Medal, and an IEEE Third Millenium Medal.