A Brief Introduction to Type2 Fuzzy Logic

Dongrui Wu
HUST

A Brief Tutorial on Interval Type-2 Fuzzy Sets and Systems

Dongrui Wu

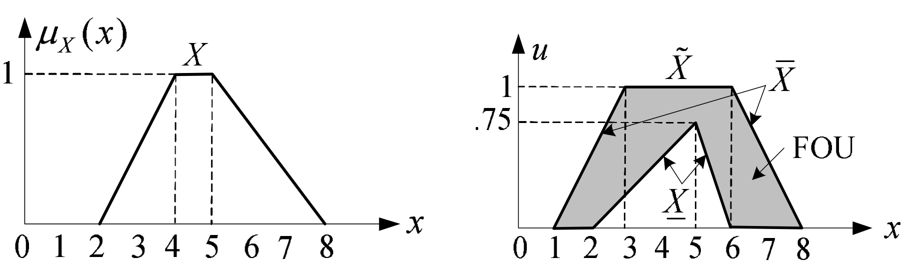
Abstract

This tutorial illustrates the basic ideas of interval type-2 (IT2) fuzzy sets and systems, and provides a Matlab implementation of IT2 fuzzy logic system (FLS). One obstacle in learning IT2 fuzzy logic is its complex notations. In this tutorial we try to avoid these notations and give the reader some intuitive understanding of IT2 FLSs. We also briefly discuss approaches for reducing the computational cost of IT2 FLSs and the fundamental differences between IT2 and type-1 FLSs.

I. Type-1 Fuzzy Sets (T1 FSs)

Type-1 fuzzy set (T1 FS) theory was first introduced by Zadeh [69] in 1965 and has been successfully applied in many areas, including modeling and control [5], [48], [67], data mining [22], [40], [66], time-series prediction [28], [30], [45], etc.

An example of a T1 FS, X, is shown in Fig. 1(a). When only integer numbers are considered in the x domain, the T1 FS can be represented as $\{0/2, 0.5/3, 1/4, 1/5, 0.67/6, 0.33/7, 0/8\}$, where $0/2$ means that number 2 has a *membership degree* of 0 in the T1 FS X, $0.5/3$ means number 3 has a membership degree of 0.5 in the T1 FS X, etc. In contrast, for a crisp set, the membership degree of each element in it can be either 0 or 1; there is no value (e.g., 0.5) in between.



(a) (b)

Fig. 1. Examples of a T1 FS (a) and an IT2 FS (b).

The *membership function* (MF), $\mu_X(x)$, of a T1 FS can either be chosen based on the user's opinion (hence, the MFs from two individuals could be quite different depending upon their experiences, perspectives, cultures, etc.), or, it can be designed using optimization procedures [23], [25], [47].

Dongrui Wu was with the Signal and Image Processing Institute, Ming Hsieh Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089-2564, USA. He is now with the Machine Learning Lab, GE Global Research, Niskayuna, NY 12309 USA (phone: 213-595-3269; email: drwu09@gmail.com).

II. Interval Type-2 Fuzzy Sets (IT2 FSs)

Despite having a name which carries the connotation of uncertainty, research has shown that there are limitations in the ability of T1 FSs to model and minimize the effect of uncertainties [20], [21], [34], [62]. This is because a T1 FS is certain in the sense that its membership grades are crisp values. Recently, type-2 FSs [70], characterized by MFs that are themselves fuzzy, have been attracting interests. Interval type-2 (IT2) FSs[1] [34], a special case of type-2 FSs, are currently the most widely used for their reduced computational cost.

An example of an IT2 FS, $\tilde{X}$, is shown in Fig. 1(b). Observe that unlike a T1 FS, whose membership for each x is a number, the membership of an IT2 FS is an interval. For example, the membership of number 3 is [0.25, 1], and the membership of number 5 is $[0.75, 1]$. Observe also that an IT2 FS is bounded from the above and below by two T1 FSs, $\overline{X}$ and $\underline{X}$, which are called *upper MF* (UMF) and *lower MF* (LMF), respectively. The area between $\overline{X}$ and $\underline{X}$ is the *footprint of uncertainty* (FOU).

IT2 FSs are particularly useful when it is difficult to determine the exact MF, or in modeling the diverse opinions from different individuals. The MFs can be constructed from surveys [31], [37], [56] or using optimization algorithms [62], [63].

III. Interval Type-2 Fuzzy Logic System (IT2 FLS)

Fig. 2 shows the schematic diagram of an IT2 FLS. It is similar to its T1 counterpart, the major difference being that at least one of the FSs in the rule base is an IT2 FS. Hence, the outputs of the inference engine are IT2 FSs, and a type-reducer is needed to convert them into a T1 FS before defuzzification can be carried out.

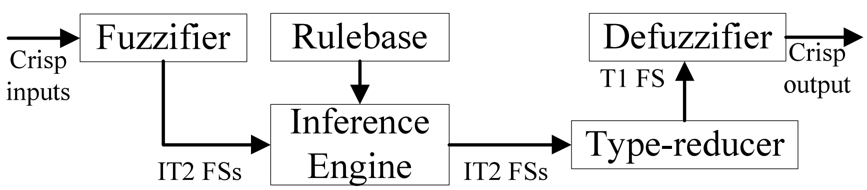


Fig. 2. An IT2 FLS.

In practice the computations in an IT2 FLS can be significantly simplified. Consider the rulebase of an IT2 FLS consisting of N rules assuming the following form:

$$R^n\text{:}\quad \text{IF } x_1 \text{ is } \widetilde{X}_1^n \text{ and } \cdots \text{ and } x_I \text{ is } \widetilde{X}_I^n, \quad \text{THEN } y \text{ is } Y^n \qquad n=1,2,...,N$$

where $\widetilde{X}_i^n$ $(i=1,\ldots,I)$ are IT2 FSs, and $Y^n=[\underline{y}^n,\overline{y}^n]$ is an interval, which can be understood as the centroid [26], [34] of a consequent IT2 FS[2], or the simplest TSK model, for its simplicity. In many applications we use $\underline{y}^n=\overline{y}^n$, i.e., each rule consequent is a crisp number.

Assume the input vector is $\mathbf{x}'=(x_1',x_2',...,x_I')$. Typical computations in an IT2 FLS involve the following steps:

1) Compute the membership of x_i' on each X_i^n, $[\mu_{\underline{X}_i^n}(x_i'),\mu_{\overline{X}_i^n}(x_i')]$, $i=1,2,...,I$, $n=1,2,...,N$.
2) Compute the firing interval of the n^{th} rule, $F^n(\mathbf{x}')$:

$$F^n(\mathbf{x}')=[\mu_{\underline{X}_1^n}(x_1')\times\cdots\times\mu_{\underline{X}_I^n}(x_I'),\mu_{\overline{X}_1^n}(x_1')\times\cdots\times\mu_{\overline{X}_I^n}(x_I')]\equiv[\underline{f}^n,\overline{f}^n],\quad n=1,...,N \tag{1}$$

[1] IT2 FSs have also been called *interval-valued fuzzy sets* in the literature [8], [9], [17], [43], [44]. They can also be mapped into *intuitionistic fuzzy sets* [1]–[3]. Deschrijver and Kerre [13] have a comprehensive study on the relationships some extensions of T1 FSs, including interval-valued FSs, intuitionistic FSs, interval-valued intuitionistic FSs [1], and L-FSs [16].

[2] The rule consequents can be IT2 FSs; however, when the popular center-of-sets type-reduction method [34] is used, these consequent IT2 FSs are replaced by their centroids in the computation; so, it is more convenient to represent the rule consequents as intervals directly.

Note that the minimum, instead of the product, can be used in (1).

3) Perform type-reduction to combine $F^n(\mathbf{x}')$ and the corresponding rule consequents. There are many such methods. The most commonly used one is the center-of-sets type-reducer[3] [34]:

$$Y_{cos}(\mathbf{x}') = \bigcup_{\substack{f^n \in F^n(\mathbf{x}') \\ y^n \in Y^n}} \frac{\sum\limits_{n=1}^{N} f^n y^n}{\sum\limits_{n=1}^{N} f^n} = [y_l,\ y_r] \tag{2}$$

It has been shown that [34], [37], [55]:

$$y_l = \min_{k\in[1,N-1]} \frac{\sum_{n=1}^{k} \overline{f}^n \underline{y}^n + \sum_{n=k+1}^{N} \underline{f}^n \underline{y}^n}{\sum_{n=1}^{k} \overline{f}^n + \sum_{n=k+1}^{N} \underline{f}^n} \equiv \frac{\sum_{n=1}^{L} \overline{f}^n \underline{y}^n + \sum_{n=L+1}^{N} \underline{f}^n \underline{y}^n}{\sum_{n=1}^{L} \overline{f}^n + \sum_{n=L+1}^{N} \underline{f}^n} \tag{3}$$

$$y_r = \max_{k\in[1,N-1]} \frac{\sum_{n=1}^{k} \underline{f}^n \overline{y}^n + \sum_{n=k+1}^{N} \overline{f}^n \overline{y}^n}{\sum_{n=1}^{k} \underline{f}^n + \sum_{n=k+1}^{N} \overline{f}^n} \equiv \frac{\sum_{n=1}^{R} \underline{f}^n \overline{y}^n + \sum_{n=R+1}^{N} \overline{f}^n \overline{y}^n}{\sum_{n=1}^{R} \underline{f}^n + \sum_{n=R+1}^{N} \overline{f}^n} \tag{4}$$

where the *switch points* L and R are determined by

$$\underline{y}^L \leqslant y_l \leqslant \underline{y}^{L+1} \tag{5}$$

$$\overline{y}^R \leqslant y_r \leqslant \overline{y}^{R+1} \tag{6}$$

and $\{\underline{y}^n\}$ and $\{\overline{y}^n\}$ have been sorted in ascending order, respectively.

y_l and y_r can be computed using the Karnik-Mendel (KM) algorithms [34] as follows:

<u>KM Algorithm for Computing y_l:</u>

a) Sort $\underline{y}_n$ $(n = 1, 2, \ldots, N)$ in increasing order and call the sorted $\underline{y}^n$ by the same name, but now $\underline{y}^1 \leqslant \underline{y}^2 \leqslant \cdots \leqslant \underline{y}^N$. Match the weights $F^n(\mathbf{x}')$ with their respective $\underline{y}^n$ and renumber them so that their index corresponds to the renumbered $\underline{y}^n$.

b) Initialize f^n by setting

$$f^n = \frac{\underline{f}^n + \overline{f}^n}{2} \qquad n = 1, 2, \ldots, N \tag{7}$$

and then compute

$$y = \frac{\sum_{n=1}^{N} \underline{y}^n f^n}{\sum_{n=1}^{N} f^n} \tag{8}$$

c) Find switch point k $(1 \leqslant k \leqslant N-1)$ such that

$$\underline{y}^k \leqslant y \leqslant \underline{y}^{k+1} \tag{9}$$

d) Set

$$f^n = \begin{cases} \overline{f}^n, & n \leqslant k \\ \underline{f}^n, & n > k \end{cases} \tag{10}$$

and compute

$$y' = \frac{\sum_{n=1}^{N} \underline{y}^n f^n}{\sum_{n=1}^{N} f^n} \tag{11}$$

e) Check if $y' = y$. If yes, stop and set $y_l = y$ and $L = k$. If no, go to Step 6.

[3] Another popular type-reducer is the centroid type-reduction method [34], which combines all the rule-output IT2 FSs by their union and then finds its centroid as the type-reduced set. Note that this method requires the rule consequents to be IT2 FSs instead of intervals.

f) Set $y = y'$ and go to Step 3.

KM Algorithm for Computing y_r:

a) Sort $\overline{y}^n$ $(n = 1, 2, \ldots, N)$ in increasing order and call the sorted $\overline{y}^n$ by the same name, but now $\overline{y}^1 \leqslant \overline{y}^2 \leqslant \cdots \leqslant \overline{y}^N$. Match the weights $F^n(\mathbf{x}')$ with their respective $\overline{y}^n$ and renumber them so that their index corresponds to the renumbered $\overline{y}^n$.

b) Initialize f^n by setting

$$f^n = \frac{\underline{f}^n + \overline{f}^n}{2} \qquad n = 1, 2, \ldots, N \tag{12}$$

and then compute

$$y = \frac{\sum_{n=1}^{N} \overline{y}^n f^n}{\sum_{n=1}^{N} f^n} \tag{13}$$

c) Find switch point k $(1 \leqslant k \leqslant N-1)$ such that

$$\overline{y}^k \leqslant y \leqslant \overline{y}^{k+1} \tag{14}$$

d) Set

$$f^n = \begin{cases} \underline{f}^n, & n \leqslant k \\ \overline{f}^n, & n > k \end{cases} \tag{15}$$

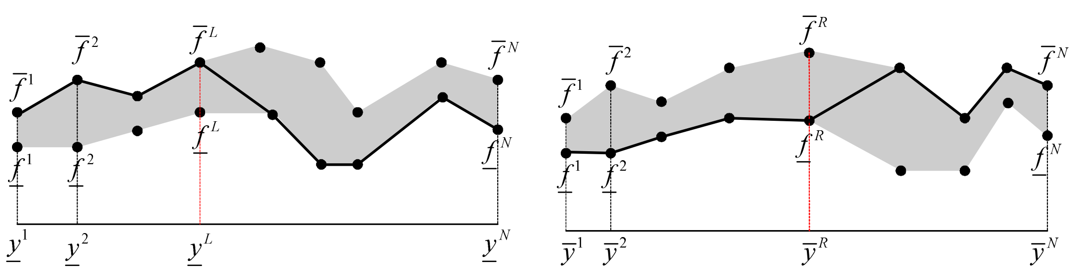
and compute

$$y' = \frac{\sum_{n=1}^{N} \overline{y}^n f^n}{\sum_{n=1}^{N} f^n} \tag{16}$$

e) Check if $y' = y$. If yes, stop and set $y_r = y$ and $R = k$. If no, go to Step 6.

f) Set $y = y'$ and go to Step 3.

The main idea of the KM algorithm is to find the switch points for y_l and y_r. Take y_l for example. y_l is the minimum of $Y_{cos}(\mathbf{x}')$. Since $\underline{y}^n$ increases from the left to the right along the horizontal axis of Fig. 3(a), we should choose a large weight (upper membership grade) for $\underline{y}^n$ on the left and a small weight (lower membership grade) for $\underline{y}^n$ on the right. The KM algorithm finds the switch point L. For $n \leqslant L$, the upper membership grades are used to calculate y_l; for $n > L$, the lower membership grades are used. This will ensure y_l be the minimum.



(a) Computing y_l: switch from the upper firing level to the lower firing level.

(b) Computing y_r: switch from the lower firing level to the upper firing level.

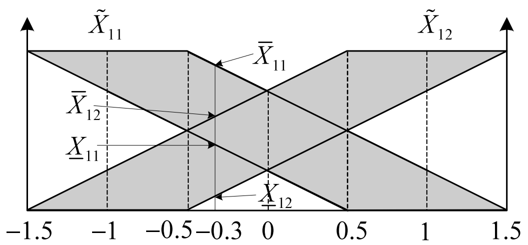
Fig. 3. Illustration of the switch points in computing y_l and y_r. The switch points can be found by the five algorithms introduced in [57].

4) Compute the defuzzified output as:

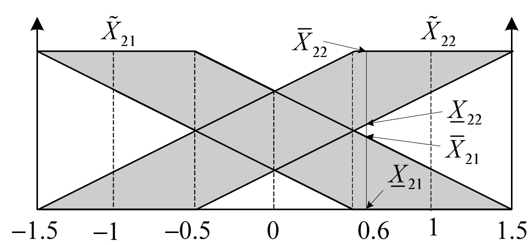
$$y = \frac{y_l + y_r}{2}. \tag{17}$$

A. Example of an IT2 FLS

In this section, the mathematical operations in an IT2 FLS are illustrated using an example. Consider an IT2 FLS that has two inputs (x_1 and x_2) and one output (y). Each input domain consists of two IT2 FSs, shown as the shaded areas in Fig. 4.



(a) Input MFs of x_1.



(b) Input MFs of x_2.

Fig. 4. MFs of the IT2 FLS.

The rulebase has the following four rules:

$$
\begin{aligned}
R^1: &\quad \text{IF } x_1 \text{ is } \widetilde{X}_{11} \text{ and } x_2 \text{ is } \widetilde{X}_{21}, \text{ THEN } y \text{ is } Y^1. \\
R^2: &\quad \text{IF } x_1 \text{ is } \widetilde{X}_{11} \text{ and } x_2 \text{ is } \widetilde{X}_{22}, \text{ THEN } y \text{ is } Y^2. \\
R^3: &\quad \text{IF } x_1 \text{ is } \widetilde{X}_{12} \text{ and } x_2 \text{ is } \widetilde{X}_{21}, \text{ THEN } y \text{ is } Y^3. \\
R^4: &\quad \text{IF } x_1 \text{ is } \widetilde{X}_{12} \text{ and } x_2 \text{ is } \widetilde{X}_{22}, \text{ THEN } y \text{ is } Y^4.
\end{aligned}
$$

The complete rulebase and the corresponding consequents are given in Table I.

TABLE I
RULEBASE AND CONSEQUENTS OF THE IT2 FLS.

x_1 \ x_2	$\widetilde{X}_{21}$	$\widetilde{X}_{22}$
$\widetilde{X}_{11}$	$Y^1 = [\underline{y}^1, \overline{y}^1] = [-1, -0.9]$	$Y^2 = [\underline{y}^2, \overline{y}^2] = [-0.6, -0.4]$
$\widetilde{X}_{12}$	$Y^3 = [\underline{y}^3, \overline{y}^3] = [0.4, 0.6]$	$Y^4 = [\underline{y}^4, \overline{y}^4] = [0.9, 1]$

Consider an input vector $\mathbf{x}' = (x_1', x_2') = (-0.3, 0.6)$. The firing intervals of the four IT2 FSs are:

$$[\mu_{\underline{X}_{11}}(x_1'), \mu_{\overline{X}_{11}}(x_1')] = [0.4,\ 0.9] \tag{18}$$
$$[\mu_{\underline{X}_{12}}(x_1'), \mu_{\overline{X}_{12}}(x_1')] = [0.1,\ 0.6] \tag{19}$$
$$[\mu_{\underline{X}_{21}}(x_2'), \mu_{\overline{X}_{21}}(x_2')] = [0,\ 0.45] \tag{20}$$
$$[\mu_{\underline{X}_{22}}(x_2'), \mu_{\overline{X}_{22}}(x_2')] = [0.55,\ 1] \tag{21}$$

The firing intervals of the four rules are:

Rule No.:	Firing Interval	$\to$	Consequent
R^1 :	$[\underline{f}^1, \overline{f}^1] = [\mu_{\underline{X}_{11}}(x_1) \cdot \mu_{\underline{X}_{21}}(x_2'), \mu_{\overline{X}_{11}}(x_1') \cdot \mu_{\overline{X}_{21}}(x_2')]$	$\to$	$[\underline{y}^1, \overline{y}^1] = [-1, -0.9]$
	$= [0.4 \times 0, 0.9 \times 0.45] = [0,\ 0.405]$		
R^2 :	$[\underline{f}^2, \overline{f}^2] = [\mu_{\underline{X}_{11}}(x_1') \cdot \mu_{\underline{X}_{22}}(x_2'), \mu_{\overline{X}_{11}}(x_1') \cdot \mu_{\overline{X}_{22}}(x_2')]$	$\to$	$[\underline{y}^2, \overline{y}^2] = [-0.6, -0.4]$
	$= [0.4 \times 0.55, 0.9 \times 1] = [0.22,\ 0.9]$		
R^3 :	$[\underline{f}^3, \overline{f}^3] = [\mu_{\underline{X}_{12}}(x_1') \cdot \mu_{\underline{X}_{21}}(x_2'), \mu_{\overline{X}_{12}}(x_1') \cdot \mu_{\overline{X}_{21}}(x_2')]$	$\to$	$[\underline{y}^3, \overline{y}^3] = [0.4, 0.6]$
	$= [0.1 \times 0, 0.6 \times 0.45] = [0,\ 0.27]$		
R^4 :	$[\underline{f}^4, \overline{f}^4] = [\mu_{\underline{X}_{12}}(x_1') \cdot \mu_{\underline{X}_{22}}(x_2'), \mu_{\overline{X}_{12}}(x_1') \cdot \mu_{\overline{X}_{22}}(x_2')]$	$\to$	$[\underline{y}^4, \overline{y}^4] = [0.9, 1]$
	$= [0.1 \times 0.55, 0.6 \times 1] = [0.055,\ 0.6]$		

From the KM algorithms, we find that $L = 1$ and $R = 3$. So,

$$
\begin{aligned}
y_l &= \frac{\overline{f}^1 \underline{y}^1 + \underline{f}^2 \underline{y}^2 + \underline{f}^3 \underline{y}^3 + \underline{f}^4 \underline{y}^4}{\overline{f}^1 + \underline{f}^2 + \underline{f}^3 + \underline{f}^4} \\
&= \frac{0.405 \times (-1) + 0.22 \times (-0.6) + 0 \times 0.4 + 0.055 \times 0.9}{0.405 + 0.22 + 0 + 0.055} \\
&= -0.7169 \\
y_r &= \frac{\underline{f}^1 \overline{y}^1 + \underline{f}^2 \overline{y}^2 + \underline{f}^3 \overline{y}^3 + \overline{f}^4 \overline{y}^4}{\underline{f}^1 + \underline{f}^2 + \underline{f}^3 + \overline{f}^4} \\
&= \frac{0 \times (-0.9) + 0.22 \times (-0.4) + 0 \times 0.6 + 0.6 \times 1}{0 + 0.22 + 0 + 0.6} \\
&= 0.6244
\end{aligned}
$$

Finally, the crisp output of the IT2 FLS, y, is:

$$
y = \frac{y_l + y_r}{2} = \frac{-0.7169 + 0.6244}{2} = -0.0463.
$$

B. Matlab Implementation

A Matlab function, *IT2FLS.m*, is provided for computing the output of an IT2 FLS given its rulebase and inputs. It uses the most efficient algorithm in [57] to compute y_l and y_r. For details, please refer to the explanations in *IT2FLS.m*. The above example is implemented in *example.m*. Note that each IT2 FS is represented by a 9-point vector $(p_1, ..., p_9)$ as shown in Fig. 5. So, the IT2 FS $\tilde{X}_{11}$ in Fig. 4(a) is represented as $(-1.5, -1.5, -.5, 1.5, -1.5, -1.5, -1.5, .5, 1)$, and $\tilde{X}_{12}$ is represented as $(-1.5, .5, 1.5, 1.5, -.5, 1.5, 1.5, 1.5, 1)$.

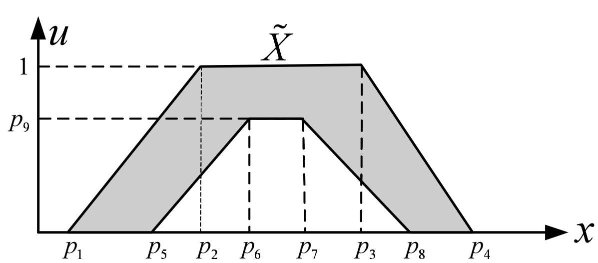


Fig. 5. The 9-point representation of an IT2 FS.

C. Approaches for Reducing the Computational Cost of IT2 FLSs

Many reported results have shown that IT2 FLSs are better able to handle uncertainties than their T1 counterparts [10], [21], [34], [62], [63]; however, the high computational cost of the iterative KM algorithms in type-reduction may hinder them from certain real-time applications. There have been many different approaches for reducing the computational cost of IT2 FLSs. [52] presents a comprehensive overview and comparison of them. These approaches can be grouped into three categories:

1) *Enhancements to the KM type-reduction algorithms* [15], [33], [55], [57], [68], which improve directly over the original KM algorithms to speed them up. [57] gives an overview and comparison of four such enhancements. It shows that the enhanced iterative algorithm with stop condition (EIASC), proposed in [57], is the fastest. It also gives the Matlab implementation of the EIASC.
2) *Alternative type-reduction algorithms* [4], [12], [14], [18], [19], [38], [42], [60], [65]. Unlike the iterative KM algorithms, these alternative type-reduction algorithms have closed-form representations. They are usually fast approximations of the KM algorithms. [54] gives an overview and comparison of nine alternative type-reduction algorithms.
3) *Simplified IT2 FLCs* [58], [63], in which the architecture of an IT2 FLC is simplified by using only a small number of IT2 FSs for the most critical input regions and T1 FSs for the rest.

Note that Category 1 and Category 2 are mutually exclusive, i.e., a method in Category 1 cannot be combined with a method in Category 2 for further computational cost savings; however, the method in Category 3 can be combined with a method in Category 1 or Category 2 for faster speed.

Because these is no comprehensive comparison on the performances of the KM algorithms based type-reduction approaches and the alternative type-reduction approaches (this is an interesting open problem), our recommendation [52] is to start from the full IT2 FLC using the EIASC algorithms [57], because most studies so far use the KM algorithms based type-reducer. If the rulebase is large, then it may also be worthwhile to use the simplified structure to further save some computational cost [58], [63]. If even more computational cost saving is desired, then alternative type-reduction algorithms like the Wu-Tan [60] or Nie-Tan [38] or Greenfield-Chiclana-Coupland-John [19] method may be considered because they are the only alternative type-reduction algorithms that are consistently faster than the EIASC algorithms.

D. Fundamental Differences between IT2 and T1 FLCs

A challenging question is: *What are the fundamental differences between IT2 and T1 FLSs?* Once the fundamental differences are clear, we can better understand the advantages of IT2 FLSs and hence better make use of them. In the literature there has been considerable effort on answering this challenging and fundamental question. Some important arguments are [53]:

1) *An IT2 FS can better model intra-personal*[4] *and inter-personal*[5] *uncertainties*, which are intrinsic to natural language, because the membership grade of an IT2 FS is an interval instead of a crisp number in a T1 FS. Mendel [35] also showed that IT2 FS is a scientifically correct model for modeling linguistic uncertainties, whereas T1 FS is not.

[4]According to Mendel [35], intra-personal uncertainty describes "*the uncertainty a person has about the word.*" It is also explicitly pointed out by psychologists Wallsten and Budescu [46] as "*except in very special cases, all representations are vague to some degree in the minds of the originators and in the minds of the receivers,*" and they suggest to model it by a T1 FS.

[5]According to Mendel [35], inter-personal uncertainty describes "*the uncertainty that a group of people have about the word,*" i.e., "*words mean different things to different people.*" It is also explicitly pointed out by psychologists Wallsten and Budescu [46] as "*different individuals use diverse expressions to describe identical situations and understand the same phrases differently when hearing or reading them.*"

2) *Using IT2 FSs to represent the FLS inputs and outputs will result in the reduction of the rulebase when compared to using T1 FSs* [21], [34], as the ability of the FOU to represent more uncertainties enables one to cover the input/output domains with fewer FSs. This makes it easier to construct the rulebase using expert knowledge and also increases robustness [59], [62], [63].
3) *An IT2 fuzzy logic controller (FLC) can give a smoother control surface than its T1 counterpart, especially in the region around the steady state* (i.e., when both the error and the change of error approach 0) [24], [59], [62], [63]. Wu and Tan [64] showed that when the baseline T1 FLC implements a linear PI control law and the IT2 FSs of an IT2 FLC are obtained from symmetrical perturbations of the T1 FSs, the resulting IT2 FLC implements a variable gain PI controller around the steady state. These gains are smaller than the PI gains of the baseline T1 FLC, which result in a smoother control surface around the steady state. The PI gains of the IT2 FLC also change with the inputs, which cannot be achieved by the baseline T1 FLC.
4) *IT2 FLCs are more adaptive and they can realize more complex input-output relationships which cannot be achieved by T1 FLCs.* Karnik and Mendel [27] pointed out that an IT2 fuzzy logic system can be thought of as a collection of many different embedded T1 fuzzy logic systems. Wu and Tan [61] proposed a systematic method to identify the *equivalent generalized T1 FSs* that can be used to replace the FOU. They showed that the equivalent generalized T1 FSs are significantly different from traditional T1 FSs, and there are different equivalent generalized T1 FSs for different inputs. Du and Ying [14], and Nie and Tan [39], also showed that a symmetrical IT2 fuzzy-PI (or the corresponding PD) controller, obtained from a baseline T1 PI FLC, partitions the input domain into many small regions, and in each region it is equivalent to a nonlinear PI controller with variable gains. The control law of the IT2 FLC in each small region is much more complex than that of the baseline T1 FLC, and hence it can realize more complex input-output relationship that cannot be achieved by a T1 FLC using the same rulebase.
5) *IT2 FLCs have a novelty that does not exist in traditional T1 FLCs.* Wu [50] showed that in an IT2 FLC *different* membership grades from the same IT2 FS can be used in different rules, whereas for traditional T1 FLC the *same* membership grade from the same T1 FS is always used in different rules. This again implies that an IT2 FLC is more complex than a T1 FLC and it cannot be implemented by a T1 FLC using the same rulebase.

More recently, Wu [53] explains two fundamental differences between IT2 and T1 FLCs: 1) *Adaptiveness*, meaning that the embedded T1 fuzzy sets used to compute the bounds of the type-reduced interval change as input changes; and, 2) *Novelty*, meaning that the upper and lower membership functions of the same IT2 fuzzy set may be used simultaneously in computing each bound of the type-reduced interval. T1 FLCs do not have these properties; thus, a T1 FLC cannot implement the complex control surface of an IT2 FLC given the same rulebase. Wu [53] also presents several methods for visualizing and analyzing the effects of these two fundamental differences, including the control surface, the P-map [51], the equivalent generalized T1 fuzzy sets [61], and the equivalent PI gains [64]. Finally, Wu [53] also examines five alternative type-reducers for IT2 FLCs and explain why they do not capture the fundamentals of IT2 FLCs.

IV. Awards and Further Readings

The following is a list of international awards given to works on type-2 fuzzy logic (there may be others):

1) Best Paper Award, IEEE Trans. on Fuzzy Systems, 1999 (Karnik, Mendel and Liang) [27].
2) Best Paper Award, IEEE Trans. on Fuzzy Systems, 2004 (Hagras) [20].
3) Best Student Paper Award, IEEE Int'l. Conf. on Fuzzy Systems, 2005 (Wu and Tan) [61].
4) First Prize, IEEE Region 10 Postgraduate Student Paper Contest, 2005 (Zeng) [71].

5) Best Student Paper Award, IEEE Int'l. Conf. on Fuzzy Systems, 2006 (Lynch, Hagras and Callaghan) [32].
6) Best Poster Presentation Award, IEEE Int'l. Conf. on Fuzzy Systems, 2006 (Tan and Kamal) [41].
7) Best Paper Award, IEEE Trans. on Fuzzy Systems, 2007 (Coupland and John) [12].
8) Best Student Paper Award, IEEE-NAFIPS (North American Fuzzy Information Processing Society) Conference, 2007 (Celikyilmaz and Turksen) [11].
9) Fuzzy Systems Pioneer Award, IEEE Computational Intelligence Society, 2008 (Mendel).
10) First Prize, IEEE Region 9 Undergraduate Student Paper Contest, 2008 (Bulla and Melgarejo) [7].
11) First Prize, IEEE Region 10 Undergraduate Student Paper Contest, 2008 (Li) [29].
12) Best Student Paper Award, NAFIPS (North American Fuzzy Information Processing Society) Conference, 2010 (Biglarbegian) [6].
13) IEEE Computational Intelligence Society Outstanding PhD Dissertation Award, 2012 (Wu) [49].

You are also encouraged to refer to the following publications for more information on (interval) type-2 fuzzy sets and systems:

- Jerry M. Mendel, *Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions*, Upper Saddle River, NJ: Prentice-Hall, 2001.
- Oscar Castillo and Patricia Melin, *Type-2 Fuzzy Logic: Theory and Applications*, Springer-Verlag, Berlin, 2008.
- Nilesh N. Karnik, Jerry M. Mendel, and Qilian Liang, "Type-2 fuzzy logic systems," *IEEE Trans. on Fuzzy Systems*, vol. 7, No. 6, pp. 643-658, 1999.
- Qilian Liang and Jerry M. Mendel, "Interval type-2 fuzzy logic systems: theory and design," *IEEE Trans. on Fuzzy Systems*, vol. 8, No. 5, pp. 535-550, 2000.
- Jerry M. Mendel and Robert I. John, "Type-2 fuzzy sets made simple," *IEEE Trans. on Fuzzy Systems*, vol. 10, No. 2, pp. 117-127, 2002.
- Jerry M. Mendel, Robert I. John and Feilong Liu, "Interval type-2 fuzzy logic systems made simple," *IEEE Trans. on Fuzzy Systems*, vol. 14, No. 6, pp. 808-821, 2006.
- Hani Hagras,"Type-2 FLCs: A new generation of fuzzy controllers," *IEEE Computational Intelligence Magazine*, vol. 2, pp. 30-43, 2007.
- Jerry M. Mendel,"Type-2 fuzzy sets and systems: an overview," *IEEE Computational Intelligence Magazine*, vol. 2, pp. 20-29, 2007.
- Jerry M. Mendel,"Advances in type-2 fuzzy sets and systems," *Information Sciences*, Vol. 177, pp. 84-110, 2007.
- Jerry M. Mendel, "Type-2 fuzzy sets and systems: How to learn about them," *IEEE SMC eNewsletter*, Issue 27, June 2009.
- Janet Aisbett, John T. Rickard, and David Morgenthaler, "Type-2 fuzzy sets as functions on spaces," *IEEE Trans. on Fuzzy Systems*, vol. 18, no. 4, pp. 841-844, 2010.

Acknowledgement

The author would like to thank Professor Jerry M. Mendel for his support and valuable suggestions.

Appendix A
Some Formal Definitions for T1 and IT2 FSs

Definition 1: A *type-1 fuzzy set* X is comprised of a *domain* D_X of real numbers (also called the *universe of discourse* of X) together with a *membership function* (MF) $\mu_X :\ D_X \to [0,1]$, i.e.,

$$X = \int_{D_X} \mu_X(x)/x \tag{22}$$

Here $\int$ denotes the collection of all points $x \in D_X$ with associated *membership grade* $\mu_X(x)$. ■

Definition 2: [34], [36] An IT2 FS $\tilde{X}$ is characterized by its MF $\mu_{\tilde{X}}(x,u)$, i.e.,

$$\begin{aligned}\widetilde{X} &= \int_{x\in D_{\tilde{X}}} \int_{u\in J_x \subseteq [0,1]} \mu_{\tilde{X}}(x,u)/(x,u) \\ &= \int_{x\in D_{\tilde{X}}} \int_{u\in J_x \subseteq [0,1]} 1/(x,u) \\ &= \int_{x\in D_{\tilde{X}}} \left[\int_{u\in J_x \subseteq [0,1]} 1/u \right] \Bigg/ x \end{aligned} \tag{23}$$

where x, called the *primary variable*, has domain $D_{\tilde{X}}$; $u \in [0,1]$, called the *secondary variable*, has domain $J_x \subseteq [0,1]$ at each $x \in D_{\tilde{X}}$; J_x is also called the *support of the secondary MF*, and is defined below in (25); and, the amplitude of $\mu_{\tilde{X}}(x,u)$, called a *secondary grade* of $\tilde{X}$, equals 1 for $\forall x \in D_{\tilde{X}}$ and $\forall u \in J_x \subseteq [0,1]$. ■

For *general type-2 FSs* [34] $\mu_{\tilde{X}}(x,u)$ can be any number in $[0,1]$, and it varies as x and/or u vary. An example of an IT2 FS is shown in Fig. 6.

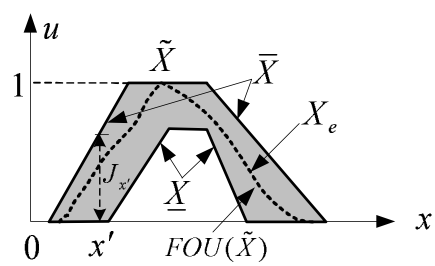


Fig. 6. An interval type-2 fuzzy set.

Definition 3: Uncertainty about $\tilde{X}$ is conveyed by the union of all its primary memberships, which is called the *footprint of uncertainty* (FOU) of $\tilde{X}$ (see Fig. 6), i.e.,

$$FOU(\tilde{X}) = \bigcup_{\forall x \in D_{\tilde{X}}} J_x = \{(x,u) : u \in J_x \subseteq [0,1]\}. \quad ■ \tag{24}$$

The size of an FOU is directly related to the uncertainty that is conveyed by an IT2 FS. So, an FOU with more area is more uncertain than one with less area.

Definition 4: The *upper membership function* (UMF) and *lower membership function* (LMF) of $\tilde{X}$ are two T1 MFs $\overline{X}$ and $\underline{X}$ that bound the FOU (see Fig. 6). ■

Note that the primary membership J_x is an *interval*, i.e.,

$$J_x = \left[\mu_{\underline{X}}(x), \mu_{\overline{X}}(x)\right] \tag{25}$$

Using (25), $FOU(\tilde{X})$ can also be expressed as

$$FOU(\tilde{X}) = \bigcup_{\forall x \in D_{\tilde{X}}} \left[\mu_{\underline{X}}(x), \mu_{\overline{X}}(x)\right] \tag{26}$$

References

[1] K. Atanassov, *Intuitionistic Fuzzy Sets: Theory and Applications*. Heidelberg/New York: Physica-Verlag,, 1999.

[2] K. Atanassov and G. Gargov, "Interval valued intuitionistic fuzzy sets," *Fuzzy Sets and Systems*, vol. 31, pp. 343–349, 1989.

[3] K. Atanassov, "Intuitionistic fuzzy sets," *Fuzzy Sets and Systems*, vol. 20, pp. 87–97, 1986.

[4] M. Begian, W. Melek, and J. Mendel, "Stability analysis of type-2 fuzzy systems," in *Proc. IEEE Int'l Conf. on Fuzzy Systems*, Hong Kong, June 2008, pp. 947–953.

[5] J. Bezdek, "Fuzzy models–what are they, and why?" *IEEE Trans. on Fuzzy Systems*, vol. 1, no. 1, pp. 1–5, 1993.

[6] M. Biglarbegian, W. Melek, and J. Mendel, "Robustness of interval type-2 fuzzy logic systems," in *Proc. NAFIPS*, Toronto, Canada, July 2010.

[7] J. Bulla and M. Melgarejo, "Implementing a simple microcontroller based interval type-2 fuzzy processor," in *IEEE Region 9 Undergraduate Student Paper Contest*, 2008.

[8] P. Burillo and H. Bustince, "Entropy on intuitionistic fuzzy sets and on interval-valued fuzzy sets," *Fuzzy Sets and Systems*, vol. 78, pp. 305–316, 1996.

[9] H. Bustince, "Indicator of inclusion grade for interval-valued fuzzy sets. Application to approximate reasoning based on interval-valued fuzzy sets," *International Journal of Approximate Reasoning*, vol. 23, no. 3, pp. 137–209, 2000.

[10] O. Castillo and P. Melin, *Type-2 Fuzzy Logic Theory and Applications*. Berlin: Springer-Verlag, 2008.

[11] A. Celikyilmaz and I. Turksen, "Enhanced type-2 fuzzy system models with improved fuzzy functions," in *Proc. NAFIPS*, San Diego, July 2007, pp. 140–145.

[12] S. Coupland and R. I. John, "Geometric type-1 and type-2 fuzzy logic systems," *IEEE Trans. on Fuzzy Systems*, vol. 15, no. 1, pp. 3–15, 2007.

[13] G. Deschrijver and E. E. Kerre, "On the relationship between some extensions of fuzzy set theory," *Fuzzy Sets and Systems*, vol. 133, no. 2, pp. 227–235, 2003.

[14] X. Du and H. Ying, "Derivation and analysis of the analytical structures of the interval type-2 fuzzy-PI and PD controllers," *IEEE Trans. on Fuzzy Systems*, vol. 18, no. 4, pp. 802–814, 2010.

[15] K. Duran, H. Bernal, and M. Melgarejo, "Improved iterative algorithm for computing the generalized centroid of an interval type-2 fuzzy set," in *Proc. NAFIPS*, New York, May 2008, pp. 1–5.

[16] J. Goguen, "*L*-fuzzy sets," *Journal of Mathematical Analysis and Applications*, vol. 18, pp. 145–174, 1967.

[17] M. B. Gorzalczany, "A method of inference in approximate reasoning based on interval-valued fuzzy sets," *Fuzzy Sets and Systems*, vol. 21, pp. 1–17, 1987.

[18] M. Gorzalczany, "Decision making in signal transmission problems with interval-valued fuzzy sets," *Fuzzy Sets and Systems*, vol. 23, pp. 191–203, 1987.

[19] S. Greenfield, F. Chiclana, S. Coupland, and R. John, "The collapsing method of defuzzification for discretised interval type-2 fuzzy sets," *Information Sciences*, vol. 179, no. 13, pp. 2055–2069, 2008.

[20] H. Hagras, "A hierarchical type-2 fuzzy logic control architecture for autonomous mobile robots," *IEEE Trans. on Fuzzy Systems*, vol. 12, pp. 524–539, 2004.

[21] H. Hagras, "Type-2 FLCs: A new generation of fuzzy controllers," *IEEE Computational Intelligence Magazine*, vol. 2, no. 1, pp. 30–43, 2007.

[22] K. Hirota and W. Pedrycz, "Fuzzy computing for data mining," *Proc. IEEE*, vol. 87, no. 9, pp. 1575–1600, 1999.

[23] S. Horikawa, T. Furahashi, and Y. Uchikawa, "On fuzzy modeling using fuzzy neural networks with back-propagation algorithm," *IEEE Trans. on Neural Networks*, vol. 3, pp. 801–806, 1992.

[24] E. A. Jammeh, M. Fleury, C. Wagner, H. Hagras, and M. Ghanbari, "Interval type-2 fuzzy logic congestion control for video streaming across IP networks," *IEEE Trans. on Fuzzy Systems*, vol. 17, no. 5, pp. 1123–1142, 2009.

[25] J. R. Jang, "Self-learning fuzzy controllers based on temporal back-propagation," *IEEE Trans. on Neural Networks*, vol. 3, pp. 714–723, 1992.

[26] N. N. Karnik and J. M. Mendel, "Centroid of a type-2 fuzzy set," *Information Sciences*, vol. 132, pp. 195–220, 2001.

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

[28] N. K. Kasabov and Q. Song, "DENFIS: Dynamic evolving neural-fuzzy inference system and its application for time-series prediction," *IEEE Trans. on Fuzzy Systems*, vol. 10, no. 2, pp. 144–154, 2002.

[29] B. Li, "Modeling of decision making process in power market with type-2 fuzzy logic and game theory," in *IEEE Region 10 Undergraduate Student Paper Contest*, 2008.

[30] S. S. Liao, T. H. Tang, and W.-Y. Liu, "Finding relevant sequences in time series containing crisp, interval, and fuzzy interval data," *IEEE Trans. on Systems, Man, and Cybernetics–B*, vol. 34, no. 5, pp. 2071–2079, 2004.

[31] F. Liu and J. M. Mendel, "Encoding words into interval type-2 fuzzy sets using an Interval Approach," *IEEE Trans. on Fuzzy Systems*, vol. 16, no. 6, pp. 1503–1521, 2008.

[32] C. Lynch, H. Hagras, and V. Callaghan, “Using uncertainty bounds in the design of an embedded real-time type-2 neuro-fuzzy speed controller for marine diesel engines,” in *Proc. IEEE Int’l Conf. on Fuzzy Systems*, Vancouver, Canada, July 2006, pp. 7217–7224.

[33] M. Melgarejo, “A fast recursive method to compute the generalized centroid of an interval type-2 fuzzy set,” in *Proc. NAFIPS*, San Diego, CA, June 2007, pp. 190–194.

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

[35] J. M. Mendel, “Computing with words: Zadeh, Turing, Popper and Occam,” *IEEE Computational Intelligence Magazine*, vol. 2, pp. 10–17, 2007.

[36] J. M. Mendel and R. I. John, “Type-2 fuzzy sets made simple,” *IEEE Trans. on Fuzzy Systems*, vol. 10, no. 2, pp. 117–127, 2002.

[37] J. M. Mendel and D. Wu, *Perceptual Computing: Aiding People in Making Subjective Judgments*. Hoboken, NJ: Wiley-IEEE Press, 2010.

[38] M. Nie and W. W. Tan, “Towards an efficient type-reduction method for interval type-2 fuzzy logic systems,” in *Proc. IEEE Int’l Conf. on Fuzzy Systems*, Hong Kong, June 2008, pp. 1425–1432.

[39] M. Nie and W. W. Tan, “Derivation of the analytical structure of symmetrical IT2 fuzzy PD and PI controllers,” in *Proc. IEEE Int’l. Conf. on Fuzzy Systems*, Barcelona, Spain, July 2010, pp. 1–8.

[40] W. Pedrycz, “Fuzzy set technology in knowledge discovery,” *Fuzzy Sets and Systems*, vol. 98, pp. 279–290, 1998.

[41] W. W. Tan and D. H. Kamal, “On-line learning rules for type-2 fuzzy controller,” in *Proc. IEEE Int’l Conf. on Fuzzy Systems*, Vancouver, Canada, July 2006, pp. 2530–2537.

[42] C. W. Tao, J. S. Taur, C.-W. Chang, and Y.-H. Chang, “Simplified type-2 fuzzy sliding controller for wing rock system,” *Fuzzy sets and systems*, 2011, in press.

[43] I. B. Türkşen, “Interval valued fuzzy sets and fuzzy connectives,” *Journal of of Interval Computations*, vol. 4, pp. 125–142, 1993.

[44] I. B. Türkşen, “Interval-valued fuzzy sets based on normal forms,” *Fuzzy Sets and Systems*, vol. 20, pp. 191–210, 1986.

[45] M. Versaci and F. C. Morabito, “Fuzzy time series approach for disruption prediction in tokamak reactors,” *IEEE Trans. on Magnetics*, vol. 39, no. 3, pp. 1503–1506, 2003.

[46] T. S. Wallsten and D. V. Budescu, “A review of human linguistic probability processing: General principles and empirical evidence,” *The Knowledge Engineering Review*, vol. 10, no. 1, pp. 43–62, 1995.

[47] L.-X. Wang and J. M. Mendel, “Fuzzy basis functions, universal approximation, and orthogonal least-squares learning,” *IEEE Trans. on Neural Networks*, vol. 3, pp. 807–813, 1992.

[48] L.-X. Wang, *A Course in Fuzzy Systems and Control*. Upper Saddle River, NJ: Prentice Hall, 1997.

[49] D. Wu, “Intelligent systems for decision support,” Ph.D. dissertation, University of Southern California, Los Angeles, CA, May 2009.

[50] D. Wu, “An interval type-2 fuzzy logic system cannot be implemented by traditional type-1 fuzzy logic systems,” in *Proc. World Conference on Soft Computing*, San Francisco, CA, May 2011.

[51] D. Wu, “P-map: An intuitive plot to visualize, understand, and compare variable-gain PI controllers,” in *Proc. Int’l. Conf. on Autonomous and Intelligent Systems*, Burnaby, BC, Canada, June 2011.

[52] D. Wu, “Approaches for reducing the computational cost of interval type-2 fuzzy logic controllers: Overview and comparison,” *IEEE Trans. on Fuzzy Systems*, 2012, submitted.

[53] D. Wu, “On the fundamental differences between interval type-2 and type-1 fuzzy logic controllers,” *IEEE Trans. on Fuzzy Systems*, 2012, in press.

[54] D. Wu, “An overview of alternative type-reduction approaches for reducing the computational cost of interval type-2 fuzzy logic controllers,” in *Proc. IEEE World Congress on Computational Intelligence*, Brisbane, Australia, June 2012.

[55] D. Wu and J. M. Mendel, “Enhanced Karnik-Mendel Algorithms,” *IEEE Trans. on Fuzzy Systems*, vol. 17, no. 4, pp. 923–934, 2009.

[56] D. Wu, J. M. Mendel, and S. Coupland, “Enhanced Interval Approach for encoding words into interval type-2 fuzzy sets and its convergence analysis,” *IEEE Trans. on Fuzzy Systems*, 2012, in press.

[57] D. Wu and M. Nie, “Comparison and practical implementation of type-reduction algorithms for type-2 fuzzy sets and systems,” in *Proc. IEEE Int’l Conf. on Fuzzy Systems*, Taipei, Taiwan, June 2011.

[58] D. Wu and W. W. Tan, “A simplified architecture for type-2 FLSs and its application to nonlinear control,” in *Proc. IEEE Conf. on Cybernetics and Intelligent Systems*, Singapore, Dec. 2004, pp. 485–490.

[59] D. Wu and W. W. Tan, “A type-2 fuzzy logic controller for the liquid-level process,” in *Proc. IEEE Int’l Conf. on Fuzzy Systems*, vol. 2, Budapest, Hungary, July 2004, pp. 953–958.

[60] D. Wu and W. W. Tan, “Computationally efficient type-reduction strategies for a type-2 fuzzy logic controller,” in *Proc. IEEE Int’l Conf. on Fuzzy Systems*, Reno, NV, May 2005, pp. 353–358.

[61] D. Wu and W. W. Tan, “Type-2 FLS modeling capability analysis,” in *Proc. IEEE Int’l Conf. on Fuzzy Systems*, Reno, NV, May 2005, pp. 242–247.

[62] D. Wu and W. W. Tan, “Genetic learning and performance evaluation of type-2 fuzzy logic controllers,” *Engineering Applications of Artificial Intelligence*, vol. 19, no. 8, pp. 829–841, 2006.

[63] D. Wu and W. W. Tan, “A simplified type-2 fuzzy controller for real-time control,” *ISA Transactions*, vol. 15, no. 4, pp. 503–516, 2006.

[64] D. Wu and W. W. Tan, “Interval type-2 fuzzy PI controllers: Why they are more robust,” in *Proc. IEEE Int’l. Conf. on Granular Computing*, San Jose, CA, August 2010, pp. 802–807.

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

[66] R. Yager, “Database discovery using fuzzy sets,” *International Journal of Intelligent Systems*, vol. 11, pp. 691–712, 1996.

[67] R. Yager and D. Filev, *Essentials of Fuzzy Modeling and Control*. John Wiley & Son, 1994.

[68] C.-Y. Yeh, W.-H. Jeng, and S.-J. Lee, “An enhanced type-reduction algorithm for type-2 fuzzy sets,” *IEEE Trans. on Fuzzy Systems*, vol. 19, no. 2, pp. 227–240, 2011.

[69] L. A. Zadeh, “Fuzzy sets,” *Information and Control*, vol. 8, pp. 338–353, 1965.

[70] L. A. Zadeh, “The concept of a linguistic variable and its application to approximate reasoning-1,” *Information Sciences*, vol. 8, pp. 199–249, 1975.

[71] J. Zeng, “Type-2 fuzzy hidden Markov models and their application to phoneme classification,” in *IEEE Region 10 Postgraduate Student Paper Contest*, 2005.