Super-Exponential Convergence of the Karnik–Mendel Algorithms for Computing the Centroid of an Interval Type-2 Fuzzy Set

Jerry M. Mendel, *Life Fellow, IEEE*, and Feilong Liu, *Student Member, IEEE*

***Abstract*—Computing the centroid of an interval T2 FS is an important operation in a type-2 fuzzy logic system (where it is called *type-reduction*), but it is also a potentially time-consuming operation. The Karnik–Mendel (KM) iterative algorithms are widely used for doing this. In this paper, we prove that these algorithms *converge monotonically* and *super-exponentially fast*. Both properties are highly desirable for iterative algorithms and explain why in practice the KM algorithms have been observed to converge very fast, thereby making them very practical to use.**

***Index Terms*—Centroid, interval type-2 fuzzy sets, Karnik–Mendel (KM) algorithms, type-2 fuzzy sets.**

I. INTRODUCTION

AN INTERVAL type-2 fuzzy set (IT2 FS) is today the most widely used T2 FS because it is computationally simple to use. When such FSs are used in a rule-based fuzzy logic system (FLS) (e.g., [4], [5], [9]–[14], [17], and [23]–[25]), the result is an *interval T2 FLS* (IT2 FLS). In such a FLS, fired-rule output sets are also IT2 FSs, and to go from such sets to a number, as is usually required in most engineering applications of a FLS, one must perform two successive operations, type-reduction and defuzzification. Type-reduction maps the output T2 FS into a type-1 (T1) FS, and defuzzification converts that T1 FS into a number.

Type-reduction methods were developed by Karnik and Mendel [6], [7] and are elaborated upon in [17]. When they are applied to a general T2 FS they require an astronomical number of computations. When they are applied to an IT2 FS, they require a very small number of computations, which is one of the major reasons that IT2 FLSs have received attention whereas general T2 FLSs have not.

Even for IT2 FLSs there can be many different kinds of type-reduction. The ones that have been developed so far all extend a T1 centroid calculation to T2 FSs, so that if all sources of uncertainty disappear the output of an IT2 FLS reduces to that of a T1 FLS. So, computing the centroid of an IT2 FS plays a central role in IT2 FLSs.

The centroid of an IT2 FS also provides a measure of the uncertainty of an IT2 FS [24], and more recently has been the basis for going from data collected from a group of subjects (about an interval that they associate with the meaning of a word) to the footprint of uncertainty (FOU) of an IT2 FS that models that word [19]–[21].

We explain what the centroid of an interval T2 FS is in Section II. Here we note that it is an interval set that is completely characterized by two numbers, its left and right end-points. There are no known closed-form formulas for these end points; however, Karnik and Mendel [6], [7] have developed iterative algorithms for computing these end-points exactly. Their algorithms have come to be known as the *Karnik–Mendel (KM) algorithms*.

For many years we have observed, by means of computer simulations, that the KM algorithms, although they are iterative, converge to their exact solutions very rapidly. The only available convergence statement for them is very pessimistic (convergence occurs in *at most* N iterations where N equals the number of sampled values of the primary variable [7]; as N increases this bound becomes very uninformative), and so we have been puzzled by the much more optimistic results (e.g., convergence in ten or fewer iterations, regardless of N, is quite common) that always appeared from the simulations. The purpose of this paper is to quantify these convergence observations and to prove *super-exponential convergence* for the algorithms. We believe that this will make the use of these algorithms much more wide spread.

Note that not only are the KM algorithms used to compute the centroid of an interval T2 FS, they are also widely used in an interval T2 FLS to compute the generalized centroid for center-of sets type reduction. In addition, they can be used to compute the so-called *fuzzy weighted average* (FWA) [1]–[3], [8], [15], when its computation is based on α-cuts, because each α-cut of a FWA has exactly the same structure as the centroid of an interval T2 FS.

The rest of this paper is organized as follows. Section II quantifies the centroid of an interval T2 FS and reviews the KM algorithms for its computation. Section III formulates continuous versions of the KM algorithms because they are used in the convergence analyses of those algorithms. Section IV provides important properties of the KM algorithms. Section V examines convergence properties of those algorithms. Section VI examines the applicability of our results. Section VII draws conclusions.

II. CENTROID AND THE KM ALGORITHMS

For readers who are not familiar with interval type-2 fuzzy sets (IT2 FS), we provide some basics about them in Appendix B. Here, we wish to reiterate the fact that an IT2 FS

Manuscript received February 26, 2005; revised October 1, 2005 and October 26, 2005.

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

Digital Object Identifier 10.1109/TFUZZ.2006.882463

$\tilde{A}$ can be decomposed into the union of all of its embedded T2 FSs [see (B-7) –(B-9)]. Consequently, the centroid [6], [7], [17] of an IT2 FS is the union of the centroids of all of its embedded T2 FSs. From (B-10), we see that this means we need to compute the centroids of all n_A embedded T1 FSs that are contained within the footprint of uncertainty (FOU) of $\tilde{A}$. The results of doing this will be a collection of n_A numbers, and these numbers will have both a smallest and largest element, $c_1(\tilde{A}) \equiv c_1$ and $c_r(\tilde{A}) \equiv c_r$, respectively. That such numbers exist is because the centroid of each of the embedded T1 FSs is a bounded number. Associated with each of these numbers will be a membership grade of 1, because the secondary grades of an IT2 FS are all equal to 1. Letting $C_{\tilde{A}}$ denote the centroid of $\tilde{A}$, this means that

$$C_{\tilde{A}} = 1/[c_l(\tilde{A}), c_r(\tilde{A})] \equiv 1/[c_l, c_r] \tag{1}$$

$$c_l = \min_{\forall\, \theta_i \in \left[\underline{\mu}_{\tilde{A}}(x_i), \bar{\mu}_{\tilde{A}}(x_i)\right]} \frac{\sum_{i=1}^{N} x_i\theta_i}{\sum_{i=1}^{N} \theta_i} \tag{2}$$

$$c_r = \max_{\forall\, \theta_i \in \left[\underline{\mu}_{\tilde{A}}(x_i), \bar{\mu}_{\tilde{A}}(x_i)\right]} \frac{\sum_{i=1}^{N} x_i\theta_i}{\sum_{i=1}^{N} \theta_i} \tag{3}$$

$\underline{\mu}_{\tilde{A}}(x_i)$ and $\bar{\mu}_{\tilde{A}}(x_i)$, $\forall\, x_i \in X$, are the lower and upper membership functions[1] (defined in Appendix B) that are associated with $\tilde{A}$. Note that in (2) and (3). The challenge is to compute c_l and c_r.

Letting

$$y(\theta_1, \ldots, \theta_N) \equiv \frac{\sum_{i=1}^{N} x_i\theta_i}{\sum_{i=1}^{N} \theta_i} \tag{4}$$

and differentiating $y(\theta_1, \ldots, \theta_N)$ with respect to θ_k, we find that

$$\frac{\partial y(\theta_1, \ldots, \theta_N)}{\partial \theta_k} = \frac{x_k - y(\theta_1, \ldots, \theta_N)}{\sum_{i=1}^{N} \theta_i}, \qquad k = 1, \ldots, N. \tag{5}$$

Unfortunately, equating $\partial y/\partial\theta_k$ to zero does not give us any information about the value of θ_k that optimizes $y(\theta_1, \ldots, \theta_N)$. When we do this, we find

$$y(\theta_1, \ldots, \theta_N) = x_k \Rightarrow \frac{\sum_{i=1}^{N} x_i\theta_i}{\sum_{i=1}^{N} \theta_i} = x_k \Rightarrow \frac{\sum_{i\neq k}^{N} x_i\theta_i}{\sum_{i\neq k}^{N} \theta_i} = x_k. \tag{6}$$

Observe that θ_k no longer appears in the final expression in (6), so that the direct calculus approach does not work. Returning to (5), we see that, because $\sum_{i=1}^{N} \theta_i > 0$, it is true that

$$\frac{\partial y(\theta_1, \ldots, \theta_N)}{\partial \theta_k} \begin{cases} \geq 0 & \text{if } x_k \geq y(\theta_1, \ldots, \theta_N) \\ < 0 & \text{if } x_k < y(\theta_1, \ldots, \theta_N) \end{cases}. \tag{7}$$

This equation gives us the direction in which θ_k should be changed in order to increase or decrease $y(\theta_i, \ldots, \theta_N)$, i.e., see (8), as shown at the bottom of the page.

Because $\theta_k \in [\underline{\mu}_{\tilde{A}}(x_k), \bar{\mu}_{\tilde{A}}(x_k)]$, the maximum value θ_k can attain is $\bar{\mu}_{\tilde{A}}(x_k)$ and the minimum value it can attain is $\underline{\mu}_{\tilde{A}}(x_k)$. Equation (8) therefore implies that $y(\theta_1, \ldots, \theta_N)$ attains its *minimum value*, c_l, if: (1) for those values of k for which $x_k < y(\theta_1, \ldots, \theta_N)$, we set $\theta_k = \underline{\mu}_{\tilde{A}}(x_k)$, and (2) for those values of k for which $x_k > y(\theta_1, \ldots, \theta_N)$, we set $\theta_k = \underline{\mu}_{\tilde{A}}(x_k)$. Similarly, we can deduce from (8) that $y(\theta_1, \ldots, \theta_N)$ attains its *maximum value*, c_r, if: (1) for those values of k for which $x_k < y(\theta_1, \ldots, \theta_N)$, we set $\theta_k = \underline{\mu}_{\tilde{A}}(x_k)$, and (2) for those values of k for which $x_k > y(\theta_1, \ldots, \theta_N)$, we set $\theta_k = \bar{\mu}_{\tilde{A}}(x_k)$. Consequently, to compute c_l or c_r θ_k *switches only one time* between $\bar{\mu}_{\tilde{A}}(x_k)$ and $\underline{\mu}_{\tilde{A}}(x_k)$. The KM algorithms (described later) locate the switch point, and in general the switch point for c_r, R, is different from the switch point for c_l, L.

Putting all of these facts together, we obtain the following formulas for c_l and c_r:

$$c_l = \frac{\sum_{i=1}^{L} x_i\bar{\mu}_{\tilde{A}}(x_i) + \sum_{i=L+1}^{N} x_i\underline{\mu}_{\tilde{A}}(x_i)}{\sum_{i=1}^{L} \bar{\mu}_{\tilde{A}}(x_i) + \sum_{i=L+1}^{N} \underline{\mu}_{\tilde{A}}(x_i)} \tag{9}$$

$$c_r = \frac{\sum_{i=1}^{R} x_i\underline{\mu}_{\tilde{A}}(x_i) + \sum_{i=R+1}^{N} x_i\bar{\mu}_{\tilde{A}}(x_i)}{\sum_{i=1}^{R} \underline{\mu}_{\tilde{A}}(x_i) + \sum_{i=R+1}^{N} \bar{\mu}_{\tilde{A}}(x_i)}. \tag{10}$$

The ***KM algorithm for computing*** c_l is as follows.

1. Initialize θ_i by setting[2]

$$\theta_i = \frac{1}{2}[\underline{\mu}_{\tilde{A}}(x_i) + \bar{\mu}_{\tilde{A}}(x_i)], \qquad i = 1, \ldots, N \tag{11}$$

and then compute

$$c' = c(\theta_1, \ldots, \theta_N) = \frac{\sum_{i=1}^{N} x_i\theta_i}{\sum_{i=1}^{N} \theta_i}. \tag{12}$$

$$\left.\begin{array}{ll} \text{If } x_k > y(\theta_1, \ldots, \theta_N) & y(\theta_1, \ldots, \theta_N) \text{ increases (decreases) as } \theta_k \text{ increases (decreases)} \\ \text{If } x_k < y(\theta_1, \ldots, \theta_N) & y(\theta_1, \ldots, \theta_N) \text{ increases (decreases) as } \theta_k \text{ decreases (increases)} \end{array}\right\} \tag{8}$$

---

[1] If $\underline{\mu}(x) = \bar{\mu}(x) = \mu(x)$ for $\forall\, x \in X$, then the IT2 FS reduces to a T1 FS. In this paper, where we are only interested in how to compute the centroid of an IT2 FS, we exclude the reduction of the IT2 FS to a T1 FS, because when this happens we have a "nonproblem" as far as this paper is concerned.

[2] Other initializations are possible, but this is the one we shall use here because it is so simple.

2. Find k $(1 \leq k \leq N-1)$ such that

$$x_k \leq c' \leq x_{k+1}. \tag{13}$$

3. Set[3]

$$\theta_i = \begin{cases} \bar{\mu}_{\tilde{A}}(x_i) & i \leq k \\ \underline{\mu}_{\tilde{A}}(x_i) & i \geq k+1 \end{cases} \tag{14}$$

and compute

$$c'' = \frac{\sum\limits_{i=1}^{k} x_i \bar{\mu}_{\tilde{A}}(x_i) + \sum\limits_{i=k+1}^{N} x_i \underline{\mu}_{\tilde{A}}(x_i)}{\sum\limits_{i=1}^{k} \bar{\mu}_{\tilde{A}}(x_i) + \sum\limits_{i=k+1}^{N} \underline{\mu}_{\tilde{A}}(x_i)} \tag{15}$$

4. Check if $c'' = c'$. If yes, stop and set $c'' = c_l$. If no, go to Step 5.

5. Set $c' = c''$ and go to Step 2.

III. CONTINUOUS VERSION OF THE KM ALGORITHMS

Although the KM algorithms are usually stated for the discrete situation, as we have just done, we found that it is much more convenient to study properties of the algorithms by using their continuous versions. Because the KM algorithms are so similar for the calculations of c_l and c_r, we focus our attention only on convergence properties of the KM algorithm for c_l and leave the results for c_r to the reader. So, to begin, we state the **continuous version[4] of the KM Algorithm for computing c_l.**

1. Compute the initial value, c_0, for c_l, as[5]

$$\begin{aligned} c_0 &\equiv \frac{\int_{-\infty}^{\infty} \frac{x[\bar{\mu}(x)+\underline{\mu}(x)]}{2} dx}{\int_{-\infty}^{\infty} \frac{[\bar{\mu}(x)+\underline{\mu}(x)]}{2} dx} \\ &= \frac{\int_{-\infty}^{\infty} x[\bar{\mu}(x)+\underline{\mu}(x)] dx}{\int_{-\infty}^{\infty} [\bar{\mu}(x)+\underline{\mu}(x)] dx} \end{aligned} \tag{16}$$

and then set $j = 1$ and

$$\alpha_1 = c_0 \tag{17}$$

2. Compute $c_l(\alpha_j)$ as

$$c_l(\alpha_j) = \frac{\int_{-\infty}^{\alpha_j} x\bar{\mu}(x)dx + \int_{\alpha_j}^{\infty} x\underline{\mu}(x)dx}{\int_{-\infty}^{\alpha_j} \bar{\mu}(x)dx + \int_{\alpha_j}^{\infty} \underline{\mu}(x)dx} \tag{18}$$

3. If convergence has occurred (see Corollary 2), STOP; otherwise, go to Step 4.

4. Set

$$\alpha_{j+1} = c_l(\alpha_j). \tag{19}$$

5. Set $j = j+1$, and go to Step 2.

Before we study the convergence properties of the KM algorithm, we pause to present some important properties about function $c_l(\alpha_j)$.

IV. PROPERTIES OF $c_l(\alpha_j)$

In Section V, we provide the KM algorithm with a graphical interpretation, but to do so we must first establish some properties about the function $c_l(\alpha_j)$ so that we can sketch it. Note that proofs of theorems and corollaries are given in Appendix A.

Theorem 1: [20], [22] We define $c_l(\alpha)$ as[6]

$$c_l(\alpha) = \frac{\int_{-\infty}^{\alpha} x\bar{\mu}(x)dx + \int_{\alpha}^{\infty} x\underline{\mu}(x)dx}{\int_{-\infty}^{\alpha} \bar{\mu}(x)dx + \int_{\alpha}^{\infty} \underline{\mu}(x)dx}. \tag{20}$$

Then

$$\arg\min_{\alpha} c_l(\alpha) = c_l \tag{21}$$

i.e.,

$$c_l = \frac{\int_{-\infty}^{c_l} x\bar{\mu}(x)dx + \int_{c_l}^{\infty} x\underline{\mu}(x)dx}{\int_{-\infty}^{c_l} \bar{\mu}(x)dx + \int_{c_l}^{\infty} \underline{\mu}(x)dx}. \tag{22}$$

■

Proof: See Appendix A.

The result in (21) is very interesting and somewhat surprising, because it shows that when the value of α is found which minimizes $c_l(\alpha)$ it will be $\alpha = c_l$. Of course, if X is discretized (for computational purposes), then $\alpha \to L \approx c_l$ but L does not exactly equal c_l, which probably explains why (20) was not observed by Karnik and Mendel.

Theorem 2: It is true that

$$c_l(\alpha) \begin{cases} > \alpha & \alpha < c_l \\ < \alpha & \alpha > c_l \end{cases} \tag{23a}$$

and, consequently, that

$$\frac{\partial c_l(\alpha)}{\partial \alpha} \begin{cases} \leq 0 & \alpha \leq c_l \\ \geq 0 & \alpha > c_l. \end{cases} \tag{23b}$$

■

---

[3] The KM algorithm for computing c_r is very similar, except in this step we set $\theta_i = \begin{cases} \underline{\mu}_{\tilde{A}}(x_i) & i \leq k \\ \bar{\mu}_{\tilde{A}}(x_i) & i \geq k+1 \end{cases}$ so that $c'' = \left[\sum_{i=1}^{k} x_i \underline{\mu}_{\tilde{A}}(x_i) + \sum_{i=k+1}^{N} x_i \bar{\mu}_{\tilde{A}}(x_i)\right] / \left[\sum_{i=1}^{k} \underline{\mu}_{\tilde{A}}(x_i) + \sum_{i=k+1}^{N} \bar{\mu}_{\tilde{A}}(x_i)\right]$.

[4] We discuss the applicability of using the continuous versions of the KM algorithms to different kinds of problems in Section VI.

[5] Here, we are using the initialization stated in (11), and c_0 is the continuous version of c' in (12).

[6] c_l in (9) is the discretized version of $c_l(\alpha)$ in (20). We have much more to say about the relationships between (9) and (20) in Section VI.

Proof: See Appendiz A.

Equation (23a) means that $c_l(\alpha)$ lies above α when $c_l(\alpha)$ is to the left of c_l, and $c_l(\alpha)$ lies below α when $c_l(\alpha)$ is to the right of c_l. Equation (23b) means $c_l(\alpha)$ has a negative (or zero) slope to the left of $\alpha = c_l$, and a positive (or zero) slope to the right of $\alpha = c_l$. That the slope cannot be zero for all values of α is obvious, or else function $c_l(\alpha)$ in (20) would be a constant for all α, which it is not.

Note that $y(\theta_1, \ldots, \theta_N) = \sum_{i=1}^{N} x_i\theta_i / \sum_{i=1}^{N} \theta_i$ is not convex[7]; hence, it is possible for this function (and its continuous counterpart) to have some flat spots, to the left of or to the right of its minimum value.

Theorem 3: The Taylor series expansion of $c_l(\alpha)$ about its minimum value c_l is

$$c_l(\alpha) = c_l + \frac{1}{2}s(c_l)(\alpha - c_l)^2 + \frac{1}{3!}\left[d(c_l) - 3s^2(c_l)\right](\alpha - c_l)^3 + \cdots \qquad (24)$$

where

$$s(c_l) = \frac{\partial^2 c_l(\alpha)}{\partial \alpha^2}\Big|_{\alpha = c_l} = \frac{\bar{\mu}(c_l) - \underline{\mu}(c_l)}{\int_{-\infty}^{c_l} \bar{\mu}(x)dx + \int_{c_l}^{\infty} \underline{\mu}(x)dx} \geq 0 \qquad (25)$$

and

$$d(c_l) \equiv 2\frac{\frac{\partial}{\partial \alpha}\left[\bar{\mu}(\alpha) - \underline{\mu}(\alpha)\right]|_{\alpha = c_l}}{\int_{-\infty}^{c_l} \bar{\mu}(x)dx + \int_{c_l}^{\infty} \underline{\mu}(x)dx}. \qquad (26)$$

■

Proof: See Appendix A.

Comment: It is explained in Appendix A why the truncated Taylor series expansion in (24) only requires first-order differentiability of $\bar{\mu}(\alpha) - \underline{\mu}(\alpha)$ at $\alpha = c_l$. If the LMF and UMF are continuous Gaussian functions, then $\bar{\mu}(\alpha) - \underline{\mu}(\alpha)$ will always be first-order differentiable at $\alpha = c_l$. If, on the other hand, the LMF and the UMF are, e.g., triangular and c_l happens to be at the apex of the triangle, then $\bar{\mu}(\alpha) - \underline{\mu}(\alpha)$ will not be first-order differentiable at $\alpha = c_l$. In such a case, the derivative in (26) should be evaluated either slightly to the left or to the right of $\alpha = c_l$. ■

Corollary 1: When

$$|\alpha - c_l| \leq \left[\frac{6\varepsilon_3}{|d(c_l) - 3s^2(c_l)|}\right]^{1/3} \qquad (27)$$

where ε_3 is a small positive number that denotes the effect of neglecting the third-order term in the Taylor series expansion of $c_l(\alpha)$, then

$$\begin{aligned} c_l(\alpha) &= c_l + \frac{1}{2}s(c_l)(\alpha - c_l)^2 + \varepsilon_3 + \cdots \\ &\approx c_l + \frac{1}{2}s(c_l)(\alpha - c_l)^2 \end{aligned} \qquad (28)$$

■

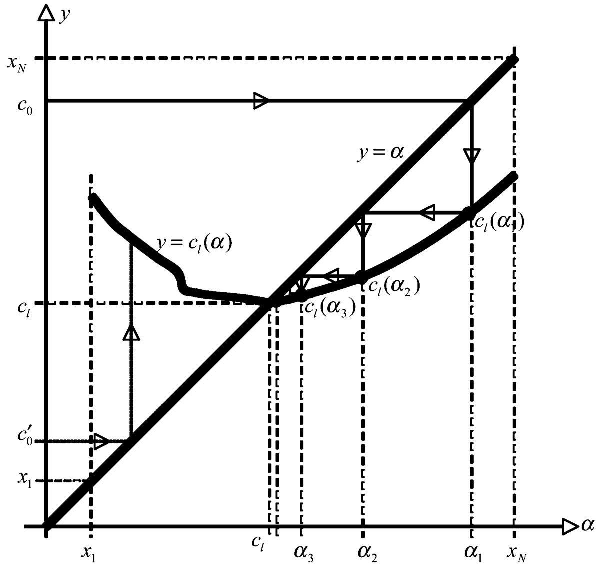

Fig. 1. Graphical interpretation of KM algorithm.

Proof: See Appendix A.

We will make very heavy use of (28) in the sequel. Its validity can be checked through the evaluation of (27). When α satisfies (27) we can state that we are in the *quadratic domain* of $c_l(\alpha)$. In the rest of this paper we assume that a value of ε_3 can be found such that (27) is satisfied.

V. CONVERGENCE PROPERTIES OF THE KM ALGORITHM

As a result of Theorem 2, we can depict $y = c_l(\alpha)$ as in Fig. 1, and can now interpret the KM algorithm graphically as also shown in that figure. To do this, we next explain the trajectory that starts at c_0 and terminates at c_l.

The KM algorithm begins by calculating c_0, using (16). c_0 is then projected horizontally until it intersects the line $y = \alpha$, which establishes that $\alpha_1 = c_0$. The intersection of the vertical line $\alpha = \alpha_1$ with the function $y = c_l(\alpha)$ leads to $y = c_l(\alpha_1)$ [as computed by (18)], which is then projected horizontally until it intersects the line $y = \alpha$, which establishes that $\alpha_2 = c_l(\alpha_1)$ [this is (19)]. The intersection of the vertical line $\alpha = \alpha_2$ with the function $y = c_l(\alpha)$ then leads to $y = c_l(\alpha_2)$, etc. So the KM algorithm can be interpreted in this way as an alternation of horizontal and vertical projections in the $\alpha - c_l(\alpha)$ plane. Even from the representative example that is depicted in Fig. 1, it is clear that convergence of the KM algorithm is quite rapid. Our main goal in this paper is to quantify this.

Theorem 4: The KM algorithms converge *monotonically*. ■

Proof: See Appendix A.

In the proof of Theorem 4, we also explain that, for the initialization of the KM algorithm given in (16), it is true that

$$c_0 \geq c_l. \qquad (29)$$

This means, of course, that c_0 is always above c_l, as shown in Fig. 1. Note that, because c_l is the minimum value of $c_l(\alpha)$, (29) must also be true for any other kind of initialization of the KM algorithm that is based on choosing a switch point (α_0) and using (18) to compute $c_l(\alpha_0)$. Even if c_0 is chosen a priori to lie

---

[7] A necessary and sufficient condition for $y(\theta_1, \ldots, \theta_N)$ to be a convex function of $\theta_1, \ldots, \theta_N$ is that its Hessian matrix, $H(\theta) = [\partial^2 y(\theta)/\partial\theta_i\partial\theta_j]$, must be positive semidefinite [16]. It is straightforward to show that $\partial^2 y(\theta)/\partial\theta_i\partial\theta_j = [2y(\theta) - (x_i + x_j)]/[\sum_{i=1}^{N}\theta_i]^2$. When $N = 2$, $\det H(\theta) = -(x_1 - x_2)^2 < 0$. This counterexample proves that $y(\theta_1, \ldots, \theta_N)$ is not always a convex function.

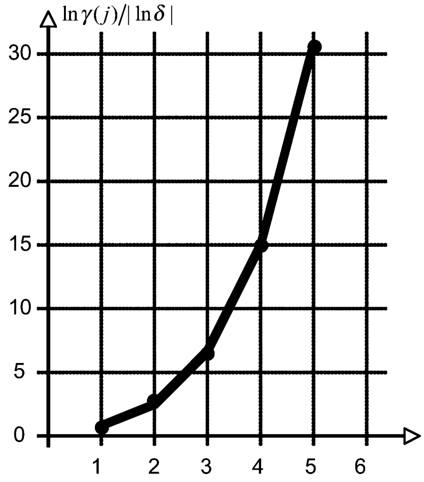

Fig. 2. Normalized convergence factor versus iteration number for the KM algorithm.

below c_l (e.g., $c_0 \equiv c_0'$ in Fig. 1) then after a single iteration of the KM algorithm its associated value of $c_l(\alpha)$ will lie above c_l (see the dotted lines in Fig. 1).

Theorem 5: Let δ be defined as

$$\delta \equiv \frac{c_l(\alpha_1) - c_l}{c_0 - c_l} = \frac{1}{2}s(c_l)(c_0 - c_l) \le 1. \tag{30}$$

When we are in the quadratic domain of $c_l(\alpha)$, it is true that

$$c_l(\alpha_j) = c_l + \frac{1}{2}s(c_l)[c_l(\alpha_{j-1}) - c_l]^2. \tag{31}$$

This is the *fundamental nonlinear iterative equation* of the KM algorithm within its quadratic domain of convergence, whose solution can be expressed as

$$c_l(\alpha_j) = c_l + (c_0 - c_l) \times \delta^{2^j - 1}, \qquad j = 2, 3, \ldots \tag{32}$$

which is indicative of *super-exponential convergence* of the KM algorithm. ■

Proof: See Appendix A.

Let $\gamma(j)$ denote the *convergence factor* of the KM algorithm from one iteration to the next, where

$$\gamma(j) \equiv \delta^{2^j - 1}. \tag{33}$$

Then[8]

$$\frac{\ln\gamma(j)}{|\ln\delta|} = 2^j - 1. \tag{34}$$

A plot of $\ln\gamma(j)/|\ln\delta|$—the *normalized convergence factor*—versus j, given in Fig. 2, is not linear (as it would be if the convergence factor was an exponential function), but is concave upwards. This is indicative of what is referred to as a *super-exponential convergence factor*.

The smaller δ is, the faster will be the convergence of the KM algorithm. Because of the monotonic convergence of the KM algorithm, as c_0 gets closer to c_l, $c_l(\alpha_1)$ gets even closer to c_l. Examining (30), this means that the closer c_0 is to c_l, the smaller δ becomes.

We demonstrate next that δ depends very strongly on the geometry of $FOU(\tilde{A})$. From (30) and (25), observe that

$$\begin{aligned}\delta &= \frac{1}{2}s(c_l)(c_0 - c_l)\\ &= \frac{\bar{\mu}(c_l) - \underline{\mu}(c_l)}{2\left[\int_{-\infty}^{c_l}\bar{\mu}(x)dx + \int_{c_l}^{\infty}\underline{\mu}(x)dx\right]}(c_0 - c_l).\end{aligned} \tag{35}$$

Let A_{LMF} and A_{UMF} denote the areas under the lower and upper MFs of $FOU(\tilde{A})$, i.e.,

$$A_{\mathrm{LMF}} = \int_{-\infty}^{\infty}\underline{\mu}(x)dx \tag{36}$$

$$A_{\mathrm{UMF}} = \int_{-\infty}^{\infty}\bar{\mu}(x)dx. \tag{37}$$

Because $\bar{\mu}(x) \ge \underline{\mu}(x), \forall\, x \in X$, it is straightforward to show that

$$\begin{aligned}&\left[\frac{\bar{\mu}(c_l) - \underline{\mu}(c_l)}{2A_{\mathrm{UMF}}}\right](c_0 - c_l) \le \delta\\ &\quad\le \min\left(\left[\frac{\bar{\mu}(c_l) - \underline{\mu}(c_l)}{2A_{\mathrm{LMF}}}\right](c_0 - c_l), 1\right)\end{aligned} \tag{38}$$

which places the dependency of δ on the geometry of $FOU(\tilde{A})$ in direct evidence.

Corollary 2: When we are in the quadratic domain of $c_l(\alpha)$, then super-exponential convergence occurs to within ε bits of accuracy when

$$|c_l(\alpha_j) - c_l(\alpha_{j-1})| \le \varepsilon. \tag{39}$$

Equation (39) is satisfied by the first integer $j \ge 2$ for which

$$\left|\delta^{2^j} - \delta^{2^{(j-1)}}\right| \le \frac{\delta\varepsilon}{c_0 - c_l}. \tag{40}$$

For small values of δ, we can also determine this first integer as follows: Compute

$$\begin{aligned}j' &= 1 + \frac{1}{\ln 2} \times \ln\left\{\frac{1}{\ln\delta} \times \ln\left[\frac{\varepsilon\delta}{(c_0 - c_l)}\right]\right\}\\ &= 1 + \frac{1}{\ln 2} \times \ln\left\{1 + \frac{1}{\ln\delta} \times \ln\left[\frac{\varepsilon}{(c_0 - c_l)}\right]\right\}\end{aligned} \tag{41}$$

$$j_\varepsilon = \text{first integer larger than } j' \tag{42}$$

and δ is given in (30). ■

Proof: See Appendix A.

Although we felt compelled to state Corollary 2, we hasten to point out that to use its results one not only needs to know the answer, namely c_l, but also $c_l(\alpha_1)$. The latter is only available after the first iteration of the KM algorithm. Hence, the *a priori* use of Corollary 2 is limited.

---

[8] Because $\delta < 1$, $\ln\delta < 0$, which is why we normalize $\ln\gamma(j)$ by $|\ln\delta|$.

TABLE I
EXAMPLE 1 KM ALGORITHM ITERATIONS

N	$c_0 = \alpha_1$	$c_l(\alpha_1)$	$c_l(\alpha_2)$	$c_l(\alpha_3)$	$c_l(\alpha_4)$
20	5.00	3.81	3.60	3.60	
40	5.00	3.82	3.61	3.59	3.59
60	5.00	3.82	3.60	3.59	3.59
80	5.00	3.82	3.60	3.59	3.59
100	5.00	3.82	3.61	3.59	3.59
250	5.00	3.82	3.61	3.59	3.59
500	5.00	3.82	3.61	3.60	3.60
1000	5.00	3.82	3.61	3.60	3.60
10,000	5.00	3.82	3.61	3.60	3.60

Note that (40) is solved by first computing its right-hand side one time, and then computing its left-hand side for $j = 2, 3, \ldots$ until the first value of j is found for which (40) is satisfied.

In the rest of this section, we present three examples that illustrate the monotonic super-exponential convergence of the KM algorithm.

Example 1: Consider a symmetric Gaussian primary MF, $\mu_A(x) = \exp\left[-(1/2)\left((x-5)/\sigma\right)^2\right]$, with uncertain standard deviation, $\sigma \in [0.25, 1.75]$, for which $x \in X = [x_1, x_N] = [0, 10]$, and $\Delta = (x_N - x_1)/N = 10/N$, so that [9], [17]

$$\underline{\mu}_{\tilde{A}}(x) = \exp\left[-\frac{1}{2}\left(\frac{(x-5)}{0.25}\right)^2\right] \tag{43}$$

$$\bar{\mu}_{\tilde{A}}(x) = \exp\left[-\frac{1}{2}\left(\frac{(x-5)}{1.75}\right)^2\right]. \tag{44}$$

Table I depicts iteration results for the KM algorithm, shown to two significant figures, for nine discretizations of $[0, 10]$, ranging from $N = 20$ to $N = 10,000$ samples. Observe the following.

- $c_0 = 5$ regardless of N, because the FOU is symmetrical.
- Convergence of the KM algorithm to $c_l \approx 3.60$ [to two significant figures ($\varepsilon = 10^{-2}$)] is rapid, regardless of N, and occurs in about three iterations.
- The slight differences in final converged values for c_l are due to the sampling interval.
- When we computed j' and j_ε using (41) and (42) respectively, in which we used $c_l = 3.60$ and $c_l(\alpha_1) = 3.82$, we obtained $j' = 2.88$ and $j_\varepsilon = 3$, which agrees with the simulation results.
- In order to see if we were within the quadratic domain of $c_l(\alpha)$, we computed the quadratic and cubic terms of (24) when $\alpha = \alpha_1 = c_0 = 5$ to be 0.46 and -0.11, respectively. Since the cubic term is more than four times smaller than the quadratic term, we are justified in neglecting the cubic term for this example.

■

Example 2: Consider a symmetric Gaussian primary MF, $\mu_A(x) = \exp\left[-(1/2)\,(x-m)^2\right]$, with uncertain mean $m \in [2.4, 7.5]$, for which $x \in X = [x_1, x_N] = [0, 10]$, and $\Delta = (x_N - x_1)/N = 10/N$, so that [9], [17]

$$\underline{\mu}_{\tilde{A}}(x) = \begin{cases} \exp\left[-\frac{1}{2}(x-7.5)^2\right], & x \le 4.98 \\ \exp\left[-\frac{1}{2}(x-2.4)^2\right], & x > 4.98 \end{cases} \tag{45}$$

$$\bar{\mu}_{\tilde{A}}(x) = \begin{cases} \exp\left[-\frac{1}{2}(x-2.4)^2\right], & x < 2.4 \\ 1 & 2.4 \le x \le 7.5 \\ \exp\left[-\frac{1}{2}(x-7.5)^2\right], & x > 7.5. \end{cases} \tag{46}$$

Table II depicts iteration results for the KM algorithm, shown to two significant figures, also for nine discretizations of $[0, 10]$, ranging from $N = 20$ to $N = 10{,}000$ samples. Observe the following.

- $c_0 \approx 4.95$ regardless of N, because the FOU is again symmetrical.
- Convergence of the KM algorithm to $c_l \approx 1.17$ [to two significant figures ($\varepsilon = 10^{-2}$)] is rapid, regardless of N, and occurs in about six iterations.
- The slight differences in final converged values for c_l are due to the sampling interval.
- When we computed j' and j_ε using (41) and (42) respectively, in which we used $c_l = 1.17$ and $c_l(\alpha_1) = 3.02$, we obtained $j' = 4.22$ and $j_\varepsilon = 5$. This is smaller than the actual value of j_ε, which, as can be seen from Table II, equals 6.
- In order to see if we were within the quadratic domain of $c_l(\alpha)$, we again computed the quadratic and cubic terms of (24), but now for $\alpha = \alpha_1 = c_0 = 4.95$, and found them to be 11.94 and -38.5, respectively. Since the cubic term is much larger than the quadratic term, we were not justified in neglecting the cubic term for this example. Even so, our estimated value of $j_\varepsilon = 5$ compares rather well with the actual value of $j_\varepsilon = 6$.

■

Example 3: Consider a nonsymmetrical FOU whose LMF is nonsymmetrical triangular and UMF is nonsymmetrical Gaussian, i.e.,

$$\underline{\mu}_{\tilde{A}}(x) = \begin{cases} \frac{0.6(x+5)}{19} & x \in [-5, 2.6] \\ \frac{0.4(14-x)}{19} & x \in [2.6, 14] \end{cases} \tag{47}$$

$$\bar{\mu}_{\tilde{A}}(x) = \begin{cases} \exp\left[-\frac{1}{2}\left[\frac{(x-2)}{5}\right]^2\right] & x \in [-5, 7.185] \\ \exp\left[-\frac{1}{2}\left[\frac{(x-9)}{1.75}\right]^2\right] & x \in [7.185, 14] \end{cases}. \tag{48}$$

Additionally, $x \in X = [x_1, x_N] = [-5, 14]$ and $\Delta = (x_N - x_1)/N = 19/N$. Table III depicts iteration results for the KM algorithm, shown to two significant figures, also for nine discretizations of $[-5, 14]$, ranging again from $N = 20$ to $N = 10{,}000$ samples. Observe the following.

- c_0 is not the same for all N, because the FOU is unsymmetrical, and sampling seems to make a difference for an unsymmetrical FOU whereas it does not for a symmetrical FOU; however, when the sampling becomes fine enough (e.g., $N \ge 500$) then $c_0 \approx 3.71$ for all such subsequent small values.
- c_l (which ranges from 0.26 to 0.45) is more dependent upon N in this example, because of the non-symmetric nature of the FOU, but regardless of N convergence of the KM algorithm [to two significant figures ($\varepsilon = 10^{-2}$)] is rapid.
- When we computed j' and j_ε using (41) and (42) respectively, in which we used $c_l = 0.45$ and $c_l(\alpha_1) = 0.95$, we

TABLE II
EXAMPLE 2 KM ALGORITHM ITERATIONS

N	$c_0=\alpha_1$	$c_l(\alpha_1)$	$c_l(\alpha_2)$	$c_l(\alpha_3)$	$c_l(\alpha_4)$	$c_l(\alpha_5)$	$c_l(\alpha_6)$	$c_l(\alpha_7)$
20	4.95	2.90	1.88	1.32	1.14	1.14		
40	4.95	2.97	1.95	1.40	1.18	1.14	1.14	
60	4.95	3.00	1.98	1.43	1.20	1.16	1.15	1.15
80	4.95	3.01	2.05	1.51	1.26	1.17	1.16	1.16
100	4.95	3.02	2.05	1.51	1.25	1.17	1.16	1.16
250	4.95	3.02	2.04	1.50	1.24	1.17	1.16	1.17
500	4.95	3.02	2.03	1.50	1.24	1.17	1.17	1.17
1000	4.95	3.02	2.04	1.50	1.24	1.18	1.17	1.17
10,000	4.95	3.02	2.04	1.50	1.24	1.18	1.17	1.17

TABLE III
EXAMPLE 3 KM ALGORITHM ITERATIONS

N	$c_0=\alpha_1$	$c_l(\alpha_1)$	$c_l(\alpha_2)$	$c_l(\alpha_3)$	$c_l(\alpha_4)$
20	3.52	0.76	0.26	0.26	
40	3.64	0.83	0.36	0.36	
60	3.68	0.86	0.41	0.39	0.39
80	3.69	0.93	0.42	0.40	0.40
100	3.69	0.93	0.42	0.41	0.41
200	3.70	0.94	0.45	0.43	0.43
500	3.71	0.94	0.46	0.44	0.44
1000	3.71	0.95	0.46	0.44	0.44
10,000	3.71	0.95	0.46	0.45	0.45

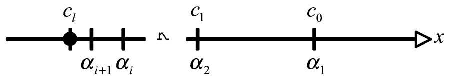

Fig. 3. Relative locations of the α_i in relation to c_0 and c_l.

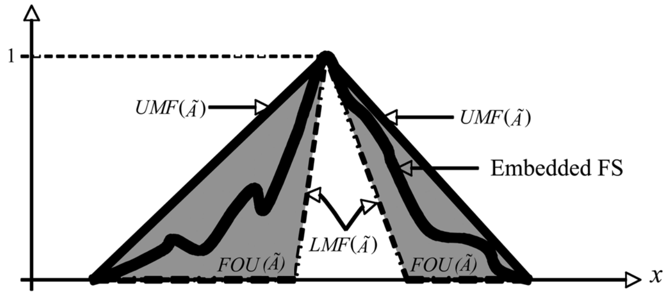

Fig. 4. FOU (shaded), LMF (dashed), UMF (solid), and an embedded FS (wavy line) for IT2 FS $\tilde{A}$.

obtained $j' = 3.03$ and $j_\varepsilon = 4$, which agrees with the simulation results.

- In order to see if we were within the quadratic domain of $c_l(\alpha)$, we again computed the quadratic and cubic terms of (24), when $\alpha = \alpha_1 = c_0 = 3.71$, to be 0.75 and -0.29, respectively. Since the cubic term is about three times smaller than the quadratic term, we are again justified in neglecting the cubic term for this example.

∎

We performed the same sort of simulations for five other examples and in all cases our estimated value for j_ε agreed with the actual value of j_ε, and the cubic term in (24) was considerably smaller than the quadratic term, so that we were justified in neglecting the cubic term for those examples.

VI. APPLICABILITY OF RESULTS

The results presented previously, in which we have used the continuous version of the KM algorithms, are clearly applicable when we begin with $FOU(\tilde{A})$ and wish to compute its centroid, because to do this we can discretize (sample) the primary variable (x) as finely as we wish. This means that in (2) and (3), there is a natural way to make all $x_{i+1} - x_i = T_s \to 0$, so that going from the discrete centroid to the continuous centroid is legitimate. Alternatively, (15) follows directly from (20) by a straightforward discretization of the latter equation in which $x_{i+1} - x_i = T_s$.

We now explain how our analysis is also applicable to the computation of the so-called *generalized centroid* (GC), which, as mentioned in Section I, is widely used in type-reduction of an interval T2 FLS. For the GC we must again compute c_l and c_r in (2) and (3), but now the x_i do *not* correspond to a sampling of the primary variable. Instead x_i are ordered values of a sequence of N positive numbers (e.g., lower MF values of rule-consequent interval T2 FSs), such that $x_i < x_2 < \cdots < x_N$ but $x_{i+1} - x_i \neq T_s$. In this case, in order to convert the discrete GC to a continuous GC, we can always find a small positive number T' such that each x_i is an integer multiple (which could be rather large) of it. We can then insert integer multiples of T' between all x_i and associate zero θ_l values at all of those points. By this construction, the GC is converted into the centroid of a fictional interval T2 FS, most of whose θ_l values are zero, and we can therefore apply the continuous analyses of this paper to using the KM algorithms for computing the GC.

VII. CONCLUSION

As noted in the Introduction, the centroid (or type-reduction) is used in IT2 FLSs and in going from MF data collected from a group of subjects to the FOU of an IT2 FS, and can even be used to compute the FWA. Although computing the centroid of an interval T2 FS is important, it is potentially a time-consuming operation. The KM iterative algorithms are widely used for computing the centroid. In this paper we have proven that these algorithms converge monotonically and super-exponentially fast. Both properties are highly desirable for iterative algorithms and explain why in practice the KM algorithms have been observed to converge very fast, thereby making them very practical to use.

An open problem is to find an optimal way to initialize the KM algorithm, optimal in the sense that such a value of c_0 would lead to the smallest value of δ without *a priori* knowledge of c_l.

APPENDIX A
PROOFS OF THEOREMS

Proof of Theorem 1: Because the proof of Theorem 1 only appears in [22], which (as of Nov. 1, 2005) is not yet published, we provide a condensed version of it here. The proof of (22), due to Mendel and Wu, proceeds in two steps.

- *Step 1*: We show that c_l satisfies the following equation:

$$[\bar{\mu}(c_l) - \underline{\mu}(c_l)]\left\{c_l\left[\int_{-\infty}^{c_l}\bar{\mu}(x)dx + \int_{c_l}^{\infty}\underline{\mu}(x)dx\right] - \left[\int_{-\infty}^{c_l}x\bar{\mu}(x)dx + \int_{c_l}^{\infty}x\underline{\mu}(x)dx\right]\right\} = 0. \quad \text{(A-1)}$$

- *Step 2*: We show that c_l can be computed using (22).

Step 1: A necessary condition for finding $\min_\alpha c_l(\alpha)$ at point $\alpha = c_l$ is that the derivative of $c_l(\alpha)$ with respect to α must be zero when evaluated at c_l, i.e.,

$$\frac{d}{d\alpha}\left\{\frac{\int_{-\infty}^{\alpha}x\bar{\mu}(x)dx + \int_{\alpha}^{\infty}x\underline{\mu}(x)dx}{\int_{-\infty}^{\alpha}\bar{\mu}(x)dx + \int_{\alpha}^{\infty}\underline{\mu}(x)dx}\right\}\Bigg|_{\alpha=c_l} = 0. \quad \text{(A-2)}$$

This equation expands to

$$\frac{[c_l\bar{\mu}(c_l) - c_l\underline{\mu}(c_l)]\left[\int_{-\infty}^{c_l}\bar{\mu}(x)dx + \int_{c_l}^{\infty}\underline{\mu}(x)dx\right]}{\left[\int_{-\infty}^{c_l}\bar{\mu}(x)dx + \int_{c_l}^{\infty}\underline{\mu}(x)dx\right]^2} - \frac{[\bar{\mu}(c_l) - \underline{\mu}(c_l)]\left[\int_{-\infty}^{c_l}x\bar{\mu}(x)dx + \int_{c_l}^{\infty}x\underline{\mu}(x)dx\right]}{\left[\int_{-\infty}^{c_l}\bar{\mu}(x)dx + \int_{c_l}^{\infty}\underline{\mu}(x)dx\right]^2} = 0 \quad \text{(A-3)}$$

from which it follows that

$$[\bar{\mu}(c_l) - \underline{\mu}(c_l)]\left\{c_l\left[\int_{-\infty}^{c_l}\bar{\mu}(x)dx + \int_{c_l}^{\infty}\underline{\mu}(x)dx\right] - \left[\int_{-\infty}^{c_l}x\bar{\mu}(x)dx + \int_{c_l}^{\infty}x\underline{\mu}(x)dx\right]\right\} = 0 \quad \text{(A-4)}$$

which is (A-1).

Step 2: Let $\alpha_A \in X$ for which

$$\bar{\mu}(\alpha_A) - \underline{\mu}(\alpha_A) = 0 \quad \text{(A-5)}$$

and $\alpha_B \in X$ for which

$$\alpha_B\left[\int_{-\infty}^{\alpha_B}\bar{\mu}(x)dx + \int_{\alpha_B}^{\infty}\underline{\mu}(x)dx\right] = \left[\int_{-\infty}^{\alpha_B}x\bar{\mu}(x)dx + \int_{\alpha_B}^{\infty}x\underline{\mu}(x)dx\right]. \quad \text{(A-6)}$$

Observe that (A-6) can be solved for α_B, as

$$\alpha_B = \frac{\int_{-\infty}^{\alpha_B}x\bar{\mu}(x)dx + \int_{\alpha_B}^{\infty}x\underline{\mu}(x)dx}{\int_{-\infty}^{\alpha_B}\bar{\mu}(x)dx + \int_{\alpha_B}^{\infty}\underline{\mu}(x)dx}. \quad \text{(A-7)}$$

Since both α_A and α_B satisfy (A-1), either or both of them may be c_l. We now show that

$$c_l(\alpha_i) \geq c_l(\alpha_B) \ \text{ for } \forall\ \alpha_i \neq \alpha_B. \quad \text{(A-8)}$$

Because c_l is the minimum of $c_l(\alpha)$, it therefore cannot be $c_l(\alpha_A)$ [unless $c_l(\alpha_A) = c_l(\alpha_B)$], but must be $c_l(\alpha_B)$. If it happens that $\alpha_A = \alpha_B = c_l$, then it may happen that $\bar{\mu}(c_l) = \underline{\mu}(c_l)$, in which case (A-1) is simultaneously satisfied by *both* of its terms equaling zero. By these arguments we see that (A-1) can never be satisfied by $\bar{\mu}(c_l) = \underline{\mu}(c_l)$ alone. Note that the condition $\bar{\mu}(c_l) = \underline{\mu}(c_l)$ means that at $x = c_l$ the upper and lower MFs touch each other, something that is perfectly permissible in the $FOU(\tilde{A})$.

Returning to the proof of (A-8), we first consider the case when $\alpha_A < \alpha_B$ for which $c_l(\alpha_A)$ can be re-expressed as shown in (A-9) at the bottom of the page. In obtaining the last line, we have substituted the left-hand side of (A-6) for the right-hand side of (A-6), where the latter appears in the numerator of the third line of (A-9).

Because $\alpha_A < \alpha_B$ and it is always true that $\bar{\mu}(x) - \underline{\mu}(x) \geq 0$, it follows that:

$$\int_{\alpha_A}^{\alpha_B}x\left[\bar{\mu}(x) - \underline{\mu}(x)\right]dx \leq \alpha_B\int_{\alpha_A}^{\alpha_B}\left[\bar{\mu}(x) - \underline{\mu}(x)\right]dx. \quad \text{(A-10)}$$

$$\begin{aligned} c_l(\alpha_A) &= \frac{\int_{-\infty}^{\alpha_A}x\bar{\mu}(x)dx + \int_{\alpha_A}^{\infty}x\underline{\mu}(x)dx}{\int_{-\infty}^{\alpha_A}\bar{\mu}(x)dx + \int_{\alpha_A}^{\infty}\underline{\mu}(x)dx} \\ &= \frac{\int_{-\infty}^{\alpha_B}x\bar{\mu}(x)dx - \int_{\alpha_A}^{\alpha_B}x\bar{\mu}(x)dx + \int_{\alpha_B}^{\infty}x\underline{\mu}(x)dx + \int_{\alpha_A}^{\alpha_B}x\underline{\mu}(x)dx}{\int_{-\infty}^{\alpha_B}\bar{\mu}(x)dx - \int_{\alpha_A}^{\alpha_B}\bar{\mu}(x)dx + \int_{\alpha_B}^{\infty}\underline{\mu}(x)dx + \int_{\alpha_A}^{\alpha_B}\underline{\mu}(x)dx} \\ &= \frac{\int_{-\infty}^{\alpha_B}x\bar{\mu}(x)dx + \int_{\alpha_B}^{\infty}x\underline{\mu}(x)dx - \int_{\alpha_A}^{\alpha_B}x\left[\bar{\mu}(x) - \underline{\mu}(x)\right]dx}{\int_{-\infty}^{\alpha_B}\bar{\mu}(x)dx + \int_{\alpha_B}^{\infty}\underline{\mu}(x)dx - \int_{\alpha_A}^{\alpha_B}\left[\bar{\mu}(x) - \underline{\mu}(x)\right]dx} \\ &= \frac{\alpha_B\left[\int_{-\infty}^{\alpha_B}\bar{\mu}(x)dx + \int_{\alpha_B}^{\infty}\underline{\mu}(x)dx\right] - \int_{\alpha_A}^{\alpha_B}x\left[\bar{\mu}(x) - \underline{\mu}(x)\right]dx}{\int_{-\infty}^{\alpha_B}\bar{\mu}(x)dx + \int_{\alpha_B}^{\infty}\underline{\mu}(x)dx - \int_{\alpha_A}^{\alpha_B}\left[\bar{\mu}(x) - \underline{\mu}(x)\right]dx} \end{aligned} \quad \text{(A-9)}$$

Upon substitution of the upper bound (A-10) into the numerator of (A-9), we see that

$$\begin{aligned} c_l(\alpha_A) &\\ \geq & \frac{\alpha_B\left[\int_{-\infty}^{\alpha_B}\bar{\mu}(x)dx+\int_{\alpha_B}^{\infty}\underline{\mu}(x)dx\right]}{\int_{-\infty}^{\alpha_B}\bar{\mu}(x)dx+\int_{\alpha_B}^{\infty}\underline{\mu}(x)dx-\int_{\alpha_B}^{\alpha_A}\left[\bar{\mu}(x)-\underline{\mu}(x)\right]dx}\\ &-\frac{\alpha_B\int_{\alpha_B}^{\alpha_A}\left[\bar{\mu}(x)-\underline{\mu}(x)\right]dx}{\int_{-\infty}^{\alpha_B}\bar{\mu}(x)dx+\int_{\alpha_B}^{\infty}\underline{\mu}(x)dx-\int_{\alpha_B}^{\alpha_A}\left[\bar{\mu}(x)-\underline{\mu}(x)\right]dx}\\ =&\ \alpha_B=c_l(\alpha_B) \end{aligned} \tag{A-11}$$

where the last part of (A-11) follows from (A-7) and (18). This completes the proof of (A-8) when $\alpha_A<\alpha_B$.

Because the proof of (A-8) when $\alpha_A>\alpha_B$ is so similar to the proof just given when $\alpha_A<\alpha_B$ we leave its details to the reader. Note though that instead of (A-10), we must now use

$$\int_{\alpha_B}^{\alpha_A}x\left[\bar{\mu}(x)-\underline{\mu}(x)\right]dx\geq\alpha_B\int_{\alpha_B}^{\alpha_A}\left[\bar{\mu}(x)-\underline{\mu}(x)\right]dx. \tag{A-12}$$

Equation (A-11) and its counterpart for $\alpha_A>\alpha_B$ together prove the truth of (A-8). Consequently, it is only α_B that is the legitimate solution of (A-1) and, therefore, $\alpha_B=c_l$, where c_l is given by (22).

Proof of Theorem 2: Because c_l is the minimum of $c_l(\alpha)$, it is true that

$$c_l(\alpha)\geq c_l\ \text{ for }\forall\,\alpha \tag{A-13}$$

hence

$$c_l(\alpha)\geq c_l>\alpha,\ \text{ when }\ \alpha<c_l. \tag{A-14}$$

This completes the proof of the first row of (23a). Next we focus on the second row of (23a).

Beginning with (22), we see that

$$c_l\left[\int_{-\infty}^{c_l}\bar{\mu}(x)dx+\int_{c_l}^{\infty}\underline{\mu}(x)dx\right]=\int_{-\infty}^{c_l}x\bar{\mu}(x)dx+\int_{c_l}^{\infty}x\underline{\mu}(x)dx. \tag{A-15}$$

Let $\alpha>c_l$ and add $\int_{c_l}^{\alpha}x\left(\bar{\mu}(x)-\underline{\mu}(x)\right)dx$ to both sides of (A-15), in order to see that

$$c_l\left[\int_{-\infty}^{c_l}\bar{\mu}(x)dx+\int_{c_l}^{\infty}\underline{\mu}(x)dx\right]+\int_{c_l}^{\alpha}x\left(\bar{\mu}(x)-\underline{\mu}(x)\right)dx=\int_{-\infty}^{\alpha}x\bar{\mu}(x)dx+\int_{\alpha}^{\infty}x\underline{\mu}(x)dx \tag{A-16}$$

Because $\alpha>c_l$, and, $\int_{-\infty}^{c_l}\bar{\mu}(x)dx+\int_{c_l}^{\infty}\underline{\mu}(x)dx>0$ and $\bar{\mu}(x)-\underline{\mu}(x)\geq0$, we obtain the following inequality from (A-16):

$$\begin{aligned}\int_{-\infty}^{\alpha}x\bar{\mu}(x)dx+\int_{\alpha}^{\infty}x\underline{\mu}(x)dx &<\alpha\left[\int_{-\infty}^{c_l}\bar{\mu}(x)dx+\int_{c_l}^{\infty}\underline{\mu}(x)dx\right]\\ &\quad+\int_{c_l}^{\alpha}\alpha\left(\bar{\mu}(x)-\underline{\mu}(x)\right)dx\\ &=\alpha\left[\int_{-\infty}^{\alpha}\bar{\mu}(x)dx+\int_{\alpha}^{\infty}\underline{\mu}(x)dx\right].\end{aligned} \tag{A-17}$$

Because $\int_{-\infty}^{\alpha}\bar{\mu}(x)dx+\int_{\alpha}^{\infty}\underline{\mu}(x)dx>0$, (A-17) can be expressed as

$$\alpha>\frac{\int_{-\infty}^{\alpha}x\bar{\mu}(x)dx+\int_{\alpha}^{\infty}x\underline{\mu}(x)dx}{\int_{-\infty}^{\alpha}\bar{\mu}(x)dx+\int_{\alpha}^{\infty}\underline{\mu}(x)dx}=c_l(\alpha) \tag{A-18}$$

which completes the proof for the second row of (23a).

To prove (23b), we begin with (A-2) and (A-3) before α is replaced by c_l [i.e., in (A-3) replace each c_l by α], in which case it is straightforward to show that

$$\frac{dc_l(\alpha)}{d\alpha}=\frac{\left[\bar{\mu}(\alpha)-\underline{\mu}(\alpha)\right]\left[\alpha-c_l(\alpha)\right]}{\int_{-\infty}^{c_l}\bar{\mu}(x)dx+\int_{c_l}^{\infty}\underline{\mu}(x)dx}. \tag{A-19}$$

Because $\bar{\mu}(\alpha)-\underline{\mu}(\alpha)\geq0$, and (23a) is true, we obtain (23b) from an analysis of (A-19) for $\alpha\leq c_l$ and $\alpha>c_l$.

Proof of Theorem 3: The Taylor series expansion of $c_l(\alpha)$ about its minimum value $\alpha^*=c_l$ is

$$\begin{aligned}c_l(\alpha)=&\,c_l(\alpha^*)+\left.\frac{\partial}{\partial\alpha}c_l(\alpha)\right|_{\alpha=\alpha^*}(\alpha-\alpha^*)\\ &+\frac{1}{2}\left.\frac{\partial^2}{\partial\alpha^2}c_l(\alpha)\right|_{\alpha=\alpha^*}(\alpha-\alpha^*)^2\\ &+\frac{1}{3!}\left.\frac{\partial^3}{\partial\alpha^3}c_l(\alpha)\right|_{\alpha=\alpha^*}(\alpha-\alpha^*)^3+\cdots.\end{aligned} \tag{A-20}$$

Since

$$\left.\frac{\partial c_l(\alpha)}{\partial\alpha}\right|_{\alpha=\alpha^*}=0. \tag{A-21}$$

(A-20) simplifies to

$$\begin{aligned}c_l(\alpha)=c_l(\alpha^*)&+\frac{1}{2}\left.\frac{\partial^2}{\partial\alpha^2}c_l(\alpha)\right|_{\alpha=\alpha^*}(\alpha-\alpha^*)^2\\ &+\frac{1}{3!}\left.\frac{\partial^3}{\partial\alpha^3}c_l(\alpha)\right|_{\alpha=\alpha^*}(a-\alpha^*)^3+\cdots.\end{aligned} \tag{A-22}$$

Because the two calculations on the right-hand side of (A-22) are straightforward exercises in calculus, and because of space limitations, we leave their details to the reader. However, here we would like to mention a couple of things about differentiability at $\alpha=c_l$.

The details of computing $\left[\partial^2 c_l(\alpha)/\partial\alpha^2\right]|_{\alpha=\alpha^*}$ show that terms involving the first derivative of $\bar{\mu}(\alpha) - \underline{\mu}(\alpha)$ always equal zero at $\alpha = \alpha^*$ [regardless of whether or not $\bar{\mu}(\alpha)$ and $\underline{\mu}(\alpha)$ are differentiable at $\alpha = \alpha^*$] because they are multiplied by a common factor that equals zero due to (22) being true when $\alpha = \alpha^*$. On the other hand, the details of computing $\left[\partial^3 c_l(\alpha)/\partial\alpha^3\right]|_{\alpha=\alpha^*}$ show that the derivative $\partial\left[\bar{\mu}(\alpha) - \underline{\mu}(\alpha)\right]/\partial\alpha$ does occur, but all second-derivative terms of $\bar{\mu}(\alpha) - \underline{\mu}(\alpha)$ are always equal to zero at $\alpha = \alpha^*$ for exactly the same reason just given about the computation of $\left[\partial^2 c_l(\alpha)/\partial\alpha^2\right]|_{\alpha=\alpha^*}$. See the comment after Theorem 3 for discussions about the differentiability of $\bar{\mu}(\alpha)-\underline{\mu}(\alpha)$ at $\alpha = c_l$.

Proof of Corollary 1: We choose ε_3 such that in (24)

$$\left|\frac{1}{3!}\left[d(c_l) - 3s^2(c_l)\right](\alpha - c_l)^3\right| \le \varepsilon_3. \tag{A-23}$$

Note that it is possible for $d(c_l) - 3s^2(c_l)$ to be positive or negative; hence, the use of the absolute value sign in (A-23). Equation (27) is a direct consequence of (A-23).

Proof of Theorem 4: Our proof is for the continuous version of the KM algorithm (see Section III), which finds c_l. We will show that $c_0 \ge c_l(\alpha_1) \ge c_l(\alpha_2) \ge \cdots \ge c_l(\alpha_i) \ge \cdots \ge c_l$.

Because c_0, as computed in (16), can be interpreted in the framework of the minimization problem in (2) for specific choices of the θ_i [in (2), the θ_i are not restricted to just the LMF and UMF values; and, the average of those values lies in the allowable interval for each of the θ_i], it must be true that $c_0 \ge c_l$. If $c_0 = c_l$ then, according to Theorem 1, c_0 is the minimum of $c_l(\alpha)$. On the other hand, if $c_0 > c_l$, then, according to the second line of (23a) (with $\alpha_1 = c_0$), we see that $c_0 > c_l(c_0) = c_l(\alpha_1)$. Combining these two cases, we see that $c_0 \ge c_l(\alpha_1)$.

From (19) of the KM algorithm α_2 is chosen as $\alpha_2 = c_l(\alpha_1)$. If α_2 equals the minimum value of $c_l(\alpha)$ then $\alpha_2 = c_l$. On the other hand, if $\alpha_2 > c_l$, then, according to the second line of (23a) [with $\alpha_2 = c_l(\alpha_1)$], we see that $\alpha_2 > c_l(\alpha_2)$ or, equivalently, that $c_l(\alpha_1) > c_l(\alpha_2)$. Combining these two cases, we see that $\alpha_2 \ge c_l(\alpha_2)$ or, equivalently, that $c_l(\alpha_1) \ge c_l(\alpha_2)$. From the first part of this proof, we now see that $c_0 \ge c_l(\alpha_1) \ge c_l(\alpha_2)$.

Continuing in this same way, it is straightforward to show that (see Fig. 3)

$$c_0 \ge c_l(\alpha_1) \ge c_l(\alpha_2) \ge \cdots \ge c_l(\alpha_i) \ge \cdots \ge c_l. \tag{A-24}$$

This means, of course, that the KM algorithm is monotonically convergent. That (A-24) contains a finite number of steps has already been proved by Karnik and Mendel [7], with their very conservative bound of N steps.

Proof of Theorem 5: Because the KM algorithm converges monotonically

$$c_l(\alpha_1) - c_l \le c_0 - c_l \tag{A-25}$$

hence

$$\delta \equiv \frac{[c_l(\alpha_1) - c_l]}{(c_0 - c_l)} \le 1. \tag{A-26}$$

Assuming we are in the quadratic domain of $c_l(\alpha)$, so that (28) holds, observe from (28) that:

$$\frac{[c_l(\alpha_1) - c_l]}{(\alpha_1 - c_l)} = \frac{1}{2}s(c_l)(\alpha_1 - c_l). \tag{A-27}$$

But, $\alpha_1 = c_0$; hence, the left-hand side of (A-27) is δ and, therefore

$$\delta = \frac{1}{2}s(c_l)(c_0 - c_l). \tag{A-28}$$

Based on (28) and (19) (which is a key equation in the KM algorithm), we see that at the jth iteration of the KM algorithm $(j = 2, 3, \ldots)$

$$\begin{aligned} c_l(\alpha_j) &= c_l + \frac{1}{2}s(c_l)(\alpha_j - c_l)^2 \\ &= c_l + \frac{1}{2}s(c_l)[c_l(\alpha_{j-1}) - c_l]^2 \end{aligned} \tag{A-29}$$

which is the *fundamental nonlinear iterative equation* of the KM algorithm within its quadratic domain of convergence. We now iterate (A-29) using (A-26) [solved for $c_l(\alpha_1) - c_l$] and (A-28). For $j = 2$

$$\begin{aligned} c_l(\alpha_2) &= c_l + \frac{1}{2}s(c_l)[c_l(\alpha_1) - c_l]^2 \\ &= c_l + \frac{1}{2}s(c_l)[\delta \times (c_0 - c_l)]^2 \\ &= c_l + \delta^2 \times \left[\frac{1}{2}s(c_l)(c_0 - c_l)\right] \times (c_0 - c_l) \\ &= c_l + \delta^3 \times (c_0 - c_l). \end{aligned} \tag{A-30}$$

For $j = 3$, (A-29) becomes:

$$\begin{aligned} c_l(\alpha_3) &= c_l + \frac{1}{2}s(c_l)[c_l(\alpha_2) - c_l]^2 \\ &= c_l + \frac{1}{2}s(c_l)\left[\delta^3 \times (c_0 - c_l)\right]^2 \\ &= c_l + \delta^6 \times \left[\frac{1}{2}s(c_l)(c_0 - c_l)\right] \times (c_0 - c_l) \\ &= c_l + \delta^7 \times (c_0 - c_l). \end{aligned} \tag{A-31}$$

Continuing in this manner, we obtain (32). Observe that when $j = 2$, $2^j - 1 = 3$, and when $j = 3$, $2^j - 1 = 7$, as in (A-30) and (A-31), respectively.

Proof of Corollary 2: Substituting (32) into (39) for both $c_l(\alpha_j)$ and $c_l(\alpha_{j-1})$, we obtain

$$\left|\delta^{2^j-1}(c_0 - c_l) - \delta^{2^{(j-1)}-1}(c_0 - c_l)\right| \le \varepsilon. \tag{A-32}$$

This can be expressed as

$$\delta^{-1}\left|\delta^{2^{j}}-\delta^{2^{(j-1)}}\right|(c_0-c_l)\le\varepsilon \tag{A-33}$$

from which we obtain (40). Note that because $\delta^{2^{j}}<\delta^{2^{(j-1)}}$ for all values of j, we use the absolute signs in (40).

If δ is relatively small (e.g., $\le 1/2$) then $\delta^{2^{j}}\ll\delta^{2^{(j-1)}}$ even for very small values of j. In this case, we can approximate (A-32) as

$$\left|\delta^{2^{(j'-1)}-1}\right|(c_0-c_l)\le\varepsilon. \tag{A-34}$$

Solving this equation for j' we obtain (41). Because j' is not necessarily an integer, we then need to choose j_ε as the first integer larger than j'.

Appendix B
Basics of Interval Type -2 Fuzzy Sets

An *interval T2 FS* $\tilde{A}$ is characterized as[9] [17], [18]

$$\tilde{A}=\int_{x\in X}\int_{u\in J_x\subseteq[0,1]}1/(x,u)=\int_{x\in X}\left[\int_{u\in J_x\subseteq[0,1]}1/u\right]\Bigg/x \tag{B-1}$$

x, the *primary variable*, has domain X; u, the *secondary variable*, has domain J_x at each $x\in X$; J_x is called the *primary membership* of x; and, the *secondary grades* of $\tilde{A}$ all[10] equal 1. Uncertainty about $\tilde{A}$ is conveyed by the union of all of the primary memberships, which is called the FOU of $\tilde{A}$, i.e.,

$$\mathrm{FOU}(\tilde{A})=\bigcup\nolimits_{x\in X}J_x. \tag{B-2}$$

The UMF and LMF of $\tilde{A}$ are two type-1 MFs that bound the FOU (e.g., see Fig. 4). The UMF is associated with the upper bound of $\mathrm{FOU}(\tilde{A})$ and is denoted $\bar{\mu}_{\tilde{A}}(x)$, $\forall\, x\in X$, and the LMF is associated with the lower bound of $\mathrm{FOU}(\tilde{A})$ and is denoted $\underline{\mu}_{\tilde{A}}(x)$, $\forall\, x\in X$. In this paper we sometimes shorten $\bar{\mu}_{\tilde{A}}(x)$ and $\underline{\mu}_{\tilde{A}}(x)$ to $\bar{\mu}(x)$ and $\underline{\mu}(x)$, respectively. Note that

$$J_x=\left[\underline{\mu}(x),\bar{\mu}(x)\right] \tag{B-3}$$

so that $\mathrm{FOU}(\tilde{A})$ in (B-2) can also be expressed as

$$\mathrm{FOU}(\tilde{A})=\bigcup_{x\in X}\left[\underline{\mu}(x),\bar{\mu}(x)\right]. \tag{B-4}$$

For continuous universes of discourse X and U, an *embedded interval T2 FS* $\tilde{A}$ is

$$\tilde{A}_e=\int_{x\in X}[1/\theta]/x \quad \theta\in J_x\subseteq U=[0,1]. \tag{B-5}$$

Set $\tilde{A}_e$ is embedded in $\tilde{A}$ such that at each x it only has one secondary variable (see Fig. 4 where secondary grades, not shown, are all equal to 1). Although there are an uncountable number of embedded IT2 FSs, such FSs are still quite useful in theoretical developments. Other examples of $\tilde{A}_e$ are $1/\bar{\mu}(x)$ and $1/\underline{\mu}(x)$, $\forall\, x\in X$, where, in this notation it is understood that the secondary grade equals 1 for $\forall\, x\in X$.

Associated with each A_e is an *embedded T1 FS* A_e, where

$$A_e=\int_{x\in X}\theta/x \quad \theta\in J_x\subseteq U=[0,1]. \tag{B-6}$$

Set A_e, which acts as the domain for $\tilde{A}_e$, is the union of all the primary memberships of the set $\tilde{A}_e$ in (B-5) (Fig. 4), and there are an uncountable number of A_e. Other examples of A_e are $\bar{\mu}(x)$ and $\underline{\mu}(x)$, $\forall\, x\in X$.

For discrete universes of discourse, in which both the primary and secondary variables[11] are discretized, there exist a countable number of embedded T2 (and T1) FSs (e.g., see [18] or [17]). In this case, $\tilde{A}_e$ and A_e are given by formulas like (B-8) and (B-11) below.

In [18] a new *Representation Theorem* was derived in which a general T2 FS, $\tilde{A}$, is expressed as the union of all of its embedded T2 FSs, $\tilde{A}_e^j$. For an IT2 FS, for which X and U are discrete, this can be stated as

$$\tilde{A}=\sum_{j=1}^{n_A}\tilde{A}_e^j \tag{B-7}$$

where

$$\tilde{A}_e^j=\sum_{i=1}^{N}\left[1/u_i^j\right]/x_i \quad u_i^j\in J_{x_i}\subseteq U=[0,1] \tag{B-8}$$

and

$$n_A=\prod_{i=1}^{N}M_i \tag{B-9}$$

in which M_i denotes the discretization levels of secondary variable u_i^j at each of the Nx_i. We can also express (B-7) and (B-8) as

$$\tilde{A}=1/FOU(\tilde{A})=1/\sum_{j=1}^{n_A}A_e^j \tag{B-10}$$

where

$$A_e^j=\sum_{i=1}^{N}u_i^j/x_i \quad u_i^j\in J_{x_i}\subseteq U=[0,1] \tag{B-11}$$

---

[9] We will intertwine our discussions about fuzzy sets that are defined on continuous or discrete universes of discourse. For discrete universes of discourse, one should replace the integral signs [e.g., in (B-1)] with summation signs. Regardless of which notation is used, each symbol represents the set-theoretic union operation.

[10] In a general T2 FS, the secondary grades can take on any value in $[0,1]$.

[11] Strictly speaking, a T2 FS whose secondary variable is discrete is not an IT2 FS, because an IT2 FS requires that the domain for the secondary variable must be an interval. It is usually for computational purposes that we discretize the secondary variable.

and it is understood that the notation in (B-10) means that the secondary grade equals 1 at all elements in $FOU(\tilde{A})$. Referring to Fig. 4, (B-10) means collecting all embedded T1 FSs into a *bundle* of such sets. This bundle will always be bounded by the UMF and the LMF of the FOU, since they are both legitimate embedded sets.

REFERENCES

[1] W. M. Dong and F. S. Wong, "Fuzzy weighted averages and implementation of the extension principle," *Fuzzy Sets Syst.*, vol. 21, pp. 183–199, 1987.

[2] Y.-Y. Guh, C.-C. Hon, and E. S. Lee, "Fuzzy weighted average: The linear programming approach via charnes and cooper's rule," *Fuzzy Sets Syst.*, vol. 117, pp. 157–160, 2001.

[3] Y.-Y. Guh, C.-C. Hon, K.-M. Wang, and E. S. Lee, "Fuzzy weighted average: A max-min paired elimination method," *Comput. Math. Applic.*, vol. 32, pp. 115–123, 1996.

[4] H. Hagras, "A hierarchical type-2 fuzzy logic control architecture for autonomous mobile robots," *IEEE Trans. Fuzzy Syst.*, vol. 12, no. 4, pp. 524–539, Aug. 2004.

[5] C. H. Rhee, "An interval type-2 fuzzy spherical shells algorithm," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, Budapest, Hungary, Jul. 2004.

[6] N. N. Karnik and J. M. Mendel, An Introduction to Type-2 Fuzzy Logic Systems 1998, USC Signal Image Process. Rep..

[7] ——, "Centroid of a type-2 fuzzy set," *Inform. Sci.*, vol. 132, pp. 195–220, 2001.

[8] D. H. Lee and D. Park, "An efficient algorithm for the fuzzy weighted average," *Fuzzy Sets Syst.*, vol. 87, pp. 39–45, 1997.

[9] Q. Liang and J. M. Mendel, "Interval type-2 fuzzy logic systems: Theory and design," *IEEE Trans. Fuzzy Syst.*, vol. 8, no. 5, pp. 535–550, Oct. 2000.

[10] ——, "Equalization of nonlinear time-varying channels using type-2 fuzzy adaptive filters," *IEEE Trans. Fuzzy Syst.*, vol. 8, no. 5, pp. 551–563, Oct. 2000.

[11] ——, "Overcoming time-varying co-channel interference using type-2 fuzzy adaptive filter," *IEEE Trans. Circuits Syst.*, vol. 47, no. 12, pp. 1419–1428, Dec. 2000.

[12] ——, "Designing interval type-2 fuzzy logic systems using an SVD-QR method: Rule reduction," *Int. J. Intell. Syst.*, vol. 15, pp. 939–957, 2000.

[13] ——, "Modeling MPEG VBR video traffic using type-2 fuzzy logic systems," in *Granular Computing: An Emerging Paradigm*. New York: Springer-Verlag, 2000.

[14] Q. Liang, N. N. Karnik, and J. M. Mendel, "Connection admission control in atm networks using survey-based type-2 fuzzy logic systems," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 30, no. 4, pp. 329–339, Aug. 2000.

[15] T.-S. Liou and M.-J. J. Wang, "Fuzzy weighted average: An improved algorithm," *Fuzzy Sets Syst.*, vol. 49, pp. 307–315, 1992.

[16] D. L. Luenberger, *Linear and Nonlinear Programming*, 2nd ed. Reading, MA: Addison-Wesley, 1984.

[17] J. M. Mendel, *Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions*. Upper Saddle River, NJ: Prentice-Hall, 2001.

[18] J. M. Mendel and R. I. John, "Type-2 fuzzy sets made simple," *IEEE Trans. Fuzzy Syst.*, vol. 10, no. 2, pp. 117–127, Apr. 2002.

[19] J. M. Mendel and H. Wu, "Centroid uncertainty bounds for interval type-2 fuzzy sets: Forward and inverse problems," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, Budapest, Hungary, Jul. 2004.

[20] ——, "Type-2 fuzzistics for symmetric interval type-2 fuzzy sets: Part 1, forward problems," *IEEE Trans. Fuzzy Syst.*, vol. 14, no. 6, pp. 781–792, Dec. 2006.

[21] ——, "Type-2 fuzzistics for symmetric interval type-2 fuzzy sets: Part 2, inverse problems," *IEEE Trans. Fuzzy Syst.*, 2007, to be published.

[22] ——, "New results about the centroid of an interval type-2 fuzzy set, including the centroid of a fuzzy granule," *Inform. Sci.*, vol. 177, pp. 360–377, 2007.

[23] J. Starczewski and L. Rutkowski, "Interval type-2 neuro-fuzzy systems based on interval consequents," in *Neural Networks and Soft Computing*. Heidelberg, Germany: Physica-Verlag, 2003, pp. 570–577.

[24] H. Wu and J. M. Mendel, "Uncertainty bounds and their use in the design of interval type-2 fuzzy logic systems," *IEEE Trans. Fuzzy Syst.*, vol. 10, no. 5, pp. 622–639, Oct. 2002.

[25] K. C. Wu, "Fuzzy interval control of mobile robots," *Comput. Elect. Eng.*, vol. 22, pp. 211–229, 1996.

Jerry M. Mendel (S'59–M'61–SM'72–F'78–LF'04) received the Ph.D. in electrical engineering from the Polytechnic Institute of Brooklyn, NY.

Currently he is Professor of Electrical Engineering at the University of Southern California, Los Angeles, where he has been since 1974. He has published over 470 technical papers and is author and/or editor of eight books, including *Uncertain Rule-based Fuzzy Logic Systems: Introduction and New Directions* (Prentice-Hall, 2001). His present research interests include: type-2 fuzzy logic systems and their applications to a wide range of problems, including target classification, smart oil field technology, and computing with words.

Dr. Mendel is a Distinguished Member of the IEEE Control Systems Society. He was President of the IEEE Control Systems Society in 1986, and is presently Chairman of the Fuzzy Systems Technical Committee and a member of the Administrative Committee of the IEEE Computational Intelligence Society. 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, the 2002 TRANSACTIONS ON FUZZY SYSTEMS Outstanding Paper Award, a 1984 IEEE Centennial Medal, and an IEEE Third Millenium Medal.

Feilong Liu (S'02) received his B.S. and M.S. in automation theory and engineering from Chinese Northeastern University, Shenyang, Lioaning Province, P. R. China, and South China University of Technology, Guangzhou, Guangdong Province, P. R. China, in 1995 and 2000, respectively. He is currently working toward the Ph.D. degree in Electrical Engineering at the University of Southern California, Los Angeles.

His current research interests include type-2 fuzzy logic theory, artificial intelligence, signal processing, pattern recognition, and applying these technologies to smart oil field problems, such as water flooding.