Perceptual Reasoning for Perceptual Computing

Jerry M. Mendel, *Life Fellow, IEEE*, and Dongrui Wu, *Student Member, IEEE*

Abstract—In 1996, Zadeh proposed the paradigm of *computing with words* (CWW). A specific architecture for making subjective judgments using CWW was proposed by Mendel in 2001. It is called a *Perceptual Computer* (Per-C), and because words can mean different things to different people, it uses interval type-2 fuzzy set (IT2 FS) models for all words. The Per-C has three elements: the encoder, which transforms linguistic perceptions into IT2 FSs that activate a CWW engine; the decoder, which maps the output of a CWW engine back into a word; and the CWW engine. Although different kinds of CWW engines are possible, this paper only focuses on CWW engines that are rule-based and the computations that map its input IT2 FSs into its output IT2 FS. Five assumptions are made for a rule-based CWW engine, the most important of which is: The result of combining fired rules must lead to a footprint of uncertainty (FOU) that resembles the three kinds of FOU that have previously been shown to model words (interior, left-shoulder, and right-shoulder FOUs). Requiring this means that the output FOU from a rule-based CWW engine will look similar in shape to an FOU in a codebook (i.e., a vocabulary of words and their respective FOUs) for an application, so that the decoder can therefore sensibly establish the word most similar to the CWW engine output FOU. Because existing approximate reasoning methods do not satisfy this assumption, a new kind of rule-based CWW engine is proposed, one that is called *Perceptual Reasoning*, and is proved to always satisfy this assumption. Additionally, because all IT2 FSs in the rules as well as those that excite the rules are either an interior, left-shoulder, or right-shoulder FOU, it is possible to carry out the sup-min calculations that are required by the inference engine, and those calculations are also in this paper. The results in this paper let us implement a rule-based CWW engine for the Per-C.

Index Terms—Computing with words, footprint of uncertainty, interval type-2 fuzzy sets, perceptual computer, perceptual reasoning (PR), rule-based systems.

I. INTRODUCTION

ZADEH coined the phrase "*computing with words*" (CWW)[1] [41], [42]. According to him, CWW is "*a methodology in which the objects of computation are words and propositions drawn from a natural language.*" It is "*inspired by the remarkable human capability to perform a wide variety of physical and mental tasks without any measurements and any computations.*" Words in the CWW paradigm can be modeled by type-1 fuzzy sets (T1 FSs) or their extension, type-2 (T2) FSs.

CWW using T1 FSs has been studied by many researchers, e.g., [3], [8], [13], [24], [27], [29], [30], [39], [41], and [42];

[1]Different acronyms have been used for "computing with words," e.g., CW and CWW. We have chosen to use the latter, since its three letters coincide with the three words in "computing with words."
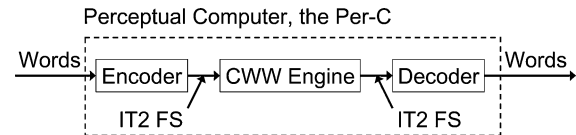

Fig. 1. Architecture of the Perceptual Computer.

however, as claimed in [15]–[18], "*Words mean different things to different people, and so are uncertain. We therefore need an FS model for a word that has the potential to capture its uncertainties, and an interval T2 FS (IT2 FS) should be used as a model of a word.*" Consequently, in this paper, IT2 FSs are used to model words.

A specific architecture is proposed in [15]–[18] for making (subjective) judgments by CWW. It is called a *Perceptual Computer*—Per-C for short—and is depicted in Fig. 1, and its use is called *Perceptual Computing*. Perceptions (i.e., granulated terms, words) activate the Per-C and are also output by the Per-C; so, it is possible for a human to interact with the Per-C just using a vocabulary of words.

In Fig. 1, the *encoder* transforms linguistic perceptions into IT2 FSs that activate a *CWW engine*. It contains an application's *codebook* that is a collection of words (the application's vocabulary) and their IT2 FS models. How to obtain IT2 FS models for words is explained in [10] and [11], and is not the subject of this paper, although some aspects of it are discussed next. The *decoder* maps the output of the CWW engine back into a word. Usually, a codebook is available, in which every word (the vocabulary) is modeled as an IT2 FS. The output of the CWW engine is mapped into a word (in that vocabulary) most similar to it. How to do this is explained in [34] and is also not the subject of this paper, although some aspects of this will also be explained next. The CWW engine, e.g. IF-THEN rules (e.g., [16]), the linguistic weighted average [31]–[33], linguistic summarizations [5], [22], [38], etc., maps IT2 FSs into IT2 FSs. *This paper focuses only on CWW engines that are rule-based and the computations that map the input IT2 FSs into the output IT2 FSs.*

In order to carry out those computations, one must first ask: "What kinds of IT2 FSs should be used to model antecedent, consequent and input words in a rule-based CWW engine?" An answer has been obtained by Liu and Mendel [10], [11] and is explained next. It is predicated on the belief that, in order to obtain a meaningful uncertainty model for a word, data about the word must be collected from a group of subjects. In their encoding method, called the *Interval Approach*, data intervals about a vocabulary of words are obtained from a group of subjects. Subjects are asked "On a scale of 0–10 where would you locate the end-points of an interval that you associate with the word___?" After some preprocessing, during which some intervals (e.g., outliers) are eliminated, the collection of remaining intervals is
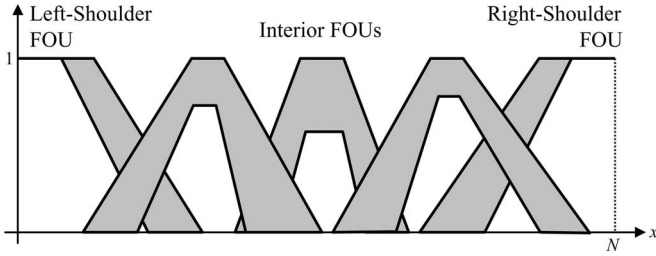
Fig. 2. FOUs for CWWs.

classified as either an interior, left-shoulder, or right-shoulder IT2 FS. Then, each of the data intervals is individually mapped into its respective T1 interior, left-shoulder, or right-shoulder MF, after which the union of all of these T1 MFs is taken, and the union is upper and lower bounded using *piecewise-linear* functions. The result is a *footprint of uncertainty* (FOU) for an IT2 FS[2], which is completely described by these lower and upper bounds, called the *lower membership function* (LMF) and the *upper membership function* (UMF), respectively. Regardless of the size of a survey, this IA method will lead to piecewise-linear shapes for the LMF and UMFs of words.

When this methodology was applied to real data [9] for a vocabulary of 32 words, the left-shoulder, right-shoulder, and interior FOUs (see Fig. 2) had the following general features.

1) *Left- (LS) and Right-Shoulder (RS) FOUs:* The legs of the LMF and UMF are not parallel.
2) *Interior FOUs:* The UMF is a trapezoid that usually is not symmetrical, and the LMF is a triangle that usually is not symmetrical[3].

It is such FOUs that will be used in the Per-C.

Recall that the general structure of a rule for p inputs $x_1 \in X_1, \ldots, x_p \in X_p$ and one output $y \in Y$, is:

$$R^i : \text{ IF } x_1 \text{ is } \tilde{F}_1^i \text{ and } \cdots \text{ and } x_p \text{ is } \tilde{F}_p^i, \quad \text{THEN } y \text{ is } \tilde{G}^i$$
$$i = 1, \ldots, M. \tag{1}$$

In this rule, the p antecedents and the consequent are modeled as IT2 FSs that are a subset of the words in a CWW codebook; hence, as just mentioned, they can only be IT2 FSs like the ones shown in Fig. 2.

Comments: 1) The codebook for a CWW application may be rather large, so that users who interface with the Per-C can operate in a user-friendly environment. Usually, only a small subset of the words in the codebook would be used to establish the M rules, especially when rules are extracted from experts. What is important is that the words used to characterize each of the p antecedents and the consequent lead to FOUs that cover the domain of each antecedent. In Fig. 2, e.g., five words do this. In our experience, five to seven words will cover an interval, e.g., 0–10.

2) A reviewer of this paper thought that (1) was an interval type-2 fuzzy logic system (IT2 FLS). It is *not*, because of the

[2]It is assumed that readers are familiar with IT2 FSs. If they are not, see, e.g., [19].
[3]In the extremely unlikely situation when all subjects give the same interval, the UMF can be a triangle.

way in which fired rules will be combined during perceptual reasoning (PR, as explained in Section II) and because type reduction and defuzzification, which are the two components of output processing for an IT2 FLS, are *not* used by us in perceptual computing. In perceptual computing, our goal is not to obtain a number at the output of the Per-C (which is what is obtained at the output of an IT2 FLS), but is instead to obtain a *word* that is most similar to a word in the codebook. ∎

How one should model the M rules, their inference mechanism, and the combining of multiple fired rules for a Per-C are questions that do not have unique answers; so, choices must be made. In this paper, the following choices (i.e., assumptions) are made:

Assumptions:

1) *The result of combining fired rules must lead to an FOU that resembles the three kinds of FOUs in a CWW codebook*: This is a very plausible requirement, since the decoder in the Per-C maps the CWW output FOU into a word in the codebook most similar to it. This is our most challenging assumption to achieve because it cannot be assumed *a priori* but must be demonstrated through analysis. We do this in Section III.

2) *Input IT2 MFs are separable*: By a *separable MF*, we mean one for which $\mu_{X_1, \ldots, X_p}(x_1, \ldots, x_p) = \mu_{X_1}(x_1) \wedge \ldots \wedge \mu_{X_p}(x_p)$. Because each word in the vocabulary has been modeled independently, and inputs to the Per-C are words, separable MFs seem reasonable.

3) *No uncertainties are included about connective words:* Although there exists a literature (e.g., [25], [26], [28], and [36]) for allowing the connective words *and* and *or* to incorporate uncertainties, except for [36], all results are for type-1 FS antecedents and consequents, and are very complicated, and [36]'s results are the most complicated. Our first approach to a CWW rule-based engine is to keep it as simple as possible, and to see if sensible results can be obtained. If they cannot be, then one possibility is to use more complicated models for connector words, but they must be in the context of IT2 FS models for words.

4) *Rules are activated by words that are modeled as either shoulder or interior IT2 FSs:* Rules will be activated by words that are in the codebook, and as we have explained, these words will be modeled as assumed earlier.

5) *Minimum t-norm is used for the* and *connective*: In a rule-based FLS product and minimum, t-norms are most popular. We have found that computing the sup-min composition in closed form is relatively straightforward for shoulder and interior FOUs (see Section IV), but computing the sup-product composition is very difficult. Because the Per-C is very different from the more popular function approximation application for an IT2 FLS, in which universal approximation dominates and product t-norm is most popular, there is no compelling reason to use product t-norm over the minimum t-norm. So, a pragmatic approach is taken in this paper, one that focuses on the minimum t-norm.

The rest of this paper is organized as follows: Section II describes a kind of reasoning that satisfies Assumption 1, and

which we call PR. It also provides computational algorithms for PR. Section III studies the properties of PR, and demonstrates that the aforesaid Assumption 1 is satisfied for it. Section IV calculates firing intervals for the small set of FOUs that can occur in a CWW situation. Section V provides some discussions including a family of applications for PR. Section VI provides conclusions and suggestions for future studies.

II. PR: ALGORITHMS

A. Introduction

There are many models for the fuzzy implication, under the rubric of *approximate reasoning*, e.g., [7] Table 11.1 lists 14. Each of these models has the property that it reduces to the truth table of material implication when fuzziness disappears, and to date none of these models has been examined using interval T2 FSs.

Following is a quote from [2] that we have found to be very illuminating:

Rational calculation is the view that the mind works by carrying out probabilistic, logical, or decision-theoretic operations. Rational calculation is explicitly avowed by relatively few theorists, though it has clear advocates with respect to logical inference. Mental logicians propose that much of cognition is a matter of carrying out logical calculations (e.g., [1], [4], [23]). *Rational description*, by contrast, is the view that behavior can be approximately described as conforming with the results that would be obtained by some rational calculation. This view does *not* assume (though it does not rule out) that the thought processes underlying behavior involve any rational calculation.

For the Per-C, logical reasoning is not implemented as prescribed by the truth table of material implication; instead, rational description is subscribed to.

Two widely used fuzzy reasoning models that fit the concept of rational description are Mamdani and TSK, because neither satisfies the complete truth table for material implication, and so are not rational calculation models. Both models have been examined using interval T2 FSs (e.g., [9], [16]); however, neither leads to a combined fired-rules output set that resembles the FOUs in our codebook (Fig. 2). Recall (e.g., see Fig. 6) that even for T1 FSs, each fired rule output fuzzy set for Mamdani implication that uses, e.g., the minimum t-norm looks like a clipped version of the consequent FS,[4] and such an FS does not resemble the consequent FS. For a TSK model, the concept of a fired output FS does not occur, because the rule consequent in a TSK rule is not an FS, but is a function of the inputs.

How fired rules are connected (combined) for a Mamdani model is open to interpretation. Zadeh connected rules [40] using the word *ELSE*, which is itself a bit vague. Some have interpreted the word *ELSE* as the *OR* connector, some have interpreted it as the *AND* connector, and not surprisingly, some have interpreted it as a blend of both the *AND* and *OR* connectors. Others prefer to perform the combining as a part of defuzzification. There is no measured evidence (data) to support any of

these rule-combining methods for a Mamdani model when the objective is to make subjective judgments.

Interestingly enough, fired rules are easily combined using the TSK model through a weighted average of rule consequent functions, where the weights are the rule firing strengths. The result, though, is not an FS; it is a point value for T1 FSs or an interval value for IT2 FSs. So, neither the Mamdani nor TSK models seem to be appropriate for the Per-C.

B. PR Described

A new fuzzy reasoning model is now proposed [21]—PR[5]—that not only fits the concept of rational description, but also satisfies Assumption 1, namely, that *the result of combining fired rules must lead to an FOU that resembles the three kinds of FOUs in a CWW codebook.*

PR consists of two steps.
1) A firing interval is computed for each rule, as would be done for both the IT2 FS Mamdani and TSK models.
2) The IT2 FS consequents of the fired rules are combined using a linguistic weighted average (LWA) [31]–[33], in which the weights are the firing intervals and the "signals" are the IT2 FS consequents.

Firing interval calculations are covered in Section IV. Next, the mechanism of PR is explained.

C. Combining the Fired Rules Using the LWA

In this section, it is assumed that, for a given $p \times 1$ vector of input IT2 FSs, $\tilde{\mathbf{X}}'$, firing intervals $F^i(\tilde{\mathbf{X}}')$ have been computed for all fired rules.[6] In practice, $n \leq M$ rules fire, and usually n is much smaller than M. Those firing intervals are denoted

$$F^i(\tilde{\mathbf{X}}') = \left[\underline{f}^i(\tilde{\mathbf{X}}'), \overline{f}^i(\tilde{\mathbf{X}}')\right] \equiv \left[\underline{f}^i, \overline{f}^i\right] \qquad (2)$$

where $\underline{f}^i$ and $\overline{f}^i$ are the lower and upper bounds for $F^i(\tilde{\mathbf{X}}')$, and often for notational simplicity, the explicit dependence of these bounds on $\tilde{\mathbf{X}}'$ is omitted. In PR, fired rules are combined using the following LWA that is denoted $\tilde{Y}_{\mathrm{PR}}$:

$$\tilde{Y}_{\mathrm{PR}} = \frac{\sum_{i=1}^{n} F^i(\tilde{\mathbf{X}}')\tilde{G}^i}{\sum_{i=1}^{n} F^i(\tilde{\mathbf{X}}')}. \qquad (3)$$

In (3), $F^i(\tilde{\mathbf{X}}')$ are intervals of nonnegative real numbers and $\tilde{G}^i$ are rule-consequent IT2 FSs. This LWA is a special case of the more general LWA in which both $\tilde{G}^i$ and $F^i(\tilde{\mathbf{X}}')$ are IT2 FSs.

Equation (3) is an *expressive* way to describe $\tilde{Y}_{\mathrm{PR}}$, meaning that $\tilde{Y}_{\mathrm{PR}}$ is not computed using multiplications, additions, or

---

[4]When it uses the product t-norm, it looks like a scaled version of the consequent FS.

[5]"Perceptual Reasoning" is a term that we have coined, because it is used by the *Perceptual* Computer, when the CWW Engine consists of IF-THEN rules.

[6]There is a very important notational difference between the firing intervals in this paper and those in the existing literature about IT2 FLSs. Here, because words are modeled as IT2 FSs and words excite all rules, the firing interval, $F^i(\tilde{\mathbf{X}}')$, is shown to depend upon the entire input IT2 FS $\tilde{\mathbf{X}}'$. In the existing IT2 FLS literature (e.g., [16]), an input to a rule is a vector of numbers $\mathbf{X}'$ that can be fuzzified as an IT2 FS, in which case T2 nonsingleton fuzzification is said to occur. In that literature, it is therefore common to see $F^i(\mathbf{X}')$ instead of $F^i(\tilde{\mathbf{X}}')$.
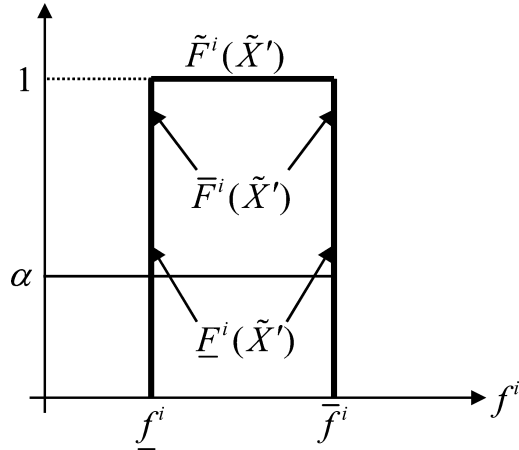
Fig. 3. $\tilde{F}^i(\tilde{\mathbf{X}}')$, the interpreted IT2 FS for firing interval $F^i(\tilde{\mathbf{X}}')$ of Rule-i.

divisions, as expressed by (3). Instead, the lower and upper MFs of $\tilde{Y}_{\mathrm{PR}}$, $\underline{Y}_{\mathrm{PR}}$, and $\overline{Y}_{\mathrm{PR}}$ are computed separately using α-cuts, as summarized in Section II-E.

In the rest of this section, we provide a brief and highly condensed explanation of how to compute $\tilde{Y}_{\mathrm{PR}}$.

D. Overview of Computing $\tilde{Y}_{\mathrm{PR}}$

In order to use the results in [31]–[33], $F^i(\tilde{\mathbf{X}}')$ is interpreted here as an IT2 FS whose MF is depicted in Fig. 3. Observe, in Fig. 3, each α-cut on $F^i(\tilde{\mathbf{X}}')$ is the *same* interval $[\underline{f}^i, \overline{f}^i]$, for $\forall \alpha \in [0, 1]$, and that $\underline{F}^i(\tilde{\mathbf{X}}') = \overline{F}^i(\tilde{\mathbf{X}}')$.

FOUs for $\tilde{G}^i$ are depicted in Fig. 4. Observe that for an interior FOU, the height of $\underline{G}^i$ is denoted $h_{\underline{G}^i}$, the α-cut on $\underline{G}^i$ is denoted[7] $[a_{ir}(\alpha), b_{il}(\alpha)]$, $\alpha \in [0, h_{\underline{G}^i}]$, and the α-cut on $\overline{G}^i$ is denoted $[a_{il}(\alpha), b_{ir}(\alpha)]$, $\alpha \in [0, 1]$.

An interior FOU for $\tilde{Y}_{\mathrm{PR}}$ is depicted in Fig. 5. The α-cut on $\overline{Y}_{\mathrm{PR}}$ is $[y_{Ll}(\alpha), y_{Rr}(\alpha)]$ and the α-cut on $\underline{Y}_{\mathrm{PR}}$ is $[y_{Lr}(\alpha), y_{Rl}(\alpha)]$, where, as shown in [31]–[33], the end points of these α-cuts are computed as solutions to the following four decoupled optimization problems[8]:

$$y_{Ll}(\alpha) = \min_{\forall f_i \in [\underline{f}^i, \overline{f}^i]} \frac{\sum_{i=1}^n a_{il}(\alpha) f_i}{\sum_{i=1}^n f_i}, \qquad \alpha \in [0, 1] \qquad (4)$$

$$y_{Rr}(\alpha) = \max_{\forall f_i \in [\underline{f}^i, \overline{f}^i]} \frac{\sum_{i=1}^n b_{ir}(\alpha) f_i}{\sum_{i=1}^n f_i}, \qquad \alpha \in [0, 1] \qquad (5)$$

$$y_{Lr}(\alpha) = \min_{\forall f_i \in [\underline{f}^i, \overline{f}^i]} \frac{\sum_{i=1}^n a_{ir}(\alpha) f_i}{\sum_{i=1}^n f_i}, \qquad \alpha \in [0, h_{\underline{Y}_{\mathrm{PR}}}] \qquad (6)$$

$$y_{Rl}(\alpha) = \max_{\forall f \in [\underline{f}^i, \overline{f}^i]} \frac{\sum_{i=1}^n b_{il}(\alpha) f_i}{\sum_{i=1}^n f_i}, \qquad \alpha \in [0, h_{\underline{Y}_{\mathrm{PR}}}] \qquad (7)$$

---

[7] In this notation, the first subscript is an index that runs from 1 to at most M, whereas the second subscript is a pneumonic for left or right.

[8] The LWA used in this paper is slightly different from the version proposed in [31] and [32] in that here $h_{\underline{Y}_{\mathrm{PR}}} = \min_i h_{\underline{G}^i}$, whereas in [31] and [32] $h_{\underline{Y}_{\mathrm{PR}}}$ may be larger than $\min_i h_{\underline{G}^i}$. A detailed explanation of this is given in [33]. We advise the reader to use the LWA in this paper because it handles the case when $h_{\underline{G}^i}$ are not all the same correctly.
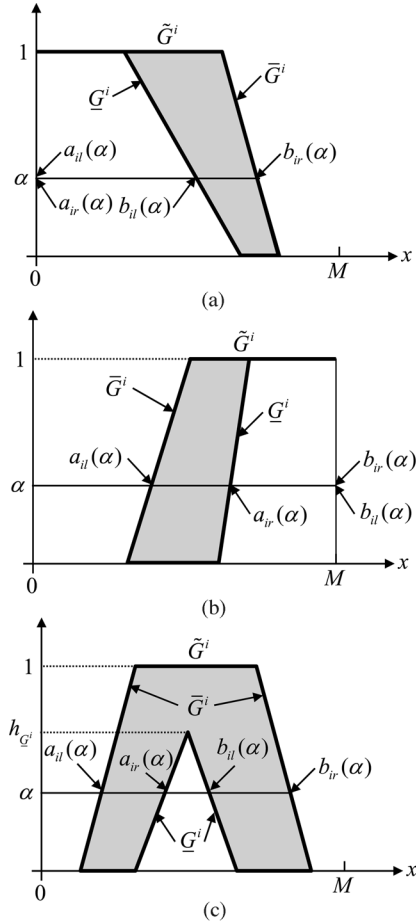

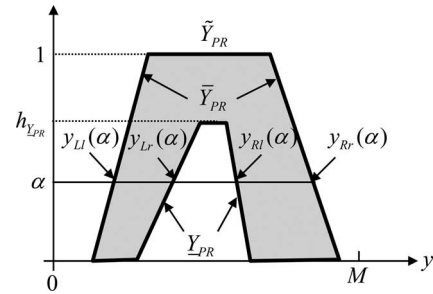Fig. 4. (a) LS, (b) RS, and (c) interior FOUs for consequent words.

Fig. 5. $\tilde{Y}_{\mathrm{PR}}$, the LWA for PR.

where

$$h_{\underline{Y}_{\mathrm{PR}}} = \min_i h_{\underline{G}^i}. \qquad (8)$$

Observe from (4) and (5) that $\overline{Y}_{\mathrm{PR}}$, characterized by $[y_{Ll}(\alpha), y_{Rr}(\alpha)]$, is completely determined by the UMFs $\overline{G}^i$, because they involve only $a_{il}(\alpha)$ and $b_{ir}(\alpha)$ (see Fig. 4), and from (6) and (7) that $\underline{Y}_{\mathrm{PR}}$, characterized by $[y_{Lr}(\alpha), y_{Rl}(\alpha)]$, is completely determined by the LMFs $\underline{G}^i$, because they only involve $a_{ir}(\alpha)$ and $b_{il}(\alpha)$. Observe also from (4) and (5) that
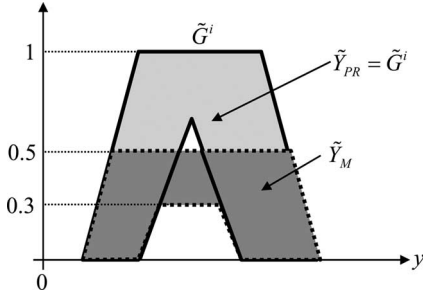
Fig. 6. Outputs of PR ($\tilde{Y}_{\mathrm{PR}}$, the solid curves) and a Mamdani inference mechanism ($\tilde{Y}_M$, the dotted curves) when only rule R^i fires with firing interval [0.3, 0.5].

$\bar{Y}_{\mathrm{PR}}$ is always normal, i.e., its $\alpha = 1$ α-cut can always be computed. This is different from many other approximate reasoning methods, e.g., the Mamdani-inference based method. For the latter, even if only one rule is fired, unless the firing interval is $[1, 1]$, the output is a clipped or scaled version of the original IT2 FS instead of a normal IT2 FS, as shown in Fig. 6. This may cause problems when the output is mapped to a word in the codebook.

E. Algorithms

In summary, knowing the firing intervals $[\underline{f}^i, \overline{f}^i]$, $i = 1, \ldots, n$, $\bar{Y}_{\mathrm{PR}}$ is computed in the following way:
1) Calculate $y_{L1}(\alpha_j)$ and $y_{Rr}(\alpha_j)$, $j = 1, \ldots, m$. To do this:
   a) Select appropriate m α-cuts for $\bar{Y}_{\mathrm{PR}}$ (e.g., divide [0, 1] into $m - 1$ intervals and set $\alpha_j = (j - 1)/(m - 1)$, $j = 1, \ldots, m$).
   b) Find the α_j α-cut on $\bar{G}^i (i = 1, \ldots, n)$; denote the end points of its interval as $[a_{il}(\alpha_j), b_{ir}(\alpha_j)]$, respectively.
   c) Use KM algorithms [6], [16] to find $y_{L1}(\alpha_j)$ in (4) and $y_{Rr}(\alpha_j)$ in (5).
   d) Repeat steps (b) and (c) for every α_j $(j = 1, \ldots, m)$.
2) Construct $\bar{Y}_{\mathrm{PR}}$ from the m α-cuts. To do this:
   a) Store the left coordinates $(y_{L1}(\alpha_j), \alpha_j)$, $j = 1, \ldots, m$.
   b) Store the right coordinates $(y_{Rr}(\alpha_j), \alpha_j)$, $j = 1, \ldots, m$.
   c) (Optional) Fit a spline curve through the $2m$ coordinates just stored.

Similarly, to compute $\underline{Y}_{\mathrm{PR}}$:
1) Calculate $y_{Lr}(\alpha_j)$ and $y_{Rl}(\alpha_j)$, $j = 1, \ldots, p$, where $\alpha_p = \min_i h_{\underline{G}^i}$. To do this:
   a) Select appropriate p α-cuts for $\underline{Y}_{\mathrm{PR}}$ (e.g., divide [0, $\min_i h_{\underline{G}^i}$] into $p - 1$ intervals and set $\alpha_j = (\min_i h_{\underline{G}^i})(j - 1)/(p - 1)$, $j = 1, \ldots, p$).
   b) Find the α_j α-cut on $\underline{G}^i$ $(i = 1, \ldots, n)$.
   c) Use KM algorithms [6], [16] to find $y_{Lr}(\alpha_j)$ in (6) and $y_{Rl}(\alpha_j)$ in (7).
   d) Repeat steps (b) and (c) for every α_j $(j = 1, \ldots, p)$.
2) Construct $\underline{Y}_{\mathrm{PR}}$ from the p α-cuts. To do this:
   a) Store the left coordinates $(y_{Lr}(\alpha_j), \alpha_j)$, $j = 1, \ldots, p$.
   b) Store the right coordinates $(y_{Rl}(\alpha_j), \alpha_j)$, $j = 1, \ldots, p$.
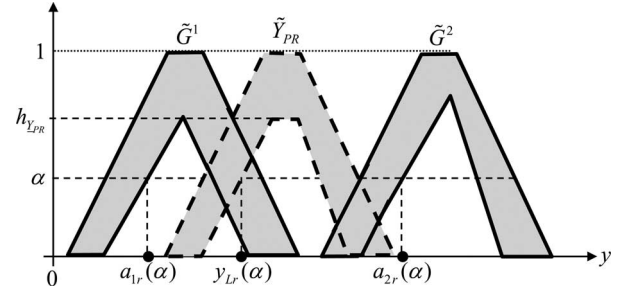
   c) (Optional) Fit a spline curve through the $2p$ coordinates just stored.

III. PR: PROPERTIES

All of the properties for PR that are described in this section help in demonstrating Assumption 1 for PR, namely, the result of combining fired rules using PR leads to an IT2 FS that resembles the three kinds of FOUs in a CWW codebook. Proofs of all theorems are given in the Appendix.

A. General Properties

Theorem 1: When all fired rules have the same consequent $\tilde{G}$, $\tilde{Y}_{\mathrm{PR}}$ defined in (3) is the same as $\tilde{G}$. ∎

An example where only one rule is fired is shown in Fig. 6.

Theorem 2: $\tilde{Y}_{\mathrm{PR}}$ is constrained by the consequents of the fired rules, i.e.,

$$\min_i a_{il}(\alpha) \leq y_{L1}(\alpha) \leq \max_i a_{il}(\alpha) \tag{9}$$

$$\min_i a_{ir}(\alpha) \leq y_{Lr}(\alpha) \leq \max_i a_{ir}(\alpha) \tag{10}$$

$$\min_i b_{il}(\alpha) \leq y_{Rl}(\alpha) \leq \max_i b_{il}(\alpha) \tag{11}$$

$$\min_i b_{ir}(\alpha) \leq y_{Rr}(\alpha) \leq \max_i b_{ir}(\alpha). \tag{12}$$

The equalities hold simultaneously if and only if all fired rules have the same consequent. ∎

Theorem 2 may be understood in this way: For PR using IT2 FSs, $\tilde{Y}_{\mathrm{PR}}$ cannot be smaller than the smallest consequent of the fired rules, and it also cannot be larger than the largest consequent of the fired rules. A graphical illustration of Theorem 2 is shown in Fig. 7. Assume only two rules are fired and $\tilde{G}^1$ lies to the left of $\tilde{G}^2$; then, $\tilde{Y}_{\mathrm{PR}}$ lies between $\tilde{G}^1$ and $\tilde{G}^2$.

Definition 1: $\underline{Y}_{\mathrm{PR}}$ is *trapezoidal looking* if its $\alpha = h_{\underline{Y}_{\mathrm{PR}}}$ α-cut is an interval instead of a single point. ∎

$\underline{Y}_{\mathrm{PR}}$ in Fig. 5 is trapezoidal looking.

Definition 2: $\underline{Y}_{\mathrm{PR}}$ is *triangle-looking* if its $\alpha = h_{\underline{Y}_{\mathrm{PR}}}$ α-cut converges to a single point. ∎

$\underline{Y}_{\mathrm{PR}}$ in Fig. 6 is triangle-looking.

Theorem 3: Generally, $\underline{Y}_{\mathrm{PR}}$ is trapezoidal-looking; however, $\underline{Y}_{\mathrm{PR}}$ is triangle-looking when all $\underline{G}^i$ are triangles with the same height h, *and* either of the following *unlikely* events is true.
1) The apexes of all $\underline{G}^i$ coincide, or
2) $\underline{f}^i = \overline{f}^i = f_i$ for all i. ∎

Fig. 7. Graphical illustration of Theorem 2 when two rules fire.

Theorem 4: Generally, $\bar{Y}_{\mathrm{PR}}$ is trapezoidal-looking; however, $\bar{Y}_{\mathrm{PR}}$ is triangle-looking when either of the following *unlikely* events is true.

1) All $\bar{G}^i$ are triangles and their apexes coincide; or
2) $\underline{f}^i = \overline{f}^i = f_i$ for all i.

Condition 1 in Theorems 3 and 4 is much less likely to occur than Condition 2. If Condition 2 occurs, then the firing interval reduces to a firing level, which can occur if both the antecedents in (1) and their associated inputs are T1 FSs (the consequents are still IT2 FSs).

B. Properties Related to Assumption 1

Theorem 5: Let $\tilde{Y}_{\mathrm{PR}}$ be defined in (3). Then, $\tilde{Y}_{\mathrm{PR}}$ is an LS if and only if:

1) At least one $\tilde{G}^i$ is an LS; and,
2) For every $\tilde{G}^i$ which is not an LS, the corresponding firing interval satisfies $\underline{f}^i = 0$. ∎

Theorem 5 demonstrates that $\tilde{Y}_{\mathrm{PR}}$ is an LS does not necessarily mean all consequents of the fired rules must be LSs.

Theorem 6: Let $\tilde{Y}_{\mathrm{PR}}$ be defined in (3). Then, $\tilde{Y}_{\mathrm{PR}}$ is a RS if and only if:

1) At least one $\tilde{G}^i$ is a RS; and,
2) For every $\tilde{G}^i$, which is not a RS, the corresponding firing interval satisfies $\underline{f}^i = 0$. ∎

Theorem 6 demonstrates that $\tilde{Y}_{\mathrm{PR}}$ is a RS does not necessarily mean all consequents of the fired rules must be RSs.

Theorem 7: Let $\tilde{Y}_{\mathrm{PR}}$ be defined in (3). Then, $\tilde{Y}_{\mathrm{PR}}$ is an interior FOU if $\tilde{G}^i$ do not satisfy the requirements in Theorems 5 and 6. More specifically, $\tilde{Y}_{\mathrm{PR}}$ is an interior FOU if and only if:

1) All $\tilde{G}^i$ are interior FOUs; or,
2) $\tilde{G}^i$ consist of more than one kind of shapes, and for each of at least two kinds of shapes, there exists at least one corresponding firing interval such that $\underline{f}^i > 0$. ∎

Theorem 7 demonstrates that $\tilde{Y}_{\mathrm{PR}}$ is an interior FOU but that does not necessarily mean all consequents of the fired rules must be interior FOUs.

Theorems 5–7 are important because they show that the output of PR is normal and similar to the word FOUs in a codebook (see Fig. 2). So, a similarity measure [34] can be used to map $\tilde{Y}_{\mathrm{PR}}$ to a word in the codebook. On the other hand, it is less intuitive to map a clipped FOU (see $\tilde{Y}_M$ in Fig. 6), as obtained from a Mamdani inference mechanism, or a crisp point, as obtained from the TSK inference mechanism, to a normal word FOU in the codebook.

IV. COMPUTING FIRING INTERVALS

A. General Results

In the IT2 FLS literature (e.g., [9], [16], [19]), computing the firing interval is simplest when inputs are modeled as singletons, more difficult when inputs are modeled as T1 FSs, and most difficult when inputs are modeled as IT2 FSs. Because rules in the Per-C are always activated by IT2 FSs, our concern must immediately be focused on computing the firing interval for this

most difficult case. Following are general results for computing the firing interval for this case [9], [16], [19].

Theorem 8: Let the p IT2 FS inputs that activate a collection of M rules be denoted $\tilde{\mathbf{X}}'$. Using the[9] minimum t-norm ($\wedge$), the results of the input and antecedent operations for the ith fired rule are contained in the *firing interval* $F^i(\tilde{\mathbf{X}}')$, which is given in (2), in which[10]

$$\underline{f}^i(\tilde{\mathbf{X}}') = \sup_{\mathbf{x}} \int_{x_1 \in X_1} \cdots \int_{x_p \in X_p} \left[\underline{\mu}_{\tilde{X}_1}(x_1) \wedge \underline{\mu}_{\tilde{F}_1^i}(x_1) \right]$$
$$\wedge \cdots \wedge \left[\underline{\mu}_{\tilde{X}_p}(x_p) \wedge \underline{\mu}_{\tilde{F}_p^i}(x_p) \right] \Big/ \mathbf{x} \qquad (13)$$

and

$$\overline{f}^i(\tilde{\mathbf{X}}') = \sup_{\mathbf{x}} \int_{x_1 \in X_1} \cdots \int_{x_p \in X_p} \left[\overline{\mu}_{\tilde{X}_1}(x_1) \wedge \overline{\mu}_{\tilde{F}_1^i}(x_1) \right]$$
$$\wedge \cdots \wedge \left[\overline{\mu}_{\tilde{X}_p}(x_p) \wedge \overline{\mu}_{\tilde{F}_p^i}(x_p) \right] \Big/ \mathbf{x}. \qquad (14)$$

∎

In evaluating (13) and (14), the supremum is attained when each term in brackets attains its supremum; hence, one needs to compute ($k = 1, \ldots, p$)

$$\underline{f}_k^i(\tilde{X}_k') \equiv \sup_{x_k} \int_{x_k \in X_k} \left[\underline{\mu}_{\tilde{X}_k}(x_k) \wedge \underline{\mu}_{\tilde{F}_k^i}(x_k) \right] \Big/ x_k \qquad (15)$$

and

$$\overline{f}_k^i(\tilde{X}_k') \equiv \sup_{x_k} \int_{x_k \in X_k} \left[\overline{\mu}_{\tilde{X}_k}(x_k) \wedge \overline{\mu}_{\tilde{F}_k^i}(x_k) \right] \Big/ x_k. \qquad (16)$$

In (15) and (16), let

$$\underline{\mu}_{\tilde{Q}_k^i}(x_k) \equiv \int_{x_k \in X_k} \left[\underline{\mu}_{\tilde{X}_k}(x_k) \wedge \underline{\mu}_{\tilde{F}_k^i}(x_k) \right] \Big/ x_k \qquad (17)$$

and

$$\overline{\mu}_{\tilde{Q}_k^i}(x_k) \equiv \int_{x_k \in X_k} \left[\overline{\mu}_{\tilde{X}_k}(x_k) \wedge \overline{\mu}_{\tilde{F}_k^i}(x_k) \right] \Big/ x_k \qquad (18)$$

so that (15) and (16) can be reexpressed, as:

$$\underline{f}_k^i(\tilde{X}_k') = \sup_{x_k} \underline{\mu}_{\tilde{Q}_k^i}(x_k) \qquad (19)$$

$$\overline{f}_k^i(\tilde{X}_k') = \sup_{x_k} \overline{\mu}_{\tilde{Q}_k^i}(x_k). \qquad (20)$$

Let $\underline{x}_{k,\max}^i$ and $\overline{x}_{k,\max}^i$ denote the values of x_k that are associated with $\sup_{x_k} \underline{\mu}_{\tilde{Q}_k^i}(x_k)$ and $\sup_{x_k} \overline{\mu}_{\tilde{Q}_k^i}(x_k)$, respectively. Then,

$$\underline{f}_k^i(\tilde{X}_k') = \underline{\mu}_{\tilde{Q}_k^i}(\underline{x}_{k,\max}^i) \qquad (21)$$

and

$$\overline{f}_k^i(\tilde{X}_k') = \overline{\mu}_{\tilde{Q}_k^i}(\overline{x}_{k,\max}^i). \qquad (22)$$

---

[9]Although our specific calculations in Section IV are only for the minimum t-norm, the results in this section are valid for both the minimum and product t-norms.

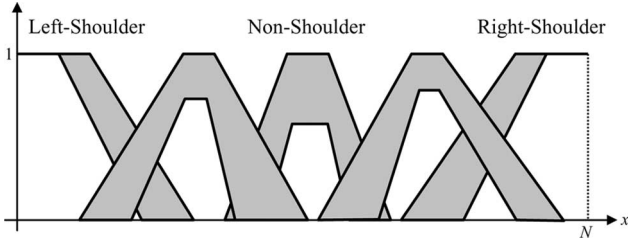[10]For a derivation of (13) and (14) using T1 FS mathematics, see [20].

Fig. 8. Typical FOUs for CWW.

This means, of course, that $\underline{f}^i(\tilde{\mathbf{X}}')$ and $\overline{f}^i(\tilde{\mathbf{X}}')$ in (13) and (14) can be reexpressed as[11]

$$\underline{f}^i(\tilde{\mathbf{X}}') = T_{k=1}^p \underline{f}_k^i(\tilde{X}_k') = T_{k=1}^p \underline{\mu}_{\tilde{Q}_k^i}(\underline{x}_{k,\max}^i) \quad (23)$$

and

$$\overline{f}^i(\tilde{\mathbf{X}}') = T_{k=1}^p \overline{f}_k^i(\tilde{X}_k') = T_{k=1}^p \overline{\mu}_{\tilde{Q}_k^i}(\bar{x}_{k,\max}^i). \quad (24)$$

Based on these discussions, the procedure to compute the firing interval $F^i(\tilde{\mathbf{X}}')$ in (2) is $(i = 1, \ldots, M)$:

1) Compute the functions $\underline{\mu}_{\tilde{Q}_k^i}(x_k)$ and $\overline{\mu}_{\tilde{Q}_k^i}(x_k)$, using the minimum t-norm in (17) and (18), respectively.

2) Compute $\underline{x}_{k,\max}^i$ and $\bar{x}_{k,\max}^i$ by maximizing $\underline{\mu}_{\tilde{Q}_k^i}(x_k)$ and $\overline{\mu}_{\tilde{Q}_k^i}(x_k)$, respectively.

3) Evaluate $\underline{f}_k^i(\tilde{X}_k')$ and $\overline{f}_k^i(\tilde{X}_k')$, using (21) and (22), respectively.

4) Compute $\underline{f}^i(\tilde{\mathbf{X}}')$ and $\overline{f}^i(\tilde{\mathbf{X}}')$ using (23) and (24), respectively.

Observe that steps 1–3 must be performed for all p antecedents, but these three steps can be done in parallel because there is no coupling among the calculations for each of the antecedents. This means that if we can perform these three steps for any one of the antecedents, the results can be applied to all p antecedents.

Next, steps 1–3 are carried out for the minimum t-norm and the FOUs that can occur in the Per-C. Although the rest of this section may appear to be tedious, without its results a rule-based CWW engine cannot be implemented. Consequently, these details are provided in order to make perceptual computing readily available to all readers of this journal.

B. Specific Results

Earlier, we explained that word FOUs have three canonical shapes—LS, RS, and interior [referred to here as a *nonshoulder* (NS)]—for which all LMFs and UMFs are piecewise-linear. Here firing intervals are computed for the FOUs depicted in Fig. 8, which include the word FOUs as a special case when the LMFs of the NS FOUs are triangles instead of trapezoids. The more general FOUs can occur when using an LWA CWW Engine.

What complicates the firing interval calculations is that a rule input[12] $\tilde{X}$ can be LS, RS, or NS, and for each of these

---

[11]The notation $T_{k=1}^p f_k$ denotes $p-1$ successive t-norms, i.e., $f_1 \wedge f_2 \wedge \ldots \wedge f_p$.

[12]In this section, the notation is simplified by not showing subscripts or superscripts on all IT2 FSs.
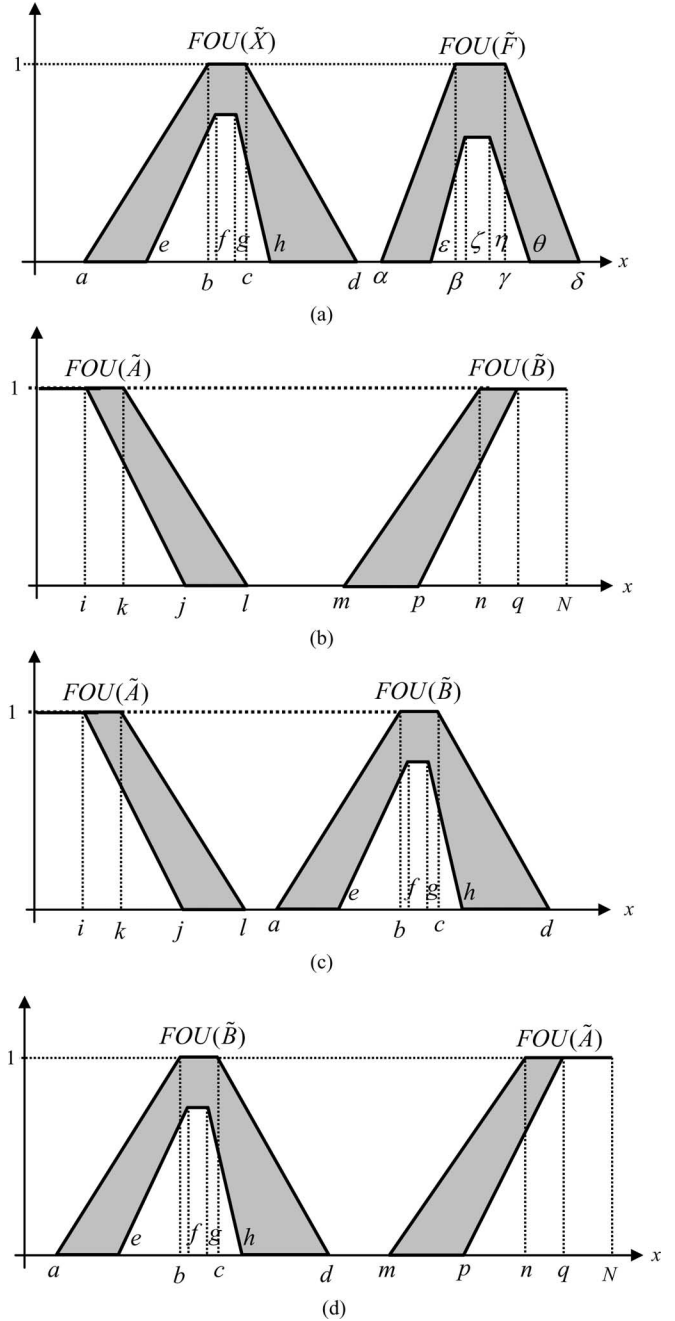

Fig. 9. Possible combinations of input and antecedent FOUs: (a) (NS, NS), (b) (LS, RS), (c) (LS, NS), and (d) (NS, RS). $\tilde{A}$ and $\tilde{B}$ may be either $\tilde{X}$ or $\tilde{F}$.

possibilities, its associated antecedent $\tilde{F}$ can also be LS, RS, or NS; hence, there are nine different cases that have to be considered—$(\tilde{X}, \tilde{F}) = $ (LS, LS), (LS, NS), (LS, RS), (NS, LS), (NS, NS), (NS, RS), (RS, LS), (RS, NS), and (RS, RS). Fig. 9 summarizes seven of these cases (Fig. 9(b)–(d) each apply to two cases, because $\tilde{A}$ and $\tilde{B}$ may be either $\tilde{X}$ or $\tilde{F}$). The two cases (LS, LS) and (RS, RS) are so simple that no figures are shown for them.

Regardless of which case one is in, all of the sup-min calculations use the results that are in the following example by making appropriate symbolic transformations.
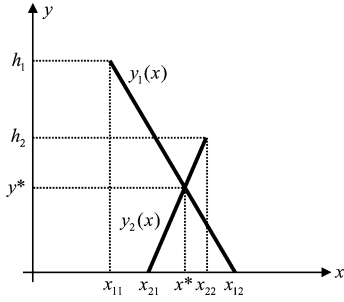
Fig. 10. Two intersecting lines, used for the calculation of $\sup_{x \in X} \min[y_1(x), y_2(x)]$.

Example 1: Here, the major computations that are used to evaluate $\sup_{x_m \in X_m} \min[\mu_{X_m}(x_m), \mu_{F_m^i}(x_m)]$ for different kinds of piecewise linear MFs are described. The situation is the one depicted in Fig. 10. Simplifying the notation, our objective is to compute $\sup_{x \in X} \min[y_1(x), y_2(x)]$, where:

$$y_1(x) = \begin{cases} \dfrac{h_1(x_{12} - x)}{(x_{12} - x_{11})}, & x_{11} \le x \le x_{12} \\ 0, & \text{otherwise} \end{cases} \quad (25)$$

$$y_2(x) = \begin{cases} \dfrac{h_2(x - x_{21})}{(x_{22} - x_{21})}, & x_{21} \le x \le x_{22} \\ 0, & \text{otherwise.} \end{cases} \quad (26)$$

Examining Fig. 10, it is easy to see that $\sup_{x \in X} \min[y_1(x), y_2(x)]$ occurs at the intersection of $y_1(x)$ and $y_2(x)$, i.e., when $x = x^*$ and $y = y^*$. It is straightforward to show that

$$x^* = \frac{h_1 x_{12}(x_{22} - x_{21}) + h_2 x_{21}(x_{12} - x_{11})}{h_1(x_{22} - x_{21}) + h_2(x_{12} - x_{11})} \quad (27)$$

$$\sup_{x \in X} \min[y_1(x), y_2(x)] = y^*$$

$$= \frac{h_1 h_2 (x_{12} - x_{21})}{h_1(x_{22} - x_{21}) + h_2(x_{12} - x_{11})}. \quad (28)$$

When $h_1 = h_2 = 1$, (27) and (28) simplify to

$$x^* = \frac{x_{12} x_{22} - x_{21} x_{11}}{(x_{22} - x_{21}) + (x_{12} - x_{11})} \quad (29)$$

$$\sup_{x \in X} \min[y_1(x), y_2(x)] = y^*$$

$$= \frac{x_{12} - x_{21}}{(x_{22} - x_{21}) + (x_{12} - x_{11})}. \quad (30)$$

∎

In order to illustrate the sup-min calculations, they are summarized in Tables I and II for the important case of (NS, NS). For the calculations of $\bar{f}$ that are given in Table I, there are two cases. When $d > \alpha$ and $c < \beta$, the trapezoidal $\text{UMF}(\tilde{X})$ is to the left of the trapezoidal $\text{UMF}(\tilde{F})$, whereas when $d > \alpha$ and $b > \gamma$, the trapezoidal $\text{UMF}(\tilde{F})$ is to the left of the trapezoidal $\text{UMF}(\tilde{X})$. In both cases, (29) and (30) were used to compute x^* and $y^* = \bar{f}$.

For the calculations of $\underline{f}$ that are given in Table II, there are also two cases. Case 1 occurs as long as r_X and l_F intersect. This case begins when $h_{\tilde{F}}$ intersects r_X [slide the dashed trapezoid for $\text{LMF}(\tilde{F})$ to the left so that the topmost point of

l_F touches r_X], and an analysis of when this occurs leads to the inequality for ζ that is stated in the table. Case 2 occurs as long as r_F and l_X intersect, and an analysis of when this occurs leads to the inequality for η that is also stated in the table. For both cases, (27) and (28) were used to compute x^* and $y^* = \underline{f}$.

The calculations of $\underline{f}$ and $\bar{f}$ for the remaining eight cases—(LS, LS), (LS, NS), (LS, RS), (NS, LS), (NS, RS), (RS, LS), (RS, NS), and (RS, RS)—although seemingly tedious are very straightforward. When $\tilde{X}$ or $\tilde{F}$ are *both* left or RSs, and the tops of their two trapezoidal MFs overlap (even at just one point), then $\sup_{\forall x} \min[\text{LMF}(\tilde{X}), \text{LMF}(\tilde{F})] = 1$ and $\sup_{\forall x} \min[\text{UMF}(\tilde{X}), \text{UMF}(\tilde{F})] = 1$, so that $\underline{f} = \bar{f} = 1$; hence, the cases (LS, LS) and (RS, RS) are not tedious at all. The details of the cases (LS, NS), (LS, RS), (NS, LS), (NS, RS), (RS, LS), (RS, NS) are left to the reader.

In order to see the forest from the trees (so to speak), summary flowcharts are provided in Figs. 11–15. They are very useful for writing a computer program to carry out all of the firing interval computations. To use these flowcharts, it is assumed that all FOUs have been prelabeled as either LS, RS, or NS.

The entry flowchart in Fig. 11 establishes which one of the nine possible cases for the input FOU and antecedent FOU has occurred.

The flowchart in Fig. 12 handles the cases (LS, LS), (RS, RS), (LS, RS), and (RS, LS). It disposes of the cases (LS, LS) and (RS, RS) very easily, and treats the cases (LS, RS) and (RS, LS) simultaneously by using the $\tilde{A} - \tilde{B}$ labeling described earlier. It includes the situations when the FOUs do not overlap or overlap in different ways.

The flowcharts in Figs. 13 and 14 handle the case (NS, NS). Fig. 13 is for the LMF calculations and is based on Table II. Fig. 14 is for the UMF calculations and is based on Table I. These flowcharts also include the case when there is no overlap between the two FOUs.

The flowchart in Fig. 15 handles the cases (LS, NS), (NS, LS), (RS, NS), and (NS, RS). Its left-hand path handles the cases (LS, NS), (NS, LS) and its right-hand path the cases (RS, NS), (NS, RS). Both paths include the situations when the FOUs do or do not overlap.

V. DISCUSSION

A. Applications for PR

A family of applications for which PR can be used is *distributed and (or) hierarchical decision making*. One example of such a situation is when there are q judges (or experts, managers, commanders, referees, etc.) who have to provide a subjective decision or judgment about a situation (e.g., quality of a submitted journal article). They will do this by providing a linguistic evaluation (i.e., a word, term, or phrase) for a collection of prespecified and preranked categories, using a prespecified codebook of terms, because it may be too problematic to provide a numerical score for these categories.

Prior to the judging, FOUs are determined for all of the words in the codebook, the categories will have had linguistic weights

TABLE I
SUP-MIN CALCULATIONS FOR TRAPEZOIDAL UMFs

UMF Definitions: $UMF(\tilde{X}) = (a,b,c,d)$, $UMF(\tilde{F}) = (\alpha,\beta,\gamma,\delta)$
Equations of UMF line segments: $r_{\underline{X}} : y = (d-x)/(d-c)$, $l_{\underline{X}} : y = (x-a)/(b-a)$,
$r_{\underline{F}} : y = (\delta-x)/(\delta-\gamma)$, $l_{\underline{F}} : y = (x-\alpha)/(\beta-\alpha)$

Case 1. $d > \alpha$ and $c < \beta$

$$x^* = \frac{d\beta - \alpha c}{(d-c)+(\beta-\alpha)}$$

$$\overline{f} = \frac{d-\alpha}{(d-c)+(\beta-\alpha)}$$

Case 2. $d > \alpha$ and $b > \gamma$

$$x^* = \frac{\delta b - a\gamma}{(\delta-\gamma)+(b-a)}$$

$$\overline{f} = \frac{\delta-a}{(\delta-\gamma)+(b-a)}$$

TABLE II
SUP-MIN CALCULATIONS FOR TRIANGLE LMFs: $h_{\underline{F}} < h_{\underline{X}}$

LMF Definitions: $LMF(\tilde{X}) = (e,f,g,h)$, $LMF(\tilde{F}) = (\varepsilon,\zeta,\eta,\theta)$, $\max_{\forall x} LMF(\tilde{X}) = h_{\underline{X}}$,
$\max_{\forall x} LMF(\tilde{F}) = h_{\underline{F}}$
Equations of LMF line segments: $r_{\underline{X}} : y = h_{\underline{X}}(h-x)/(h-g)$, $l_{\underline{X}} : y = h_{\underline{X}}(x-e)/(f-e)$,
$r_{\underline{F}} : y = h_{\underline{F}}(\theta-x)/(\theta-\eta)$, $l_{\underline{F}} : y = h_{\underline{F}}(x-\varepsilon)/(\zeta-\varepsilon)$
Calculation of $h_{\underline{F}} = r_{\underline{X}} : x'' = [h(h_{\underline{X}} - h_{\underline{F}}) + g h_{\underline{F}}]/h_{\underline{X}}$
Calculation of $h_{\underline{F}} = l_{\underline{X}} : x' = [e(h_{\underline{X}} - h_{\underline{F}}) + f h_{\underline{F}}]/h_{\underline{X}}$

Case 1: $\sup\min_{\forall x}[LMF(\tilde{X}), LMF(\tilde{F})]$ occurs at the intersection of $r_{\underline{X}}$ and $l_{\underline{F}}$

$$\zeta \geq \frac{h(h_{\underline{X}} - h_{\underline{F}}) + g h_{\underline{F}}}{h_{\underline{X}}}$$

$$x^* = \frac{h_{\underline{X}} h(\zeta - \varepsilon) + h_{\underline{F}} \varepsilon(h-g)}{h_{\underline{X}}(\zeta - \varepsilon) + h_{\underline{F}}(h-g)}$$

$$\underline{f} = \frac{h_{\underline{F}} h_{\underline{X}}(h-\varepsilon)}{h_{\underline{X}}(\zeta - \varepsilon) + h_{\underline{F}}(h-g)}$$

Case 2: $\sup\min_{\forall x}[LMF(\tilde{X}), LMF(\tilde{F})]$ occurs at the intersection of $l_{\underline{X}}$ and $r_{\underline{F}}$

$$\eta \leq \frac{e(h_{\underline{X}} - h_{\underline{F}}) + f h_{\underline{F}}}{h_{\underline{X}}}$$

$$x^* = \frac{h_{\underline{F}} \theta(f-e) + h_{\underline{X}} e(\theta - \eta)}{h_{\underline{X}}(\theta - \eta) + h_{\underline{F}}(f-e)}$$

$$\underline{f} = \frac{h_{\underline{F}} h_{\underline{X}}(\theta - e)}{h_{\underline{X}}(\theta - \eta) + h_{\underline{F}}(f-e)}$$
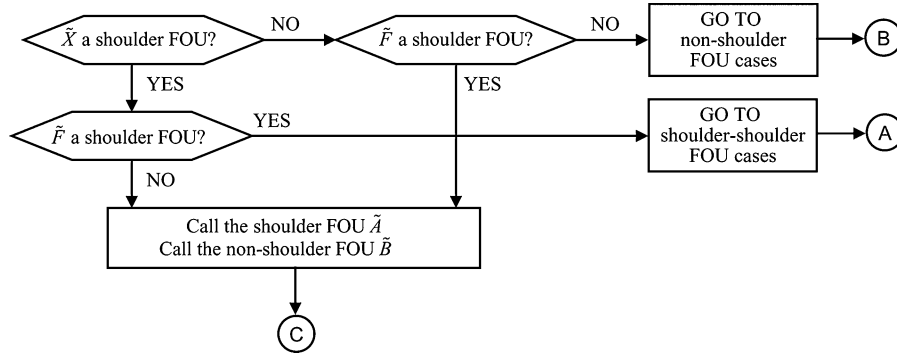
Fig. 11.   Entry flowchart.

Fig. 12.   Sup-min calculations when both of the FOUs are shoulders. See Fig. 9(b) for definitions of notation.
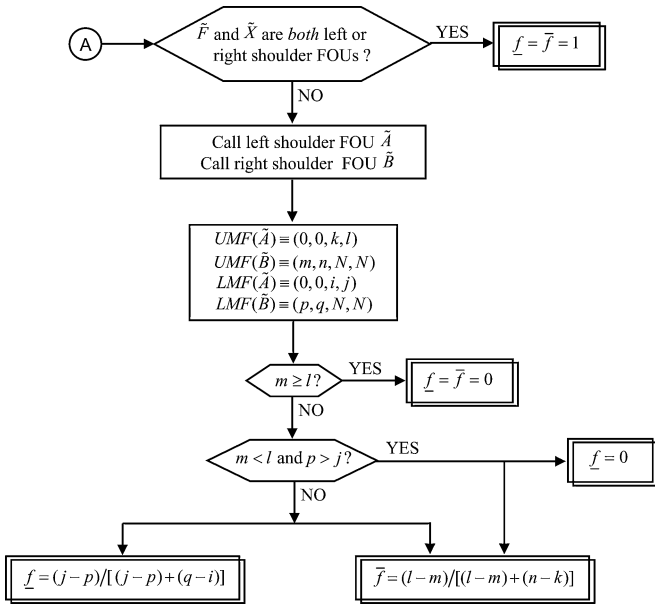

Fig. 13.   Sup-min LMF calculations when both MFs are trapezoids. See Fig. 9(a) for definitions of notation.

assigned to them, and those weights will also have had FOUs determined for them. The judges do not have to be concerned with any of the *a priori* rankings and modeling; it will all have been done before the judges will be asked to judge.

After the judges assign their linguistic scores to each category, an LWA can be computed for each of the judges. The q LWAs are then sent to a control (command) center (e.g., the Associate Editor); however, because judges may not be of equal expertise, each judge's level-of-expertise will have also been prespecified using a linguistic term provided by the judge from a small vocabulary of terms. The q LWAs can also be aggregated by another LWA and it is also sent to the control center, where a final decision or judgment is made. LWA FOUs will look like the ones in Fig. 8.

At the control center, one way in which a decision can be made is by using a collection of IF-THEN rules. These rules will be activated by a combination of the LWA FOUs from the individual judges and the aggregated LWA FOU. Exactly what the rules would look like is beyond the scope of this paper. Fired
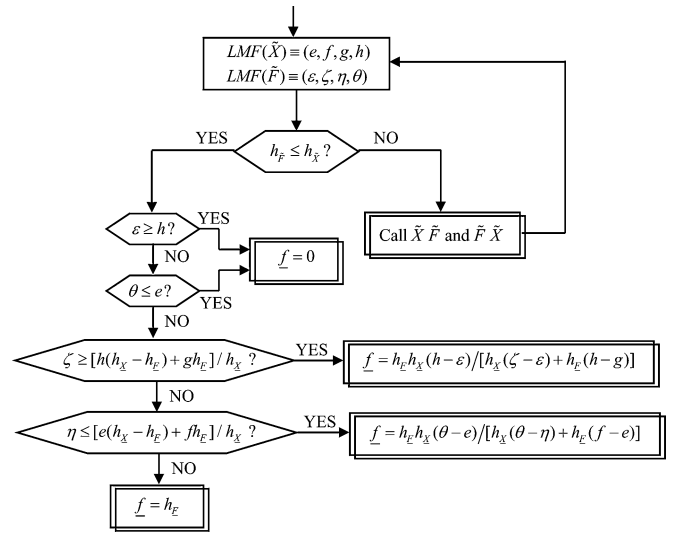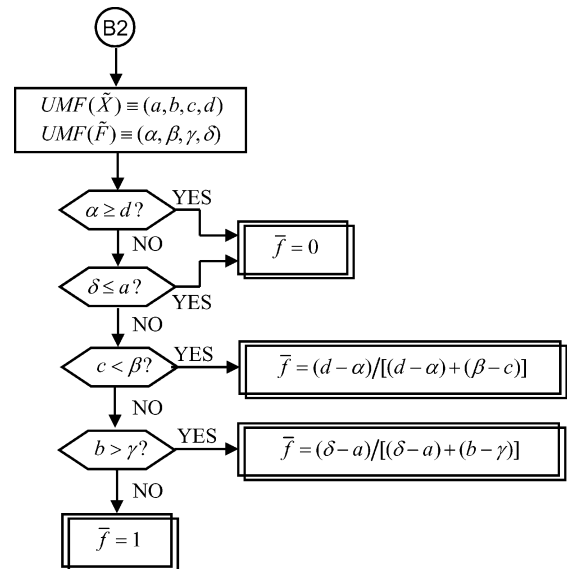

Fig. 14.   Sup-min UMF calculations when both MFs are trapezoids. See Fig. 9(a) for definitions of notation.
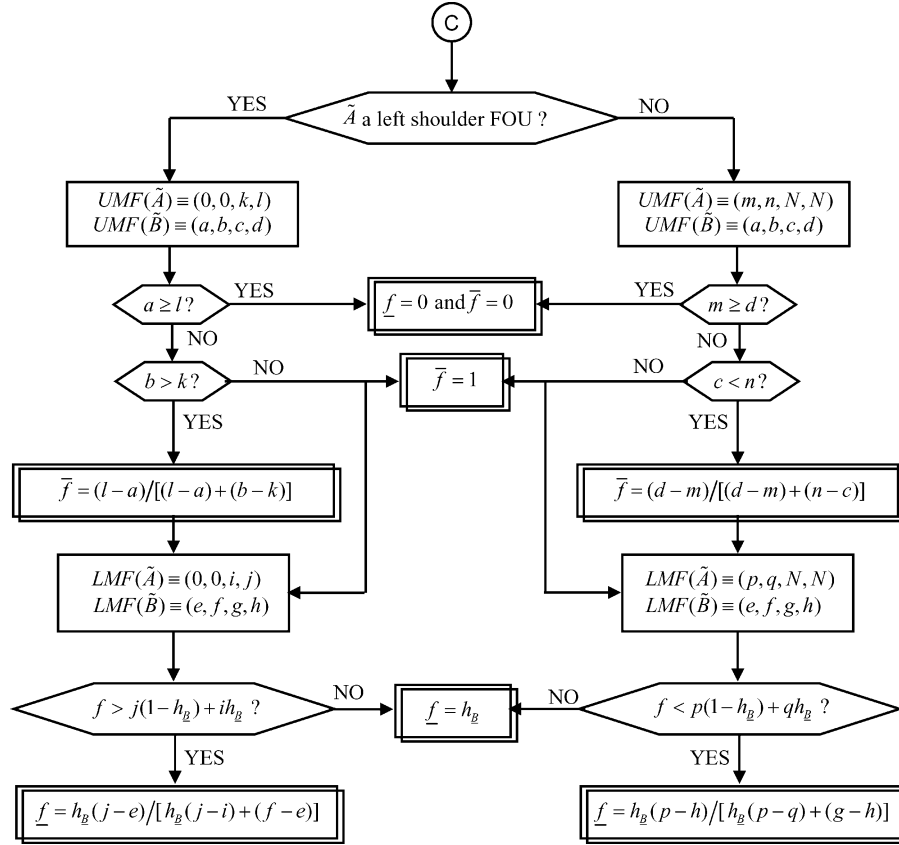
Fig. 15. Sup-min calculations when one of the FOUs is a shoulder and the other FOU is an NS.

rules would be combined using PR, after which (see Fig. 1) the PR FOU would be decoded into a suitable word by the Per-C decoder. It is expected that similarity and ranking will play important roles in accomplishing this. If the final result is a preestablished category (e.g., accept, rewrite, reject), instead of any word from the codebook, then a rule-based classifier would be used as part of the decoder, where again PR would be used to aggregate the fired rules prior to the classification.

We hope that by explaining some high-level details of this application the readers will find other applications for PR.

B. PR Versus IT2 Mamdani

A reviewer wanted to know how different the results would be by using PR versus, e.g., an IT2 Mamdani rule-based fuzzy logic system. While we do not have a concrete answer to this question (it would be application dependent, and would also depend upon the number of words in the codebook), we would like to make some general observations relating to the question as it applies to the Per-C.

1) In the Per-C that uses PR, the flow of computations is: IT2 FSs activate a set of rules, firing intervals are computed, and PR is used to compute $\mathrm{FOU}(\tilde{Y}_{\mathrm{PR}})$, after which $\mathrm{FOU}(\tilde{Y}_{\mathrm{PR}})$ is mapped into a word by the decoder. As proven in this paper, $\mathrm{FOU}(\tilde{Y}_{\mathrm{PR}})$ will resemble the FOUs in the codebook, and we believe this to be desirable from a human reasoning point-of-view. Additionally, no infor-

mation is lost by mapping directly from $\mathrm{FOU}(\tilde{Y}_{\mathrm{PR}})$ into a word.

2) In an IT2 Mamdani rule-based fuzzy logic system, the flow of computations is: IT2 FSs activate a set of rules, firing intervals are computed, and a decision must then be made about how to proceed. In one approach, the firing intervals are combined with their respective IT2 FS consequents, resulting in an IT2 fired-rule output FS, after which the union of all such IT2 FSs is computed, the result being $\mathrm{FOU}(\tilde{Y}_{M|\mathrm{Union}})$. If n rules fire, $\mathrm{FOU}(\tilde{Y}_{M|\mathrm{Union}})$ will be very broad and will not remotely resemble any of the word FOUs in the codebook. We believe this to be undesirable from a human reasoning point-of-view. It may happen though that when $\mathrm{FOU}(\tilde{Y}_{M|\mathrm{Union}})$ is decoded by the decoder, exactly the same word is obtained as obtained from $\mathrm{FOU}(\tilde{Y}_{\mathrm{PR}})$. We are presently studying when and if this can occur. Note that for this kind of an IT2 Mamdani rule-based fuzzy logic system, no information is lost by mapping directly from $\mathrm{FOU}(\tilde{Y}_{M|\mathrm{Union}})$ into a word.

In another approach, the firing intervals and their respective IT2 FS consequents are used in a type-reduction method, the result being an interval-valued set $Y_{M|\mathrm{TR}}$, which must then be decoded into a word by the decoder. Although there are different ways in which this can be done (e.g., the centroid of each word in the codebook can be precomputed, and then the word whose centroid is most similar to $Y_{M|\mathrm{TR}}$ would be chosen; or, the center-of gravity of each word's centroid can be precomputed,
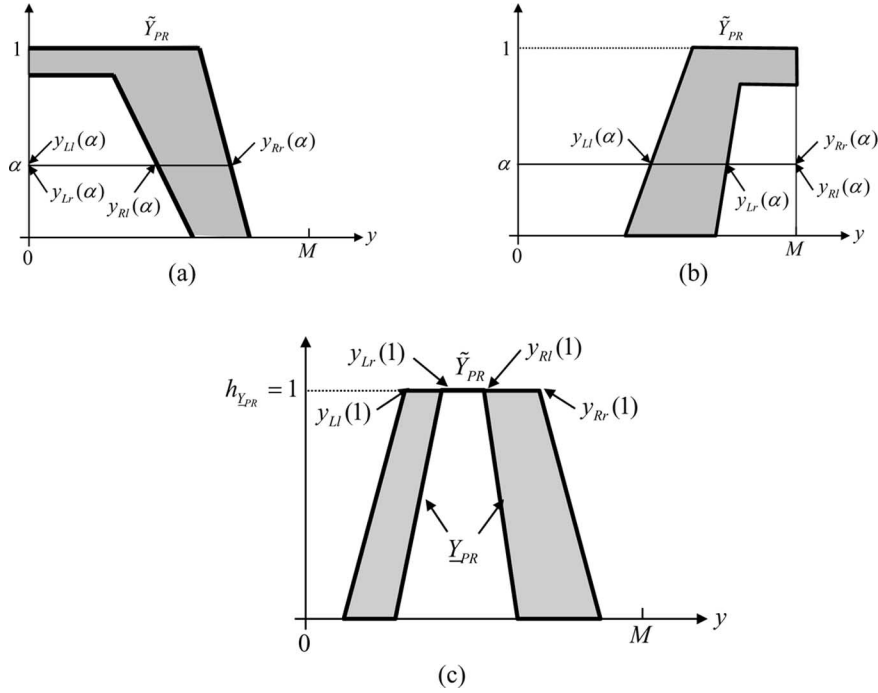
Fig. 16. α-cuts on an (a) LS $\tilde{Y}_{\mathrm{PR}}$, (b) RS $\tilde{Y}_{\mathrm{PR}}$, and (c) interior $\tilde{Y}_{\mathrm{PR}}$ with $h_{\underline{Y}_{\mathrm{PR}}} = 1$.

and then the word whose centroid's center-of-gravity is most similar to the center-of-gravity of $Y_{M|\mathrm{TR}}$ would be chosen), we are concerned by the fact that some information about the word is lost through type reduction, and even more information is lost when the center-of-gravity of a centroid is computed. Even with all of this lost information, it may still happen that the resulting word is the same as that obtained from $\mathrm{FOU}(\tilde{Y}_{\mathrm{PR}})$. We do not plan to study when or if this can occur because losing information does not seem like a good thing to us.

VI. CONCLUSION

A new CWW engine—PR—has been proposed in this paper. It uses IF-THEN rules; however, unlike traditional IF-THEN rules that use Mamdani or TSK models, PR uses an LWA to combine fired rules. It is proved that combining fired rules using the LWA leads to an IT2 FS whose FOU resembles three kinds of FOUs that have previously been shown to model words (interior FOU, LS FOU, and RS FOU), something that the authors feel is a highly desirable property for perceptual computing. To the best knowledge of the authors, no other method of approximate reasoning has this property.

Because IT2 FS word models that would be used in a rule-based CWW engine for both rule antecedents and inputs to the rules are piecewise linear interior, LS, or RS FOUs, it is possible to precompute the firing intervals for such an engine. These closed-form formulas are also in this paper, and although tedious to derive, because there are many cases that have to be considered, they are easy to program.

Referring to the Per-C in Fig. 1, the results in this paper now let a rule-based CWW engine be fully implemented. Future publications will explain all components of the Per-C and will illustrate its applications to distributed hierarchical decision-making situations.

APPENDIX

PROOFS OF THE THEOREMS IN SECTION III

Proofs of Theorems 1–7 are provided in this appendix. To begin, some preliminary results are provided.

A.1: Preliminary Results

Lemma 1: Let $y_{Lr}(\alpha)$ be defined in (6), where $a_{ir}(\alpha)$ have been sorted in ascending order and $\underline{f}^i \geq 0$. The properties of $y_{Lr}(\alpha)$ include [12]:

1) Because $y_{Lr}(\alpha)$ is a weighted average of $a_{ir}(\alpha)$, and $\underline{f}^i \geq 0$,

$$a_{1r}(\alpha) \leq y_{Lr}(\alpha) \leq a_{nr}(\alpha). \tag{A1}$$

2) $y_{Lr}(\alpha)$ is a nondecreasing function of $a_{ir}(\alpha)$.
3) $y_{Lr}(\alpha)$ can be reexpressed as

$$y_{Lr}(\alpha) = \min_{k \in [1, n-1]} \frac{\sum_{i=1}^{k} a_{ir}(\alpha)\overline{f}^i + \sum_{i=k+1}^{n} a_{ir}(\alpha)\underline{f}^i}{\sum_{i=1}^{k} \overline{f}^i + \sum_{i=k+1}^{n} \underline{f}^i} \tag{A2}$$

and can be computed by a KM algorithm [16], [35]. ∎

Note that $y_{Ll}(\alpha), y_{Rl}(\alpha)$, and $y_{Rr}(\alpha)$ have similar properties. These properties, whose proofs are in [12], will be used heavily in proving the theorems in this section.

Lemma 2: An IT2 FS $\tilde{Y}_{\mathrm{PR}}$ is an LS [see Fig. 16(a)] if and only if $y_{Ll}(1) = 0$ and $y_{Lr}(h_{\underline{Y}_{\mathrm{PR}}}) = 0$. ∎

Proof: Intuitively, an IT2 FS $\tilde{Y}_{\text{PR}}$ is an LS if and only if $y_{Ll}(\alpha) = 0$ for $\forall \alpha \in [0, 1]$ and $y_{Lr}(\alpha) = 0$ $\forall \alpha \in [0, h_{\underline{Y}_{\text{PR}}}]$, as shown in Fig. 16(a). Because only convex IT2 FSs are used in PR, we have $y_{Ll}(\alpha) \leq y_{Ll}(1)$ $\forall \alpha \in [0, 1]$. Consequently, $y_{Ll}(1) = 0$ means $y_{Ll}(\alpha) = 0$ $\forall \alpha \in [0, 1]$. Similarly, $y_{Lr}(h_{\underline{Y}_{\text{PR}}}) = 0$ means $y_{Lr}(\alpha) = 0$ $\forall \alpha \in [0, h_{\underline{Y}_{\text{PR}}}]$.

Lemma 3: An IT2 FS $\tilde{Y}_{\text{PR}}$ is a RS [see Fig. 16(b)] if and only if $y_{Rr}(1) = M$ and $y_{Rl}(h_{\underline{Y}_{\text{PR}}}) = M$. ∎

The proof of Lemma 3 is so similar to that of Lemma 2 that it is left to the reader.

Lemma 4. An IT2 FS $\tilde{Y}_{\text{PR}}$ is an interior FOU if and only if $y_{Lr}(h_{\underline{Y}_{\text{PR}}}) > 0$ and $y_{Rl}(h_{\underline{Y}_{\text{PR}}}) < M$. ∎

An example of interior $\tilde{Y}_{\text{PR}}$ is shown in Fig. 16(c).

Proof: When $y_{Lr}(h_{\underline{Y}_{\text{PR}}}) > 0$ and $y_{Rl}(h_{\underline{Y}_{\text{PR}}}) < M$, $\tilde{Y}_{\text{PR}}$ is not an LS by Lemma 2, and it is also not an RS by Lemma 3. Consequently, $\tilde{Y}_{\text{PR}}$ must be an interior FOU.

A.2: Proof of Theorem 1

When all fired rules have the same consequent $\tilde{G}$, (3) is simplified to[13]

$$\tilde{Y}_{\text{PR}} = \frac{\sum_{i=1}^{n} F^i(\tilde{\mathbf{X}}')\tilde{G}}{\sum_{i=1}^{n} F^i(\tilde{\mathbf{X}}')}. \tag{A3}$$

Denote the α-cut on $\bar{G}$ as $[a_l(\alpha), b_r(\alpha)]$ ($\alpha \in [0, 1]$) and the α-cut on $\underline{G}$ as $[a_r(\alpha), b_l(\alpha)]$ ($\alpha \in [0, h_{\underline{G}}]$). Then, the α-cuts on $\tilde{Y}_{\text{PR}}$ in (4)–(7) are computed as

$$y_{Ll}(\alpha) = \min_{f_i \in [\underline{f}^i, \bar{f}^i]} \frac{\sum_{i=1}^{n} a_l(\alpha)f_i}{\sum_{i=1}^{n} f_i} = a_l(\alpha), \qquad \alpha \in [0, 1] \tag{A4}$$

$$y_{Rr}(\alpha) = \max_{f_i \in [\underline{f}^i, \bar{f}^i]} \frac{\sum_{i=1}^{n} b_r(\alpha)f_i}{\sum_{i=1}^{n} f_i} = b_r(\alpha), \qquad \alpha \in [0, 1] \tag{A5}$$

$$y_{Lr}(\alpha) = \min_{f_i \in [\underline{f}^i, \bar{f}^i]} \frac{\sum_{i=1}^{n} a_r(\alpha)f_i}{\sum_{i=1}^{n} f_i} = a_r(\alpha), \qquad \alpha \in [0, h_{\underline{G}}] \tag{A6}$$

$$y_{Rl}(\alpha) = \max_{f_i \in [\underline{f}^i, \bar{f}^i]} \frac{\sum_{i=1}^{n} b_{il}(\alpha)f_i}{\sum_{i=1}^{n} f_i} = b_l(\alpha), \qquad \alpha \in [0, h_{\underline{G}}] \tag{A7}$$

i.e.,

$$[y_{Ll}(\alpha), y_{Rr}(\alpha)] = [a_l(\alpha), b_r(\alpha)], \qquad \alpha \in [0, 1] \tag{A8}$$

$$[y_{Lr}(\alpha), y_{Rl}(\alpha)] = [a_r(\alpha), b_l(\alpha)], \qquad \alpha \in [0, h_{\underline{G}}]. \tag{A9}$$

Because every α-cut on $\tilde{Y}_{\text{PR}}$ is the same as the corresponding α-cut on $\tilde{G}$, it follows that $\tilde{Y}_{\text{PR}} = \tilde{G}$.

---

[13]Recall that (3) is an "expressive" equation, so we cannot "cancel" $F^i(\tilde{\mathbf{X}}')$ in its numerator and denominator.

A.3: Proof of Theorem 2

Equation (10) is readily seen from Part 1 of Lemma 1. The other three inequalities can be proved similarly.

When all n fired rules have the same consequent $\tilde{G}$, we know from Theorem 1 that

$$y_{Ll}(\alpha) = \min_{\forall i} a_{il}(\alpha) = \max_{\forall i} a_{il}(\alpha) = a_l(\alpha) \tag{A10}$$

$$y_{Lr}(\alpha) = \min_{\forall i} a_{ir}(\alpha) = \max_{\forall i} a_{ir}(\alpha) = a_r(\alpha) \tag{A11}$$

$$y_{Rl}(\alpha) = \min_{\forall i} b_{il}(\alpha) = \max_{\forall i} b_{il}(\alpha) = b_l(\alpha) \tag{A12}$$

$$y_{Rr}(\alpha) = \min_{\forall i} b_{ir}(\alpha) = \max_{\forall i} b_{ir}(\alpha) = b_r(\alpha). \tag{A13}$$

When all $\tilde{G}^i$ are not the same, at least one of (A10)–(A13) does not hold; hence, equalities in (9)–(12) hold simultaneously if and only if all fired rules have the same consequent.

A.4: Proof of Theorem 3

Because (see Fig. 4) $b_{il}(\alpha) \geq a_{ir}(\alpha)$, observe from (6) and (7) that

$$y_{Lr}(\alpha) = \min_{\forall f_i \in [\underline{f}^i, \bar{f}^i]} \frac{\sum_{i=1}^{n} a_{ir}(\alpha)f_i}{\sum_{i=1}^{n} f_i} \leq \max_{\forall f_i \in [\underline{f}^i, \bar{f}^i]} \frac{\sum_{i=1}^{n} a_{ir}(\alpha)f_i}{\sum_{i=1}^{n} f_i}$$

$$\leq \max_{\forall f_i \in [\underline{f}^i, \bar{f}^i]} \frac{\sum_{i=1}^{n} b_{il}(\alpha)f_i}{\sum_{i=1}^{n} f_i} = y_{Rl}(\alpha) \tag{A14}$$

i.e., $y_{Lr}(\alpha) \leq y_{Rl}(\alpha)$, so that in general $\underline{Y}_{\text{PR}}$ is trapezoidal-looking (see Fig. 5).

If all $\underline{G}^i$ are triangles with the same height h, then according to (8), $h_{\underline{Y}_{\text{PR}}} = \min_i h_{i\underline{G}^i} = h$. If, in addition, the apexes of all $\underline{G}^i$ coincide at $x = \lambda$, then the α-cuts on all of them collapse to a point, i.e., $a_{ir}(h) = b_{il}(h) = \lambda$ $\forall i = 1, \ldots, n$. Consequently

$$y_{Lr}(h) = \min_{\forall f_i \in [\underline{f}^i, \bar{f}^i]} \frac{\sum_{i=1}^{n} a_{ir}(h)f_i}{\sum_{i=1}^{n} f_i} = \lambda \min_{\forall f_i \in [\underline{f}^i, \bar{f}^i]} \frac{\sum_{i=1}^{n} f_i}{\sum_{i=1}^{n} f_i} = \lambda \tag{A15}$$

$$y_{Rl}(h) = \max_{\forall f_i \in [\underline{f}^i, \bar{f}^i]} \frac{\sum_{i=1}^{n} b_{il}(h)f_i}{\sum_{i=1}^{n} f_i} = \lambda \max_{\forall f_i \in [\underline{f}^i, \bar{f}^i]} \frac{\sum_{i=1}^{n} f_i}{\sum_{i=1}^{n} f_i} = \lambda \tag{A16}$$

i.e., $y_{Lr}(h) = y_{Rl}(h) = \lambda$; hence, $\underline{Y}_{\text{PR}}$ is triangle-looking with height h.

Finally, if all $\underline{G}^i$ are triangles with the same height h, then $a_{ir}(h) = b_{il}(h)$, and $\underline{Y}_{\text{PR}}$ also has height h. In addition, if $\underline{f}^i = \bar{f}^i = f_i$, then

$$y_{Lr}(h) = \min_{\forall f_i \in [\underline{f}^i, \bar{f}^i]} \frac{\sum_{i=1}^{n} a_{ir}(h)f_i}{\sum_{i=1}^{n} f_i} = \frac{\sum_{i=1}^{n} a_{ir}(h)f_i}{\sum_{i=1}^{n} f_i} \tag{A17}$$

$$y_{Rl}(h) = \max_{\forall f_i \in [\underline{f}^i, \bar{f}^i]} \frac{\sum_{i=1}^{n} b_{il}(h)f_i}{\sum_{i=1}^{n} f_i} = \frac{\sum_{i=1}^{n} b_{il}(h)f_i}{\sum_{i=1}^{n} f_i}$$

$$= \frac{\sum_{i=1}^{n} a_{ir}(h)f_i}{\sum_{i=1}^{n} f_i} = y_{Lr}(h). \tag{A18}$$

Hence, again, $\underline{Y}_{\text{PR}}$ is triangle-looking with height h.

A.5: Proof of Theorem 4

Because all $\bar{G}^i$ have equal height 1, the approach used to prove Theorem 3 can also be used to prove Theorem 4. The details are left to the reader.

A.6: Proof of Theorem 5

Recall that $y_{Ll}(1)$ is computed as

$$y_{Ll}(1) = \min_{\forall f_i \in [\underline{f}^i, \overline{f}^i]} \frac{\sum_{i=1}^n a_{il}(1) f_i}{\sum_{i=1}^n f_i}. \tag{A19}$$

We consider two cases for (A19).

1. All $\tilde{G}^i$ are LSs, i.e., all $a_{il}(1) = 0$. Obviously, $y_{Ll}(1) = 0$ in this case. Similarly, we can show that $y_{Lr}(h_{\underline{Y}_{\mathrm{PR}}}) = 0$; hence, according to Lemma 2, $\tilde{Y}_{\mathrm{PR}}$ is an LS.

2. Not all $\tilde{G}^i$ are LSs, i.e., not all $a_{il}(1)$ are 0. In this case, $y_{Ll}(1)$ needs to be computed using a KM algorithm, and $\{a_{il}(1)\}$ need to first be sorted in ascending order. Assume K $(1 \le K < n)$ of the n $\tilde{G}^i$ are LSs. Because LSs have $a_{il}(1) = 0$, in the sorted $\{a_{il}(1)\}$

$$a_{il}(1) \begin{cases} = 0, & i = 1, \dots, K \\ > 0, & i = K+1, \dots, n. \end{cases} \tag{A20}$$

According to Part 1 of Lemma 1

$$y_{Ll}(1) \ge a_{1l}(1) = 0. \tag{A21}$$

According to Part 3 of Lemma 1, and also using the assumed fact that $\underline{f}^i = 0 \quad \forall i \ge K+1$

$$
\begin{aligned}
y_{Ll}(1) &= \min_{k \in [1, n-1]} \frac{\sum_{i=1}^k a_{il}(1)\overline{f}^i + \sum_{i=k+1}^n a_{il}(1)\underline{f}^i}{\sum_{i=1}^k \overline{f}^i + \sum_{i=k+1}^n \underline{f}^i} \\
&\le \frac{\sum_{i=1}^K a_{il}(1)\overline{f}^i + \sum_{i=K+1}^n a_{il}(1)\underline{f}^i}{\sum_{i=1}^K \overline{f}^i + \sum_{i=K+1}^n \underline{f}^i} \\
&= 0. \tag{A22}
\end{aligned}
$$

Equations (A21) and (A22) together demonstrate that $y_{Ll}(1) = 0$. Similarly, we can show that $y_{Lr}(h_{\underline{Y}_{\mathrm{PR}}}) = 0$. From Lemma 2, we know $\tilde{Y}_{\mathrm{PR}}$ is an LS.

In summary, $\tilde{Y}_{\mathrm{PR}}$ is an LS only when:

Case 1: All $\tilde{G}^i$ are LSs; or

Case 2: At least one $\tilde{G}^i$ is an LS, and the remaining $\tilde{G}^i$ have $\underline{f}^i = 0$.

Because Case 1 is included in Case 2, we only present Case 2 in the statement of Theorem 5.

A.7: Proof of Theorem 6

The proof of this theorem is so similar to the proof of Theorem 5 that it is left to the readers.

A.8: Proof of Theorem 7

The correctness of Theorem 7 is readily seen from Theorems 5 and 6, i.e., when either of Cases 1 and 2 is true, $\tilde{Y}_{\mathrm{PR}}$ is neither an LS nor a RS, and hence, it must be an interior FOU.

REFERENCES

[1] M. D. S. Brain, "On the relation between the natural logic of reasoning and standard logic," *Psychol. Rev.*, vol. 85, pp. 1–21, 1978.

[2] N. Chater, M. Oaksford, R. Nakisa, and M. Redington, "Fast, frugal and rational: How rational norms explain behavior," *Org. Behav. Hum. Decis. Process.*, vol. 90, no. 1, pp. 63–86, 2003.

[3] F. Herrera, E. Herrera-Viedma, and L. Martinez, "A fusion approach for managing multi-granularity linguistic term sets in decision making," *Fuzzy Sets Syst.*, vol. 114, pp. 43–58, 2000.

[4] B. Inhelder and J. Piaget, *The Growth of Logical Thinking From Childhood to Adolescence.* New York: Basic Books, 1958.

[5] J. Kacprzyk and R. R. Yager, "Linguistic summaries of data using fuzzy logic," *Int. J. Gen. Syst.*, vol. 30, pp. 33–154, 2001.

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

[7] G. J. Klir and B. Yuan, *Fuzzy Sets and Fuzzy Logic: Theory and Applications.* Upper-Saddle River, NJ: Prentice-Hall, 1995.

[8] J. Lawry, "A methodology for computing with words," *Int. J. Approx. Reason.*, vol. 28, pp. 51–89, 2001.

[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] F. Liu and J. M. Mendel, "An *interval approach* to fuzzistics for interval type-2 fuzzy sets," in *Proc. IEEE Int. Conf. Fuzzy Syst. (FUZZ-IEEE 2007)*, London, UK, pp. 1030–1035.

[11] F. Liu and J. M. Mendel, "Encoding words into interval type-2 fuzzy sets using an *Interval Approach*," *IEEE Trans. Fuzzy Syst.*, vol. 16, no. 6, Dec. 2008.

[12] F. Liu and J. M. Mendel, "Aggregation using the fuzzy weighted average, as computed by the Karnik–Mendel algorithms," *IEEE Trans. Fuzzy Syst.*, vol. 16, no. 1, pp. 1–12, Feb. 2008.

[13] M. Margaliot and G. Langholz, "Fuzzy control of a benchmark problem: A computing with words approach," *IEEE Trans. Fuzzy Syst.*, vol. 12, no. 2, pp. 230–235, Apr. 2004.

[14] J. M. Mendel, "Computing with words, when words can mean different things to different people," in *Proc. 3rd Int. ICSC Symp. Fuzzy Logic Appl.*, Rochester, NY, Jun. 1999, pp. 158–164.

[15] J. M. Mendel, "The perceptual computer: An architecture for computing with words," in *Proc. IEEE Int. Conf. Fuzzy Syst. (FUZZ-IEEE 2001)*, Melbourne, Australia, Dec. 2001, pp. 35–38.

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

[17] J. M. Mendel, "An architecture for making judgments using computing with words," *Int. J. Appl. Math. Comput. Sci.*, vol. 12, no. 3, pp. 325–335, 2002.

[18] J. M. Mendel, "Computing with words and its relationships with fuzzistics," *Inf. Sci.*, vol. 177, pp. 988–1006, 2007.

[19] J. M. Mendel, "Type-2 fuzzy sets and systems: An overview," *IEEE Comput. Intell. Mag.*, vol. 2, no. 2, pp. 20–29, Feb. 2007.

[20] J. M. Mendel, R. I. John, and F. Liu, "Interval type-2 fuzzy logic systems made simple," *IEEE Trans. Fuzzy Syst.*, vol. 14, no. 6, pp. 808–821, Dec. 2006.

[21] J. M. Mendel and D. Wu, "Perceptual reasoning: A new computing with words engine," in *Proc. IEEE Granular Comput. Conf.*, San Jose, CA, Nov. 2007, pp. 446–451.

[22] A. Niewiadomski, J. Kacprzyk, J. Ochelska, and P. S. Szczepaniak, "Interval-valued linguistic summaries of databases," *Control Cybern.*, vol. 35, no. 2, pp. 415–444, 2006.

[23] L. J. Rips, *The Psychology of Proof.* Cambridge, MA: MIT Press, 1994.

[24] S. H. Rubin, "Computing with words," *IEEE Trans. Syst., Man, Cybern. B, Cybern*, vol. 29, no. 4, pp. 518–524, Aug. 1999.

[25] I. B. Türksen, "Interval-valued fuzzy sets based on normal forms," *Fuzzy Sets Syst.*, vol. 20, pp. 191–210, 1986.

[26] I. B. Türksen, "Interval valued fuzzy sets and fuzzy connectives," *J. Interval Comput.*, vol. 4, pp. 125–142, 1993.

[27] I. B. Türksen, "Type-2 representation and reasoning for CWW," *Fuzzy Sets Syst.*, vol. 127, pp. 17–36, 2002.

[28] I. B. Türksen and D. D. W. Yao, "Representation of connectives in fuzzy reasoning: The view through normal forms," *IEEE Trans. Syst., Man, Cybern.*, vol. 14, no. 1, pp. 146–150, 1984.

[29] H. Wang and D. Qiu, "Computing with words via Turing machines: A formal approach," *IEEE Trans. Fuzzy Syst.*, vol. 11, no. 6, pp. 742–753, Dec. 2003.

[30] J. H. Wang and J. Hao, "A new version of 2-tuple fuzzy linguistic representation model for computing with words," *IEEE Trans. Fuzzy Syst.*, vol. 14, no. 3, pp. 435–445, Jun. 2006.

[31] D. Wu and J. M. Mendel, "The linguistic weighted average," in *Proc. IEEE Int. Conf. Fuzzy Syst. (FUZZ-IEEE 2006)*, Vancouver, CA, pp. 3030–3037.

[32] D. Wu and J. M. Mendel, "Aggregation using the linguistic weighted average and interval type-2 fuzzy sets," *IEEE Trans. Fuzzy Syst.*, vol. 15, no. 6, pp. 1145–1161, Dec. 2007.

[33] D. Wu and J. M. Mendel, "Corrections to 'aggregation using the linguistic weighted average and interval type-2 fuzzy sets'," *IEEE Trans. Fuzzy Syst.*, vol. 16, no. 6, Dec. 2008.

[34] D. Wu and J. M. Mendel, "A vector similarity measure for linguistic approximation: Interval type-2 and type-1 fuzzy sets," *Inf. Sci.*, vol. 178, pp. 381–402, 2008.

[35] D. Wu and J. M. Mendel, "Enhanced Karnik–Mendel algorithms," *IEEE Trans. Fuzzy Syst.*, to be published.

[36] H. Wu and J. M. Mendel, "On choosing models for linguistic connector words for Mamdani fuzzy logic systems," *IEEE Trans. Fuzzy Syst.*, vol. 12, no. 1, pp. 29–44, Feb. 2004.

[37] H. Wu and J. M. Mendel, "Antecedent connector word models for interval type-2 fuzzy logic system," in *Proc. IEEE Int. Conf. Fuzzy Syst. (FUZZ-IEEE 2004)*, Budapest, Hungary, pp. 1099–1104.

[38] R. R. Yager, "A new approach to the summarization of data," *Inf. Sci.*, vol. 28, pp. 69–86, 1982.

[39] R. R. Yager, "On the retranslation process in Zadeh's paradigm of computing with words," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 2, pp. 1184–1195, Apr. 2004.

[40] L. A. Zadeh, "Outline of a new approach to the analysis of complex systems and decision processes," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-3, no. 1, pp. 28–44, Jan. 1973.

[41] L. A. Zadeh, "Fuzzy logic = computing with words," *IEEE Trans. Fuzzy Syst.*, vol. 4, no. 2, pp. 103–111, May 1996.

[42] L. A. Zadeh, "From computing with numbers to computing with words—from manipulation of measurements to manipulation of perceptions," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 4, no. 1, pp. 105–119, Jan. 1999.

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

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

Prof. Mendel is a Distinguished Member of the IEEE Control Systems Society. He was the President of the IEEE Control Systems Society in 1986, and is currently the Chairman of the Fuzzy Systems Technical Committee and an Elected Member of the Administrative Committee of the IEEE Computational Intelligence Society. He was the recipient of numerous awards, including the 1983 Best Transactions Paper Award of the IEEE Geoscience and Remote Sensing Society, the 1992 Signal Processing Society Paper Award, the 2002 IEEE TRANSACTIONS ON FUZZY SYSTEMS Outstanding Paper Award, a 1984 IEEE Centennial Medal, an IEEE Third Millenium Medal, a Pioneer Award from the IEEE Granular Computing Conference, May 2006, and the 2008 Fuzzy Systems Pioneer Award from the IEEE Computational Intelligence Society.

Dongrui Wu (S'05) received the B.E. degree in automatic control from the University of Science and Technology of China, Hefei, China, in 2003, and the M.Eng. degree in electrical engineering from National University of Singapore, Singapore, in 2005. Currently, he is working toward the Ph.D. degree in electrical engineering at the University of Southern California, Los Angeles.

His current research interests include control theory and applications, robotics, optimization, pattern classification, information fusion, computing with words, and computational intelligence and their applications to smart oilfield technologies.

Mr. Wu was the recipient of the Best Student Paper Award from the IEEE International Conference on Fuzzy Systems, Reno, Nevada, 2005.