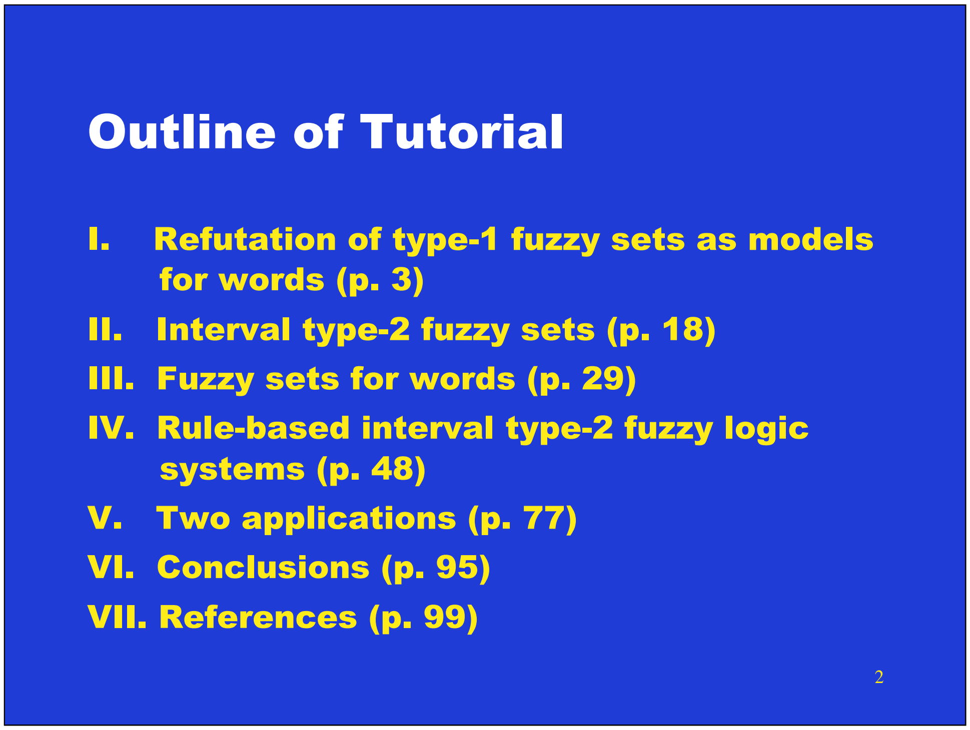
Fuzzy Sets for Words: Why Type-2 Fuzzy Sets Should be Used and How They Can be Used

Jerry M. Mendel

University of Southern California

Los Angeles, CA 90089-2564

Note: This was presented as a two-hour tutorial at IEEE FUZZ in Budapest Hungary in 2004. I have made some modifications to it, and have added notes to most of the slides so that it now serves as a more useful on-line learning experience.



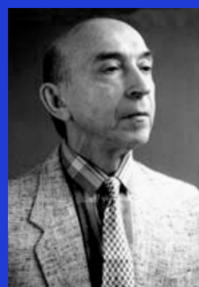
Each of the listed major sections of this tutorial will be preceded by a title page. The page on which a new section begins is shown in parentheses after the title of that section.

Part I: Refutation of type-1 fuzzy sets as models for words

3

Some people may find this section very controversial. I will make some further comments about this later.

Fuzzy Sets



- In 1965, Lotfi Zadeh published the first paper about fuzzy sets
- Opened up a new and important field that has withstood the tests of time (>30,000 articles)

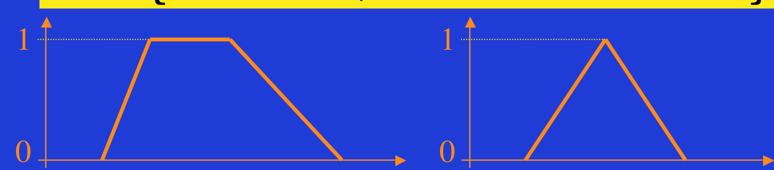
4

If I have nothing to add to what is on the slide, the accompanying notes page will be blank.

Fuzzy Set Defined

- **Unlike a crisp set whose MF takes on only two values 0 or 1, the MF of a fuzzy set can range between 0 and 1**

$$A = \left\{ \left(x, \mu_A(x) \right) \middle| x \in X, 0 \le \mu_A(x) \le 1 \right\}$$



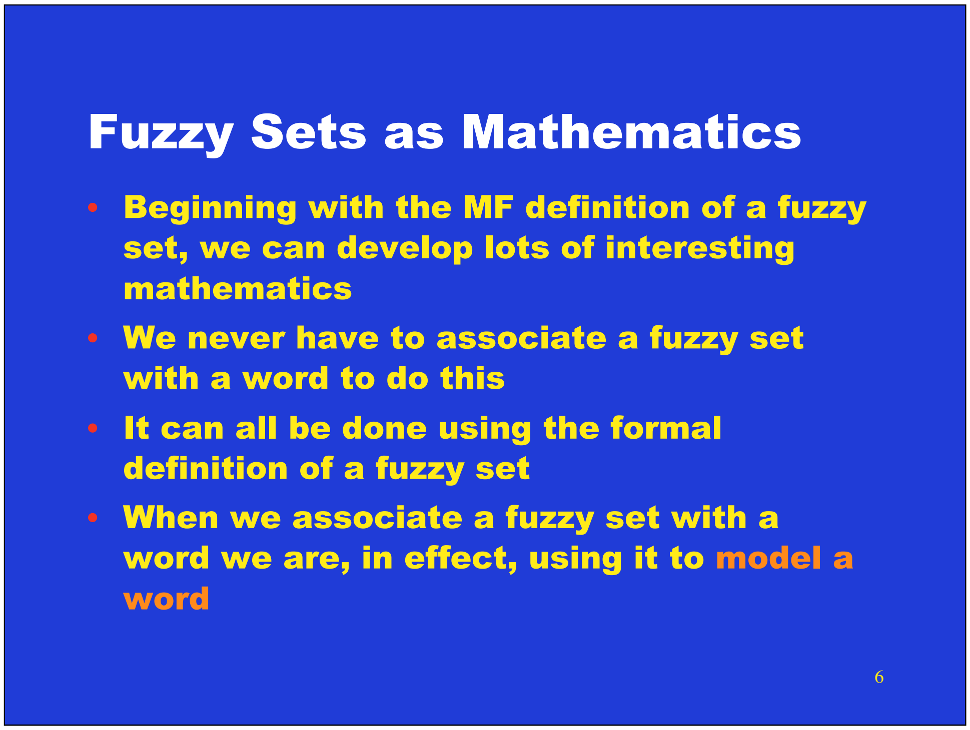
MF: Membership function

The equation for *A* defines the fuzzy set by means of its MF.

Shown on this slide are two well-known examples of a MF, a trapezoid and a triangle.

It is now becoming popular to refer to the fuzzy set that is defined on this slide as a "type-1 fuzzy set," because people are also working with "type-2 fuzzy sets."

We will have a lot more to say about type-2 fuzzy sets later in this tutorial.



Distinguishing between mathematics and applications of it as models of different kinds of physical situations is not a new idea.

Consider, for example, differential equations, which can be studied entirely within the framework of mathematics, or as models for physical systems.

The study of differential equations within the framework of mathematics is well-established. It covers issues such as existence and uniqueness of solution, solutions, etc.

It is through the use of scientifically accepted principles (e.g. Newton's Laws, Kirchoff's Laws) that we are led to differential equation models for mechanical, electrical, etc. systems.

Fuzzy Sets Used to Model Words

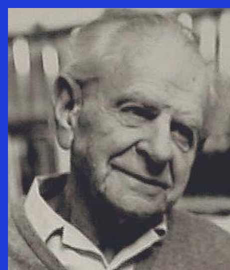
Fuzzy sets used to model words can be interpreted as a *scientific theory*, as distinct from fuzzy sets as mathematics

- **Is the use of fuzzy sets to model words *scientific*?**
- **If so, is it a *correct* scientific theory?**

7

When fuzzy sets are used in situations where they are not modeling words, then these questions are not usually (have not been) asked.

Falsificationism



Sir Karl Popper
(1902-1994)

A theory is scientific only if it is refutable by a conceivable event. Every genuine* test *of a scientific theory, then, is logically an attempt to refute or to falsify it, and one genuine counter instance falsifies the whole theory.

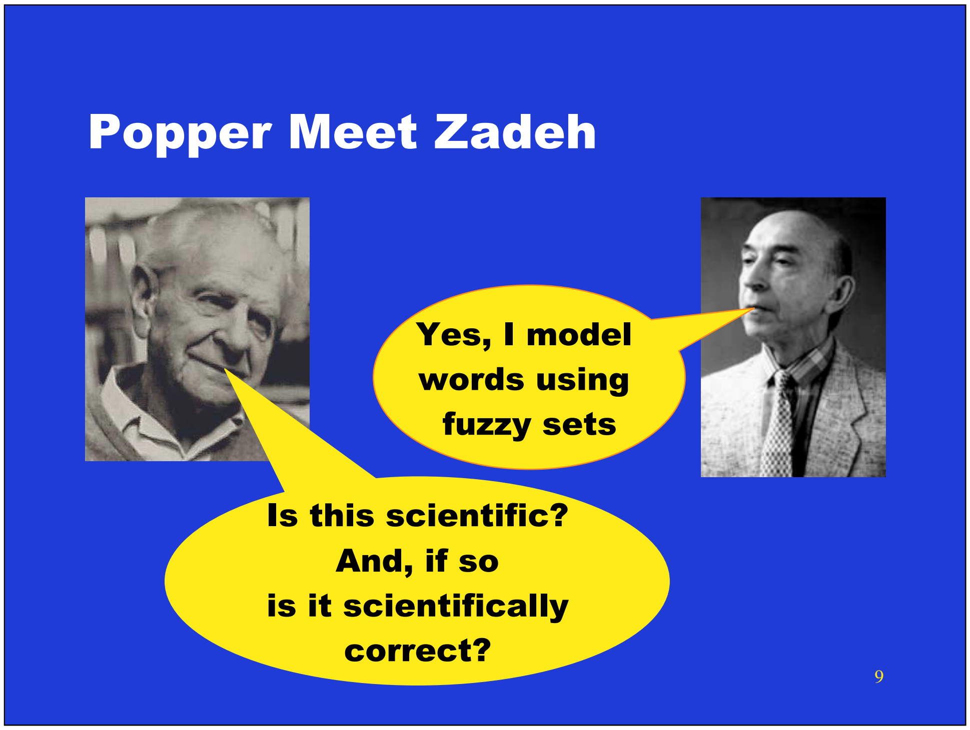
- Ref.: Karl Popper, *The Logic of Scientific Discovery*, Hutchinson, London, 1959

8

According to the *Stanford Encyclopedia of Philosophy*, "Karl Popper is generally regarded as one of the greatest philosophers of science of this (20th) century." For information about him, go to:

http://plato.stanford.edu/entries/popper

On the Karp Popper Web (http://www.eeng.dcu.ie/~tkpw), it is stated that: "Falsificationism is the idea that science advances by unjustified exaggerated guesses followed by unstinting criticism. Only hypotheses capable of clashing with observation reports are allowed to count as scientific."

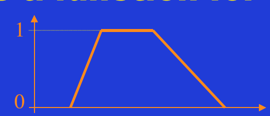


I will now use Popper's Falsificationism to answer the question "Is it scientifically correct to model words using fuzzy sets?"

But first I will ask the question "Who among us has not used word examples for fuzzy sets?

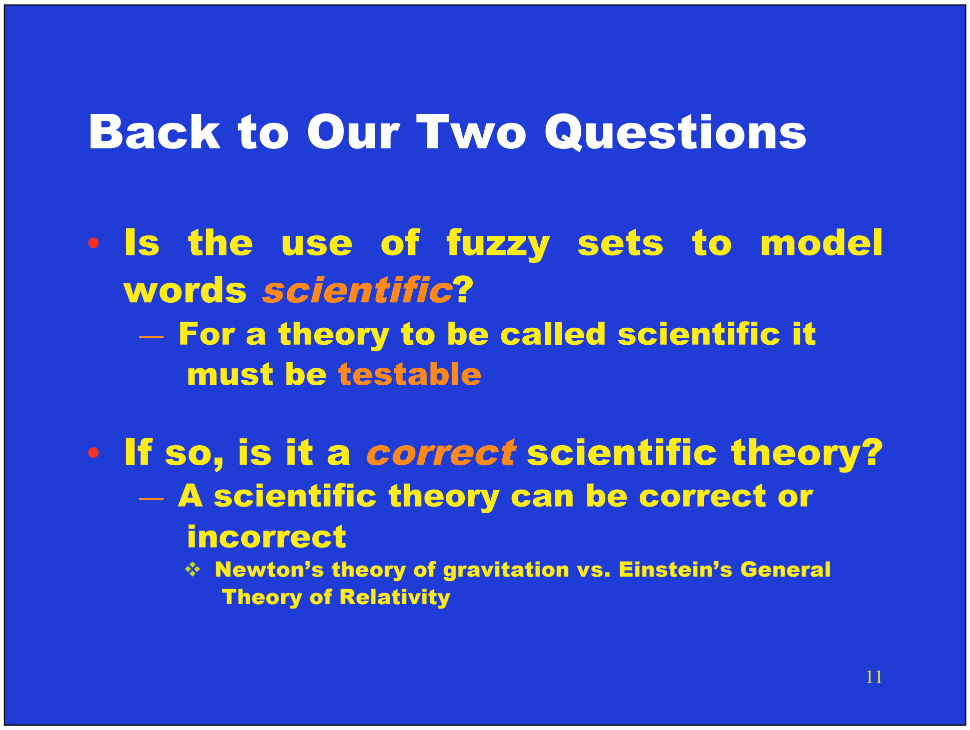
Who Among Us Has Not Used Word Examples For Fuzzy Sets?

- In his 1965 paper, Zadeh's first example of a fuzzy set connected it to a word
- We all do this!
- We don't have to do it!
- MF pictures can be drawn without attaching a word label to the fuzzy set
- Just choose a function for the MF and then sketch it



10

This is not meant to be an attack on fuzzy sets. Some people may have misinterpreted it as such. It may be the first time someone has questioned Zadeh's use of fuzzy sets as models for words.



Some theories are not scientific because they are not testable, e.g., astrology, phrenology.

However, a theory that is scientific can be correct or incorrect.

Until a "correct" scientific theory is refuted it remains accepted as a correct scientific theory. Once it has been refuted, it is replaced by the new "correct" scientific theory, which then waits to be refuted.

Fuzzy Sets for Words *Are Scientific*: 1

- Methods have been described in the FS literature to elicit knowledge from people for the construction of a MF
  - Polling
  - Interval estimation
  - Direct rating
  - Reverse rating
  - Transition interval estimation

12

Most engineering applications of fuzzy sets do not construct a MF by surveying people and using any of these knowledge elicitation methods.

Going from knowledge elicited from people to a MF is an "inverse problem."

In most engineering applications of fuzzy sets we choose the shape of the MF and either pre-specify its parameters or we "learn" its parameters through an optimization procedure that uses application-based data (e.g., a temperature profile), but not elicited MF data.

A reference for some of these knowledge elicitation methods is: I. B. Türksen, "Measurement of Membership Functions and Their Acquisition," *Fuzzy Sets and Systems*, vol. 40, pp. 5-38, 1991 (especially Section 5).

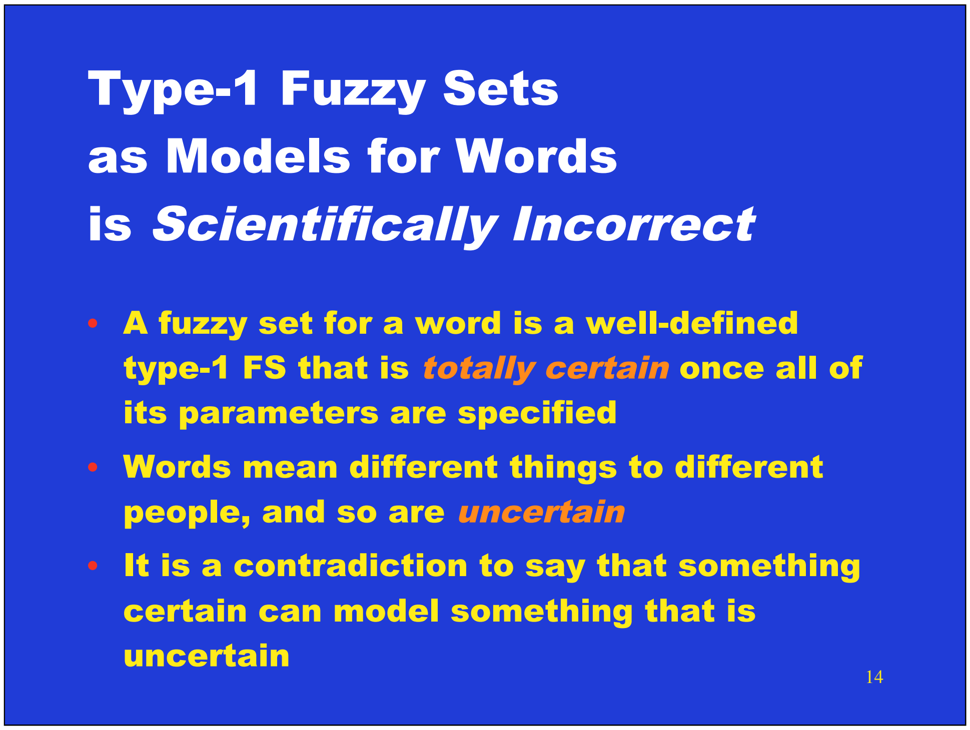
Fuzzy Sets for Words *Are Scientific*: 2

- **Because data can be obtained from people, this permits us to test fuzzy sets as models for words, i.e to attempt to refute or falsify the scientific theory of fuzzy sets as models for words**

13

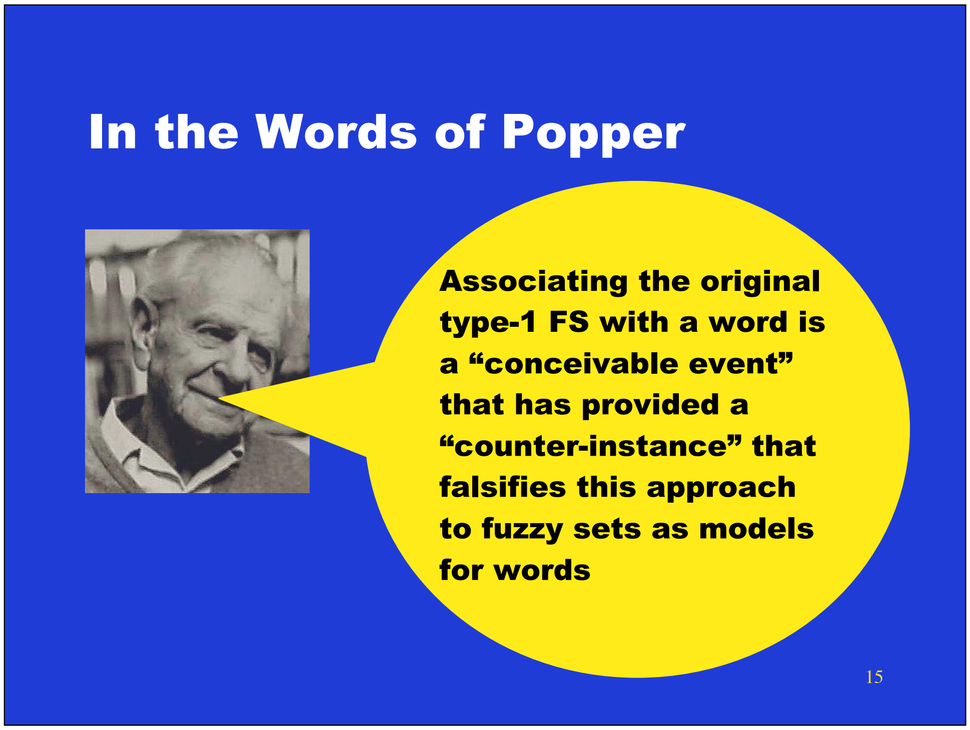
So, the previous work on how to collect data from people and then use it to obtain type-1 MFs makes Zadeh's "Type-1 fuzzy sets as a model for words" scientific.

But, remember a scientific theory can be correct or incorrect.



This slide contains a simple three-step refutation of type-1 fuzzy sets as a model for words.

Again, some may find this very controversial. It does not refute fuzzy sets as mathematics or its use in most applications (e.g., fuzzy logic control, rule-based classification, etc.). It has only refuted type-1 fuzzy sets as models for words, and therefore calls their use into question in the application of "computing with words."

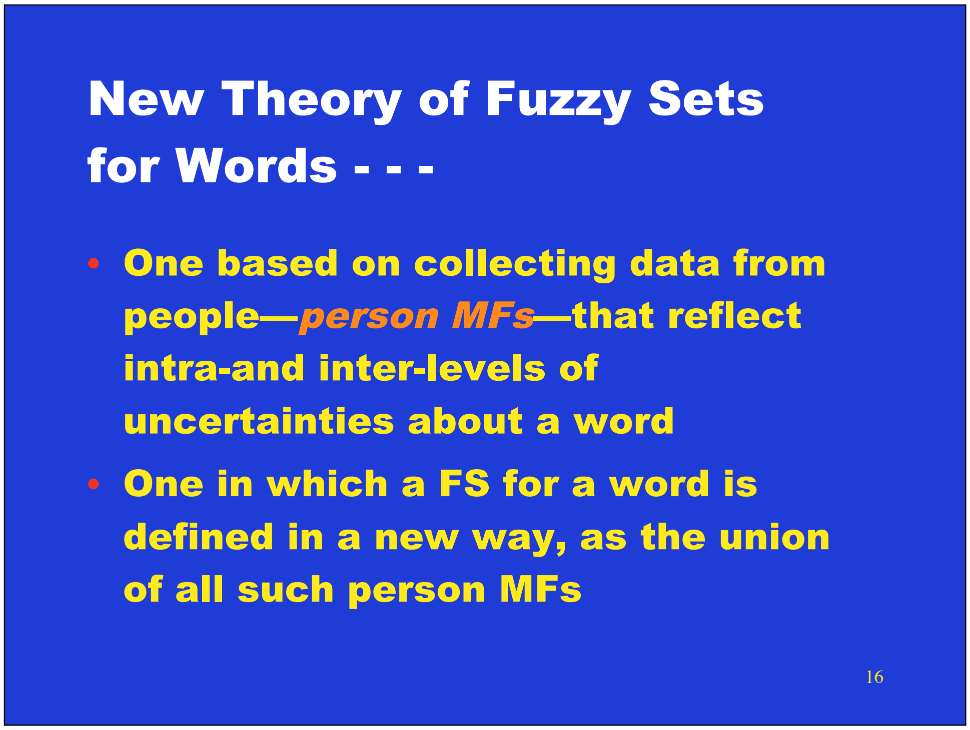


We needed a basis for exploring whether or not type-1 fuzzy sets as a model for words is or is not scientifically correct. Popper's widely accepted Falsificationism provided us with such a basis.

Zadeh has suggested that type-1 fuzzy sets can be used as a model for a "prototypical word." Such a model does not capture the uncertainties in a word; however, if a study does not require such uncertainties to be used (e.g., propagated through a rule-based system) then one may be able to use a type-1 fuzzy set model for a word.

This is analogous to determinism versus randomness. If one chooses to ignore the presence of random uncertainties, he/she can work within the framework of determinism. Whether or not this is correct, depends upon the objectives of a specific study.

When a system that is designed using fuzzy logic is going to interact directly with people using words, then it is important to use a fuzzy set model for the words that captures the fact that words mean different things to different people. A type-1 fuzzy set model for a word cannot do this.

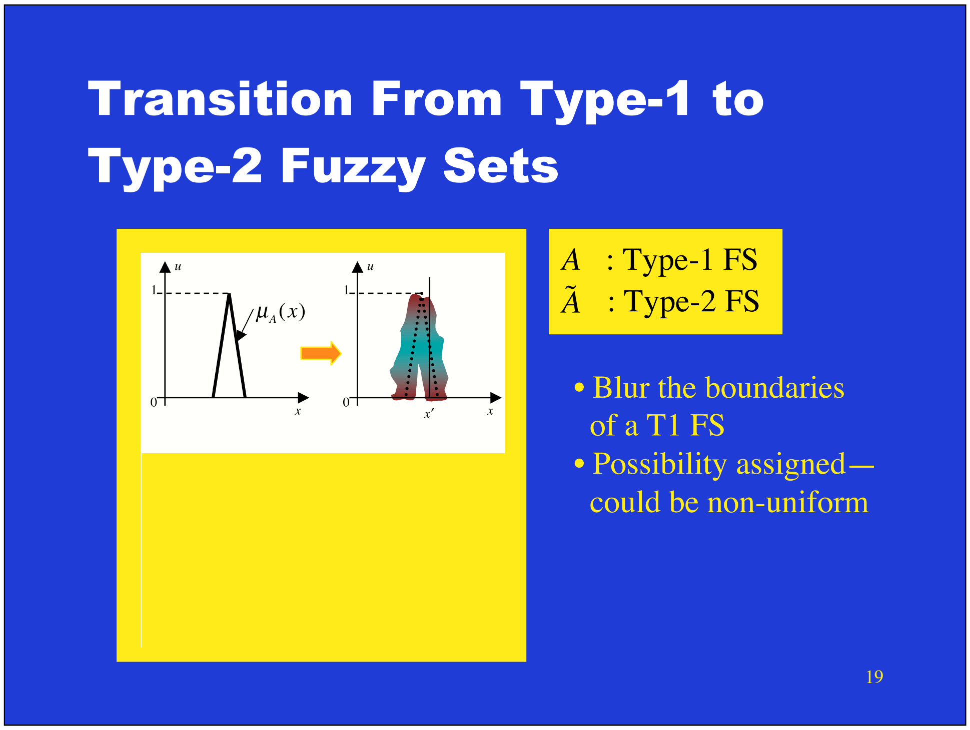


So, we need a new theory of fuzzy sets for words, one that is testable and may ultimately be refuted.

Pause

- The new theory is in terms of interval type-2 fuzzy sets
- We must, therefore, pause to introduce material about such fuzzy sets
- Although a more general theory of T2 FS's exists, we shall focus exclusively on interval T2 FSs

Part II. Interval Type-2 Fuzzy Sets



The following short article introduces the reader to type-2 fuzzy sets through a series of questions and answers: J. M. Mendel, "Type-2 Fuzzy Sets: Some Questions and Answers," *IEEE Connections*, *Newsletter of the IEEE Neural Networks Society*, vol. 1, Aug. 2003, pp. 10-13. It can be accessed at:

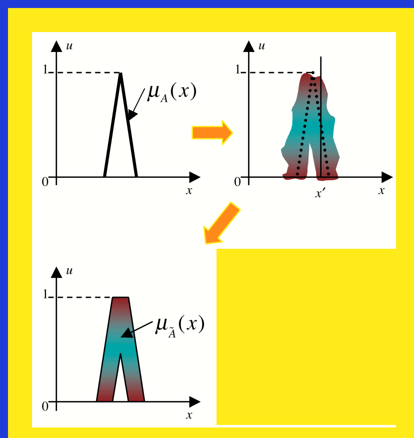
http://sipi.usc.edu/~mendel

In order to distinguish a type-2 fuzzy set from a type-1 fuzzy set we use a tilde symbol over the capital letter.

Blurring of the boundaries can be thought of as moving the triangle MF to the left and right in a non-uniform manner.

The different colors in the right-hand figure on this slide denote non-uniform possibilities, whose amplitudes would be coming out of the page in a third-dimension.

Transition From Type-1 to Type-2 Fuzzy Sets

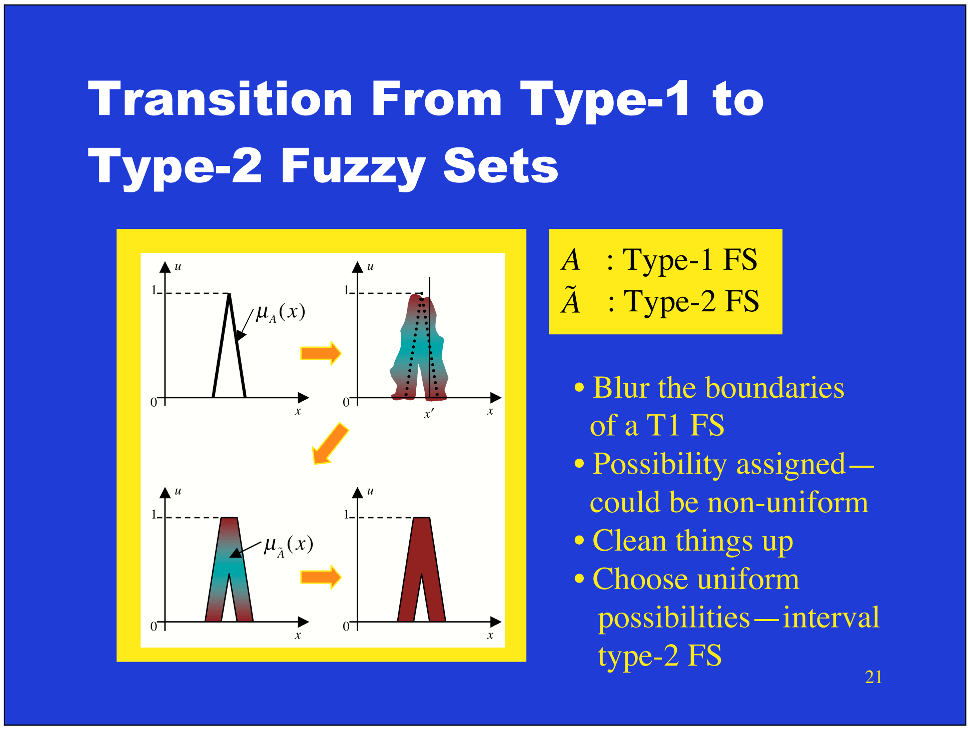


A : Type-1 FS
$\tilde{A}$: Type-2 FS

- Blur the boundaries of a T1 FS
- Possibility assigned—could be non-uniform
- Clean things up

20

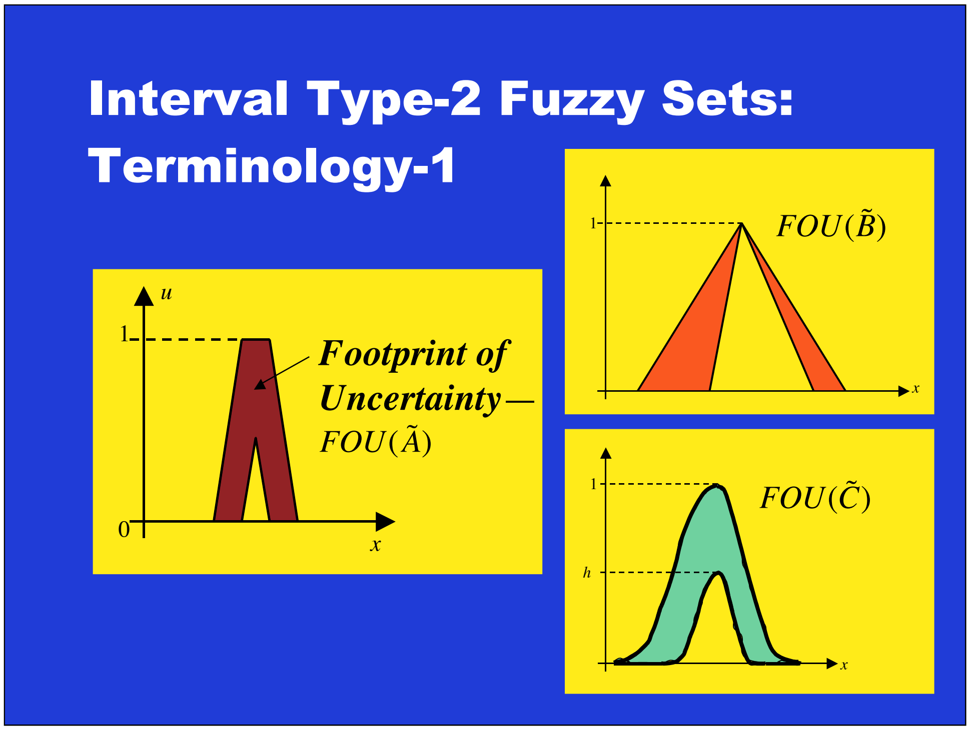
"Clean things up" means using well-defined geometric shapes for the upper and lower bounds of the blurred MF. Here we use a trapezoid to clean up the upper bound of the blurred MF, and a triangle to clean up the lower bound of the blurred MF.



As of this date (2005) it is only practical to work with interval type-2 fuzzy sets, because computations for general type-2 fuzzy sets are horrendous whereas they are simple for interval type-2 fuzzy sets.

In the future, it may be possible to use more general type-2 fuzzy sets.

See Mendel (2001) for a comprehensive treatment of general and interval type-2 fuzzy sets.



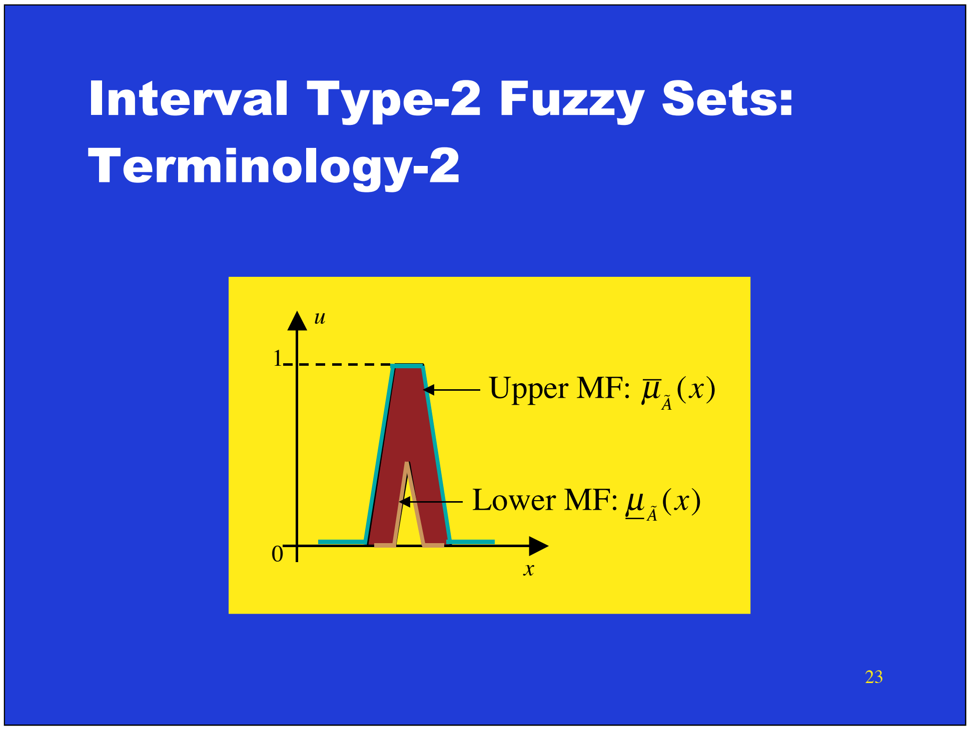
Yes, type-2 fuzzy sets requires some new terminology, just as probability (versus determinism) does.

It is not the purpose of this tutorial to provide mathematical definitions of new terms. They can be found in Mendel (2001).

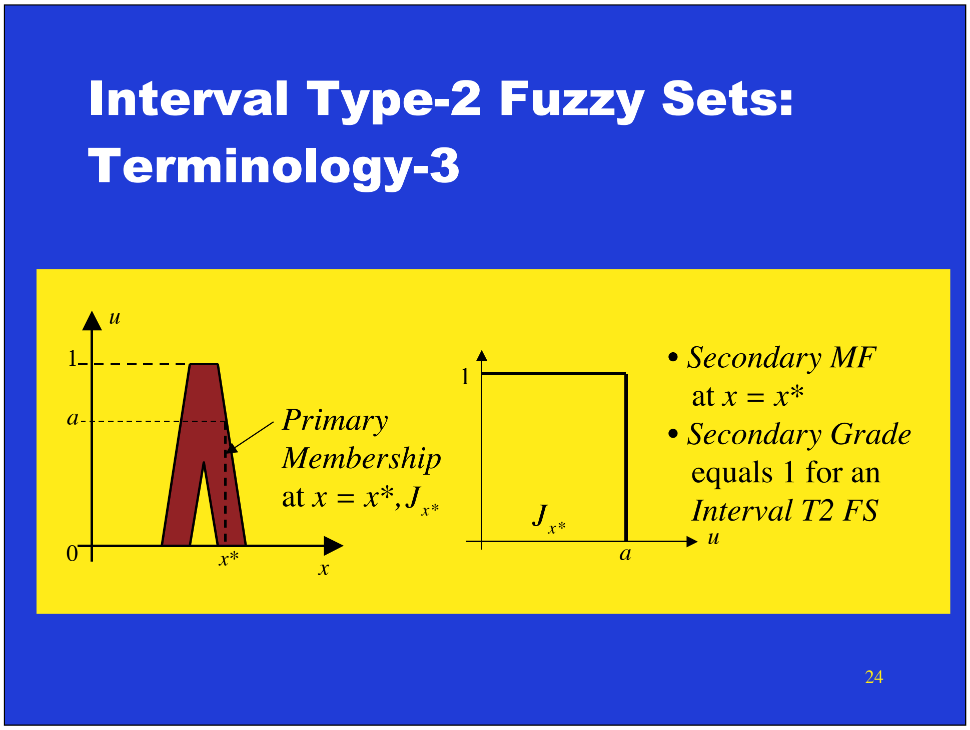
FOU: Footprint of uncertainty

In the above figures, the FOU is the shaded region. These regions can have different geometric shapes (which is why there are three figures shown).

The more (less) area in the FOU the more (less) is the uncertainty.



Each FOU has an upper and a lower bound (MF), and, as we will see, it is these bounds that are used in the mathematics of interval type-2 fuzzy sets.



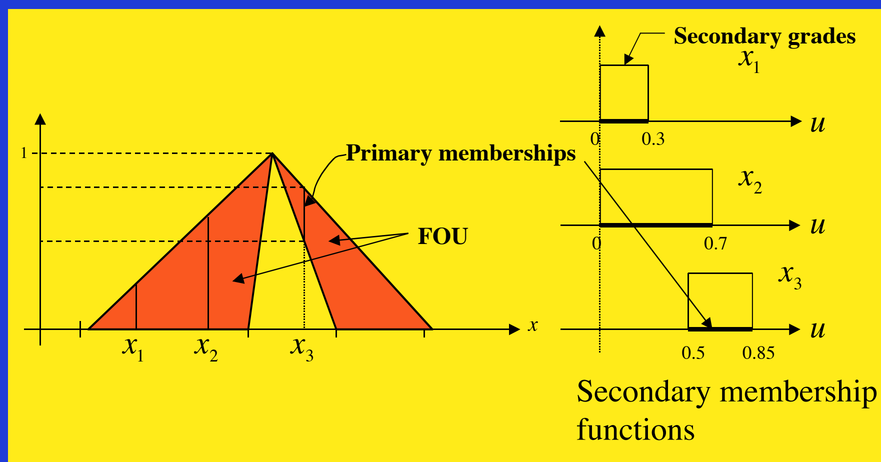
At each value of the primary variable, x, its MF is no longer a point value. It is a function whose domain is called the "primary membership."

The distribution that sits on top of the primary membership is called a "secondary MF."

The amplitude of the secondary MF is called the "secondary grade."

For an interval type-2 fuzzy set, the secondary grade equals 1 over the entire FOU; hence, the new third dimension of a type-2 fuzzy set does not convey any new information for an interval type-2 fuzzy set. It is the FOU that completely characterizes an interval type-2 fuzzy set.

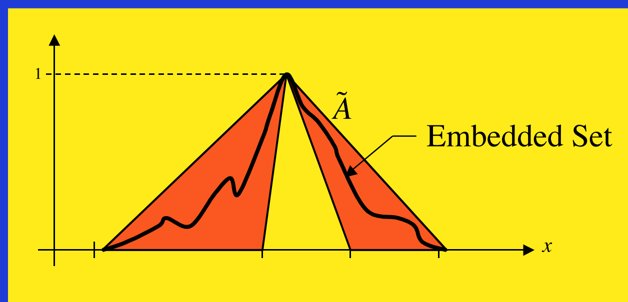
Interval Type-2 Fuzzy Sets: Terminology-4



This example illustrates much of the new terminology.

It shows the FOU of an interval type-2 FS and the secondary MFs at three values of the primary variable.

Interval Type-2 Fuzzy Sets: Terminology-5



- Embedded T2 FS
- Embedded T1 FS (the domain of an embedded T2 FS)

Theorem: $\tilde{A}$ **equals the union of all its embedded T2 FSs**

This theorem, which is called a "Representation Theorem," was first proved for a general type-2 fuzzy set in: J. M. Mendel and R. I. John, "Type-2 Fuzzy Sets Made Simple," *IEEE Trans. on Fuzzy Systems*, vol. 10, pp. 117-127, April 2002.

FS: Fuzzy set

Embedded FSs are also called "wavy-slices."

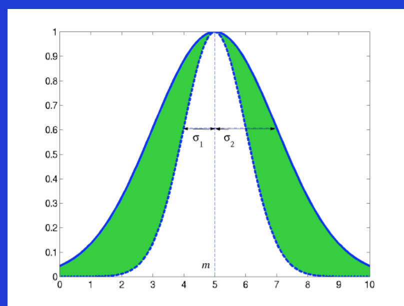
The wavy slice shown above has all of its secondary grades equal to 1 (they are not shown).

Imagine constructing all possible wavy slice. They would completely cover the FOU, so that their union would be the FOU.

Embedded FSs are useful in theoretical developments, but, because there are so many of them for a FOU, they are not used in computation.

Interval Type-2 Fuzzy Sets: Terminology-6

• ***Primary MF***
A T1 MF that is parameterized. When the parameters vary over some intervals we obtain a FOU



• Gaussian primary MF: uncertain standard deviation

$$\mu_{\tilde{A}}(x) = \exp\left[-\tfrac{1}{2}\left(\frac{x-m}{\sigma}\right)^2\right]$$

$$\sigma \in [\sigma_1, \sigma_2]$$

27

Here is an example of starting with a type-1 MF and incorporating some uncertainty about one of its parameters, its standard deviation, in constructing a FOU.

The underlying type-1 MF is called a "primary MF."

T1: type-1

Interval Type-2 Fuzzy Sets: Some Notation

- ***Point-Representation of a T2 FS***

$$\tilde{A} = \left\{ \left((x,u), \mu_{\tilde{A}}(x,u) = 1 \right) \middle| \forall x \in X, \forall u \in J_x \subseteq [0,1] \right\}$$

- ***Secondary MF: a Vertical Slice***

$$\mu_{\tilde{A}}(x') = \int_{u \in J_{x'}} 1/u = \left[\underline{\mu}_{\tilde{A}}(x'), \bar{\mu}_{\tilde{A}}(x') \right] \quad x' \in X$$

- ***Vertical-Slice Representation of a T2 FS***

$$\tilde{A} = \int_{x \in X} \mu_{\tilde{A}}(x)/x$$

28

Okay! I had to include some equations :).

T2: type-2

Visualize a FOU (e.g., see the FOU on slide 25), and draw a vertical line at any value of the primary variable, x. This is a "vertical slice."

Each vertical slice is an interval set, whose end-points are the lower and upper MF values, obtained where the vertical line intersects the FOU.

Unlike the wavy slice representation for an interval type-2 fuzzy set, that is not useful for computation, the vertical slice representation has been very useful for computation.

Part III. Fuzzy Sets For Words

29

Return to Fuzzy Sets for Words

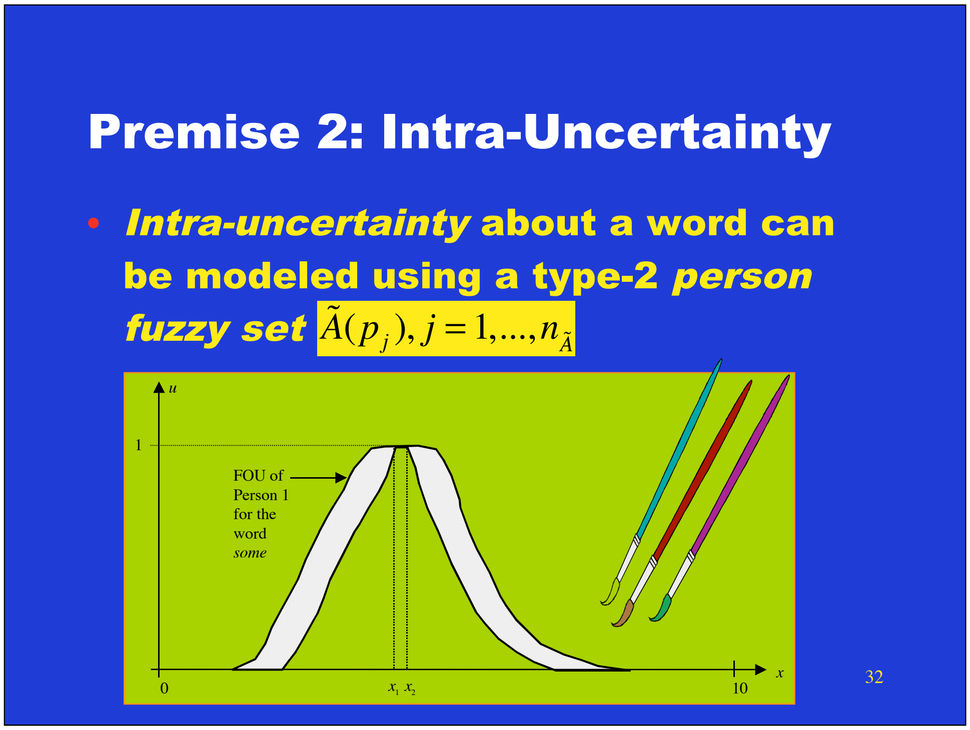
- With this short background into interval type-2 fuzzy sets, we can return to fuzzy sets for words
- Recall, we had demonstrated that to use a type-1 FS for a word is scientifically incorrect
- Our new fuzzy set model for a word is built upon a few premises

30

Premise 1: Uncertainty

- Words mean different things to different people, and are therefore uncertain. Uncertainty about a word is of two kinds:
  - Intra-uncertainty, which is the uncertainty that a person has about the word
  - Inter-uncertainty, which is the uncertainty that a group of people have about the word

31



The paint brushes are meant to conceptually denote that each person is only able to provide a "broad-brush" FOU for a word, where the breadth of a brush is associated with how much uncertainty the person has for a specific word.

A "thin" brush would conceptually be associated with a word for which a person only has a small amount of uncertainty. A "fat" brush would conceptually be associated with a word for which a person has a large amount of uncertainty.

The amount of uncertainty a person has can vary as a function of the primary variable. In the FOU shown above, there is a small range of values where a person has no uncertainty about the word. This is okay.

Comments

- In principal, person MFs could be collected from a group of people
- In practice, this may be very difficult to do
  - People would have to know what a "MF" means
  - How to collect such data has yet to be established
- All we need to know, at this point, is that in principal person MFs could be obtained

33

Only very knowledgeable people would be able to provide person MFs, because to do so that person must understand the concept of a MF.

Most people have no idea what a MF is.

For such people another way of collecting data about words is needed, and then an automatic way is needed to go from a collection of such data to a FOU.

For additional discussions about this, see: J. M. Mendel and H. Wu, "Type-2 Fuzzistics for Symmetric Interval Type-2 Fuzzy Sets: Part 1, Forward Problems, and Part 2, Inverse Problems," *IEEE Trans. on Fuzzy Systems*, 2005.

The term "fuzzistics" has been coined for how to go from uncertain data about words that are elicited from people to parametric interval type-2 fuzzy set models for the words. It is a blending of the words, fuzzy and statistics.

Premise 3

- **Assume all secondary grades of person MF data equal 1**
  - **First-order versus second-order uncertainties**
- **Focus only on first-order uncertainties**
- **Terms FOU and MF can be used interchangeably**

34

When we use type-1 fuzzy sets, we do not know the "optimal" shape and parameters for the MFs. Applications that use fuzzy sets seem to be very robust to the actual choice made for the MFs.

So, if we cannot be certain what the best shape is for the MF of a type-1 FS (the primary MF), then is seems unrealistic to me that we should think about using anything other than a uniform set of secondary grades for a type-2 fuzzy set.

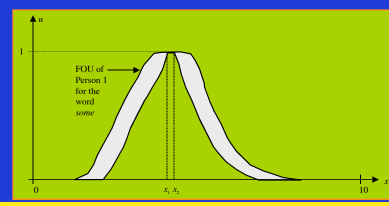
In this way the FOU may be said to capture "first-order uncertainties," whereas the FOU and non-uniform secondary grades capture first- and second-order uncertainties.

We have already explained why the FOU provides a complete description for an interval type-2 FS. Consequently, we can use "FOU" and "MF" interchangeably for interval type-2 FSs.

Mathematical Description of a Type-2 Person Fuzzy Set

Person fuzzy set

$$\tilde{A}(p_j) = \int_{x \in X} \mu_A(x \mid p_j) \big/ x$$



Type-2 Person MF

$$\mu_A(x \mid p_j) = \left[a_{\tilde{A}}(x \mid p_j), b_{\tilde{A}}(x \mid p_j) \right] \subseteq [0,1]$$

Lower and upper bounds of Type-2 Person MF

$a_{\tilde{A}}(x \mid p_j)$ and $b_{\tilde{A}}(x \mid p_j)$

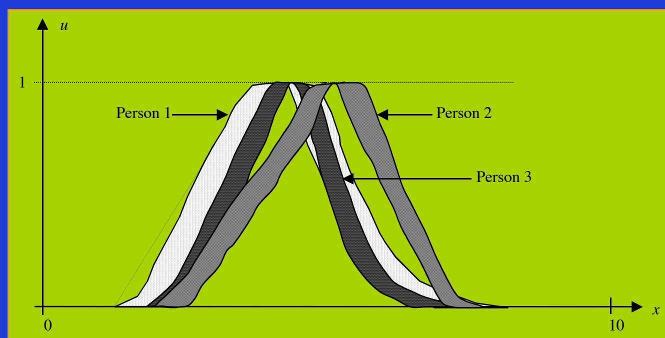
35

Note the conditioning of all quantities on the j th person. We need this extra notation so that we can distinguish FOUs from person-to-person.

This notation also lets us easily add (or remove) more person MFs to (or from) our database of person MFs.

Intra-Uncertainty (Cont’d.)

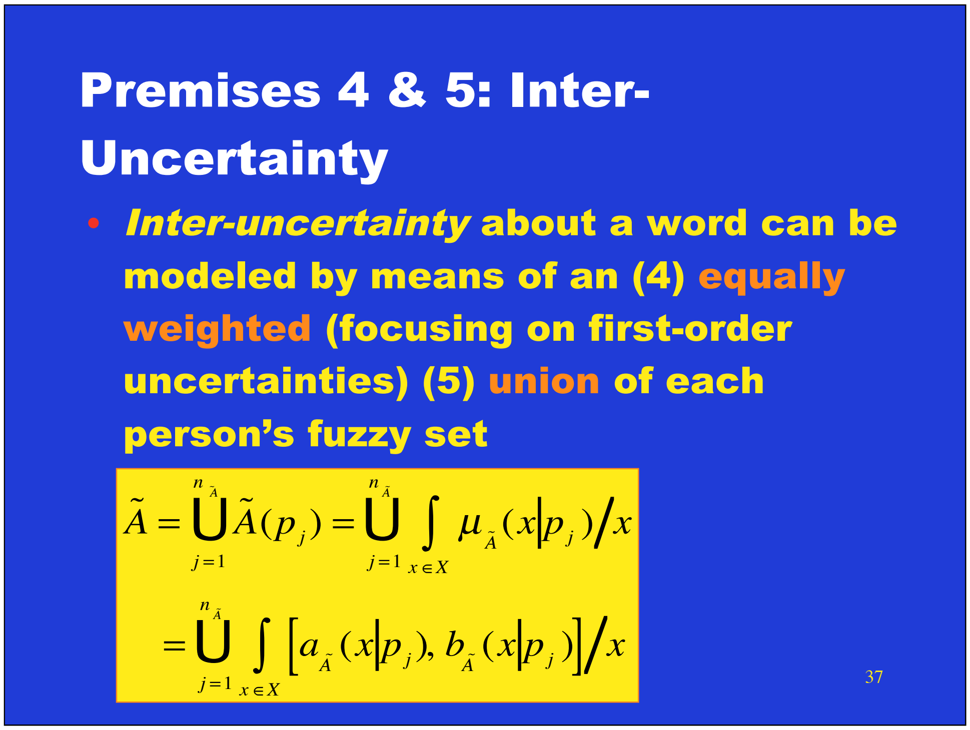
- **Collect person MFs from a group of people**



36

Again, because words mean different things to different people, and we most likely will be interacting with more than one person, we need to collect person MFs from a representative group of the people we will be interacting with.

A specific application may only involve teenage girls, beer-drinking men, naturalists, bikers, etc. It is important then to collect person MFs from members of the representative group.



Objections to using an equal weighting and the union are listed on the next slide.

Regardless, what is finally needed is a "fusion" of person MFs across the complete set of available person MFs.

Such a "fusion" or "aggregation" must capture both the intra-and inter-uncertainties about a word.

Comments

- **Objections could be raised about:**
  - **Equal weighting of person MFs**
  - **Unioning person MFs**
- **Regardless of the weighting strategy or method for combining, we will arrive at some composite—aggregated—new FS model for a word**

38

Equal weighting of person MFs: Maybe some people are more knowledgeable about a word than are others. But who decides this? And, who decides on how to weight each person? Such a person-specific weighting introduces another level of uncertainty.

Unioning person MFs: The union operation preserves the upper and lower bounds associated with all person MFs, which seems to me to be a good thing. Of course, just as in statistics, where outliers must be established and eliminated, we need to to a similar thing for "outlier person MFs." Let's assume that this has been done prior to the unioning of the remaining person MFs.

Parsimony

- **Because the new person-MF representation of a word preserves all of the MF data that is collected from people it is not a parsimonious model for word fuzzy sets**
- **Approximate the data using parametric model to obtain a parsimonious model**

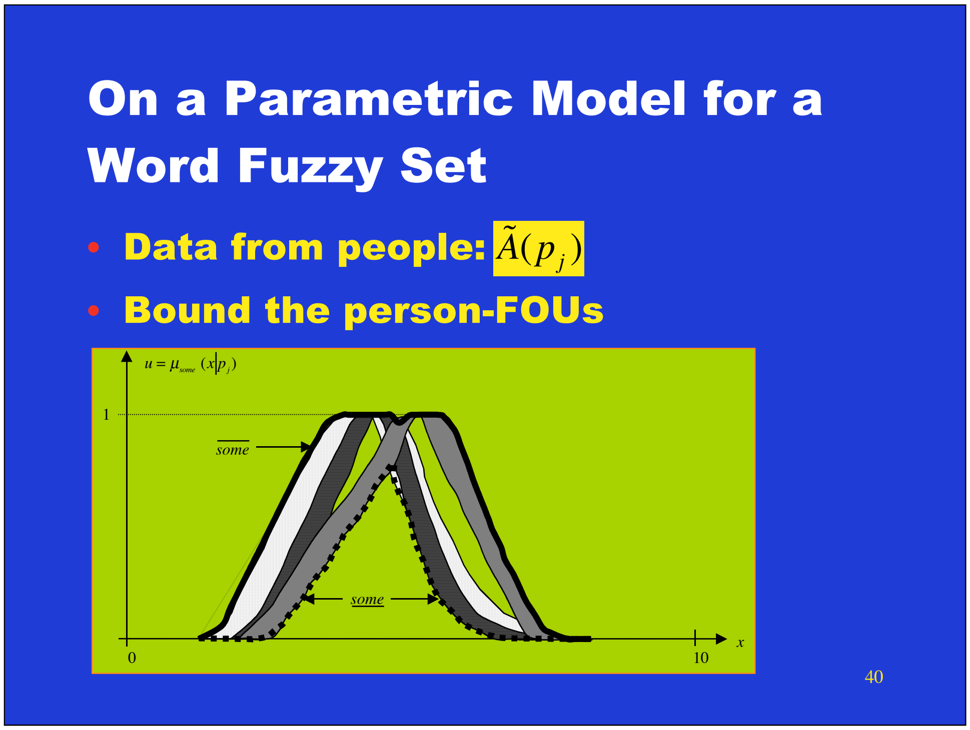
39

The word "parsimonious" has a nice ring to it.

It is a word that is frequently used in the system identification literature, where people seek models having the fewest number of parameters such that the models are consistent with the data.

So far, our composite FOU model is in terms of raw data, a collection of person FOUs.

We now propose to do what people commonly do in system identification, namely to replace raw data by a **parametric** mathematical model.



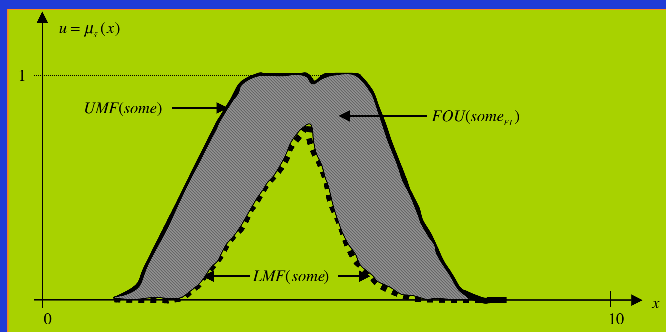
We now explain how to do this.

First, we bound the person FOUs, meaning that we establish lower and upper bounds for the union of all person MFs.

In the figure on this slide, the heavy solid curve is the upper bound, and the heavy dashed curve is the lower bound.

On a Parametric Model for a Word Fuzzy Set (Cont'd.)

- **Fill-in all points between the lower and upper bounds:** $\tilde{A}_{FI}$



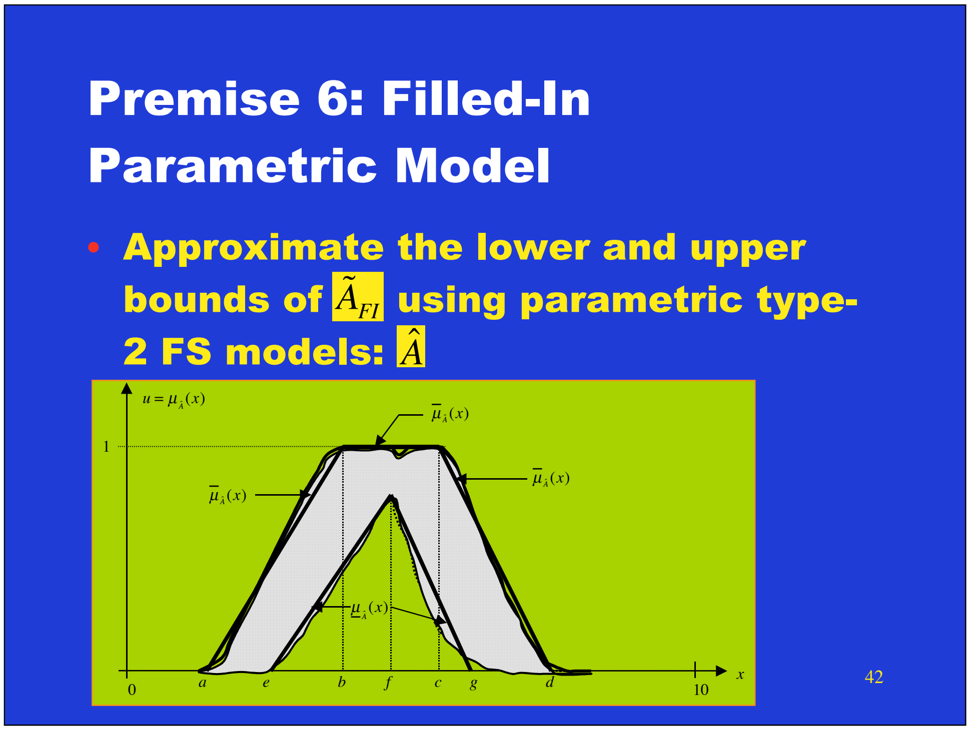
41

FI: filled-in

We fill in all points between the lower and upper bounds because we want the resulting FOU to be for an interval type-2 FS.

Such a FS cannot have gaps in its FOU.

Our rationalization for filling-in is that if we could collect person MFs from a large enough group of persons (e.g., at least 30), then it is highly likely that fill-in would occur automatically.



It is easy to approximate the lower and upper bounds for the filled in FOU.

The notation *A-hat* means the "approximation or estimate of *A*."

We begin by choosing parametric mathematical models (equations) for the lower and upper MFs, e.g. in the figure above, we use a triangle to model the LMF, and a trapezoid to model the UMF.

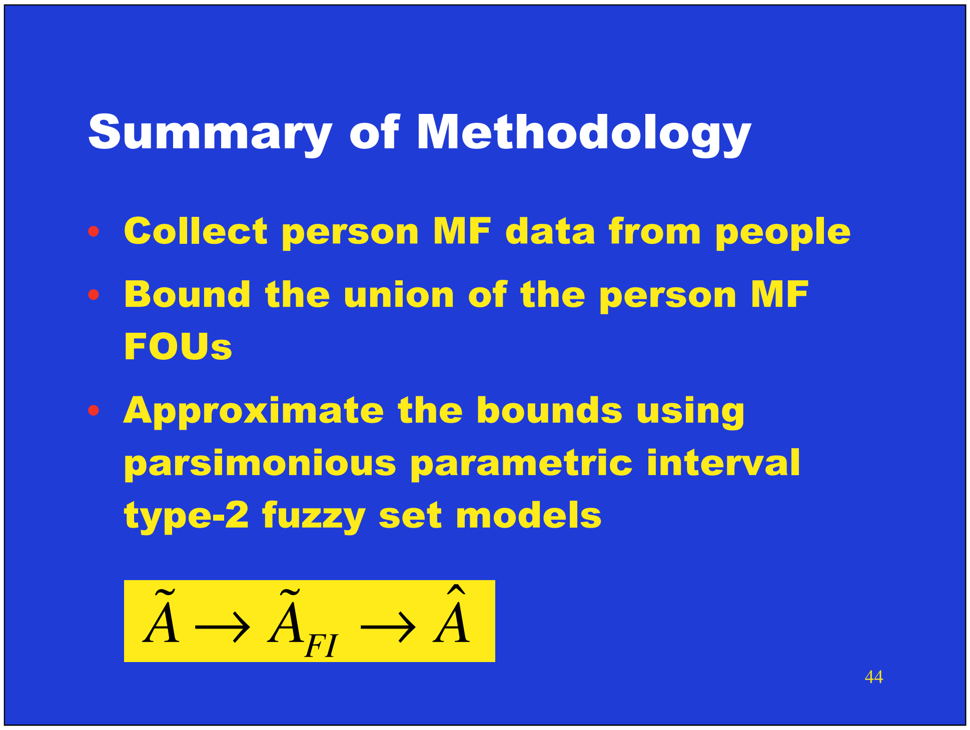
We then fit the parametric models to sampled values of the lower and upper MFs using the method of least squares. This is where the data is used.

Premise 6: Filled-In Parametric Model (Cont'd.)

- Choose parametric models for $\underline{\mu}_{\hat{A}}(x)$ and $\overline{\mu}_{\hat{A}}(x)$
- Use person MF data to optimize the parameters of the parametric models
- $$FOU(\hat{A}) = \bigcup_{\forall x \in X} \left[\underline{\mu}_{\hat{A}}(x), \overline{\mu}_{\hat{A}}(x)\right] \big/ x$$

43

Just as the composite person MFs led to a FOU, the parametric models for the lower and upper MFs also lead to a FOU, one that is for *A-hat*.



The methodology for how to model a word using an interval type-2 FS should now be clear.

A-hat preserves the uncertainties that a group of people have about a word. A type-1 FS cannot do this.

We again remind the reader that person MFs can only be collected from very knowledgeable people, and that other techniques for obtaining a FOU for a word must be established for all other people.

Partial Conclusions: 1

- Type-1 fuzzy sets as models for words is scientifically incorrect
- We have proposed a new interval type-2 fuzzy set model for words, one that accounts for intra-and inter-uncertainties and requires data
- Our new theory of fuzzy sets for words is testable and is therefore subject to refutation

45

Partial Conclusions: 2

- Research is needed on how to collect person MFs or other data that can lead to an aggregated FOU for a word
- *Type-2 Fuzzistics—from data to a FOU—a new field*
- Computing with words needs to use this new theory of fuzzy sets for words
- Interval type-2 fuzzy sets and their associated mathematics will be used

46

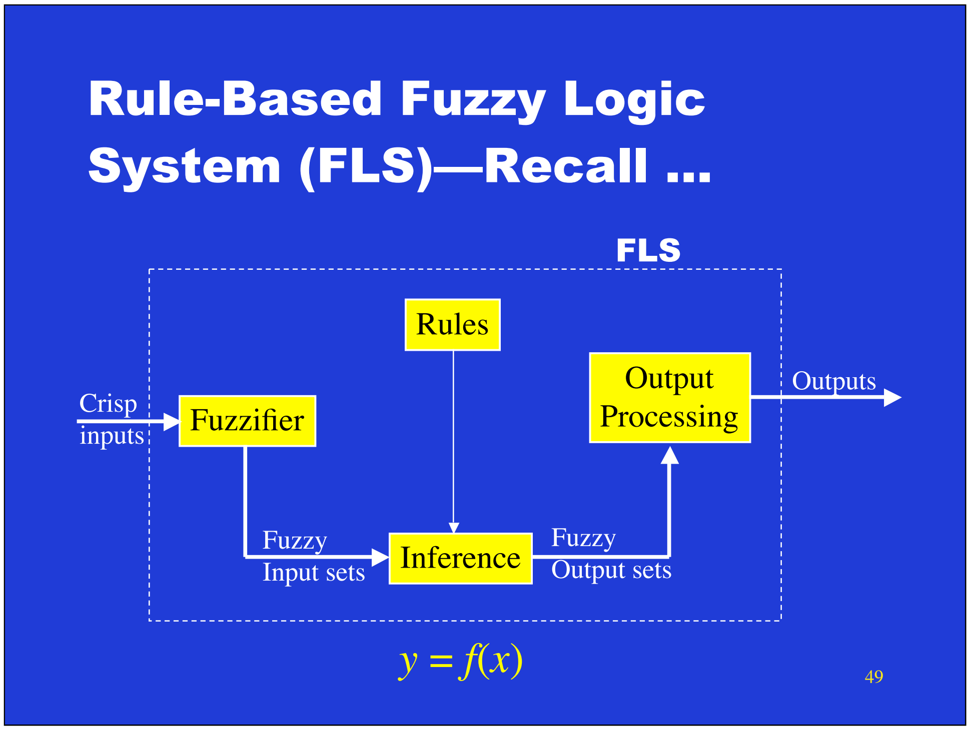
Partial Conclusions: 3

- It's okay to use type-1 fuzzy sets in non-word applications because we are basically using them as mathematics
- Even so, if an application has uncertainties associated with it, it may be better to use type-2 fuzzy sets
- So, let's now turn to the use of interval type-2 fuzzy sets for non-word applications, in rule-based fuzzy logic systems

47

Part IV. Rule-Based Interval Type-2 Fuzzy Logic Systems

48



Recall that a FLS is also known as a Mamdani FLS, fuzzy expert system, fuzzy model, fuzzy system, or fuzzy logic controller.

Rules are the heart of a FLS, and may be provided by experts or can be extracted from numerical data. In either case, the rules that we are interested in can be expressed as a collection of IF–THEN statements.

Fuzzy sets are associated with terms that appear in the antecedents or consequents of rules, and with the inputs to and output of the FLS. Membership functions are used to describe these fuzzy sets.

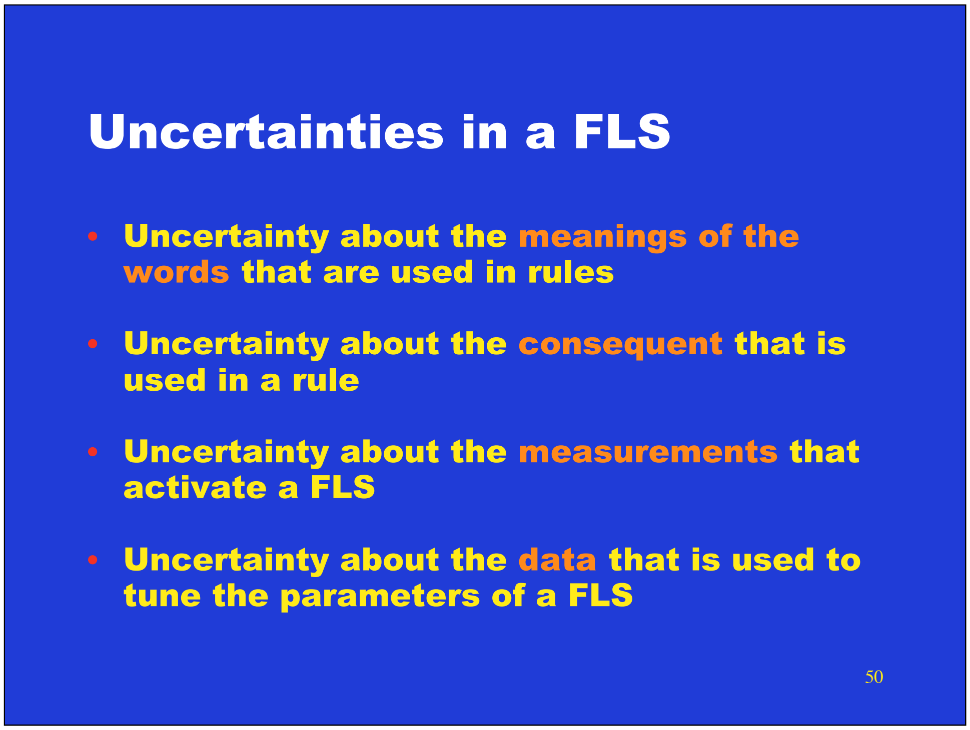
A FLS that is described completely in terms of type-1 fuzzy sets is called a *type-1 FLS*

The **fuzzifier** maps crisp numbers into fuzzy sets. It is needed to activate rules that are in terms of linguistic variables, which have fuzzy sets associated with them.

The **inference engine** of the FLS maps fuzzy sets into fuzzy sets. It handles the way in which rules are activated and combined.

In many applications of a FLS, crisp numbers must be obtained at its output. This is accomplished by the **output processor**, and is known as **defuzzification.**

The mathematical formula that expresses the defuzzified output of the FLS in terms of its inputs is $y = f(x)$; it represents a highly non-linear transformation of the inputs.



There can be four sources of uncertainty in a FLS. They are stated above.

We have already spent a lot of time explaining that, because words can mean different things to different people, there is uncertainty about the meanings of words.

Consequents for rules are either obtained from experts, by means of knowledge mining (engineering), or are extracted directly from data. Because experts don't all agree, a survey of experts will usually lead to a histogram of possibilities for the consequent of a rule. This histogram represents the uncertainty about the consequent of a rule, and this kind of uncertainty is different from that associated with the meanings of the words used in the rules

Measurements are usually corrupted by noise; hence, they are uncertain.

A FLS contains many design parameters whose values can be optimized by the designer before the FLS is operational. There are many ways to do this, and all make use of a set of data, usually called the *training set.* This set consists of input–output pairs for the FLS, and, if these pairs are measured signals, then they are as uncertain as are the measurements that excite the FLS. In this case—one that is quite common in practice—the FLS must be tuned using unreliable data, which is yet another form of uncertainty.

Uncertainties and Membership Functions

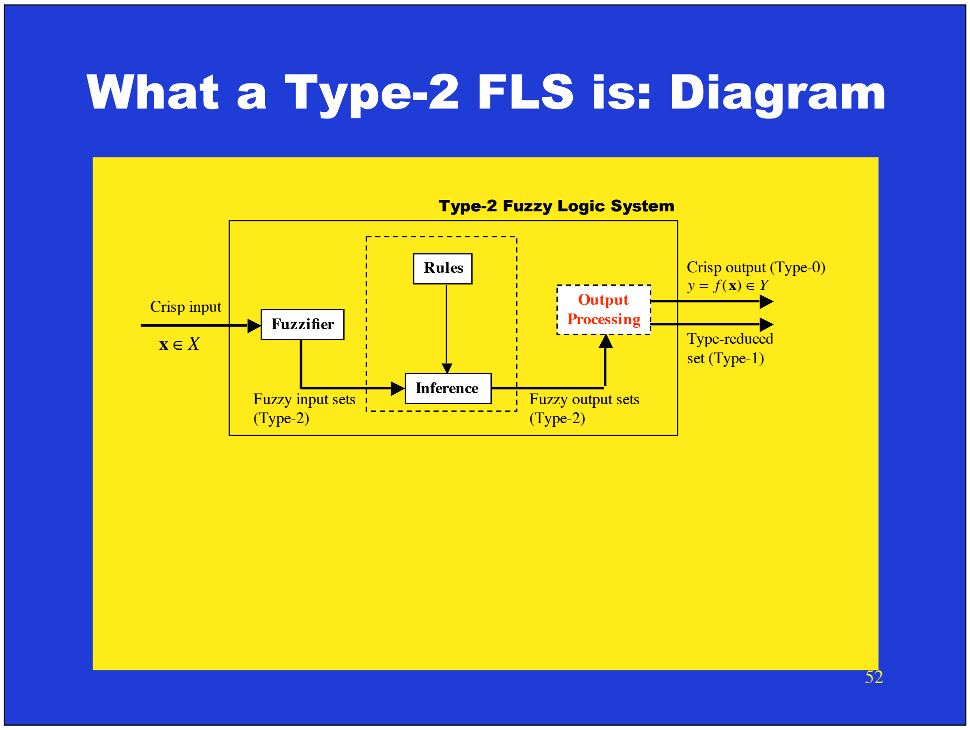
- **Uncertainties translate into uncertain membership functions**
- **Type-1 MFs are totally determined—there is nothing uncertain about them**
- **Hence type-1 MFs are unable to *handle* uncertainties**
- **We can handle uncertainties using interval type-2 fuzzy sets**

51

By "handle" we mean directly model so as to minimize the effects of.

Uncertainties can be handled using general type-2 fuzzy sets; but, the associated computations are unrealistic, which makes their use today impractical.

Handling uncertainties using interval type-2 fuzzy sets is very practical, because the associated computations are very manageable.



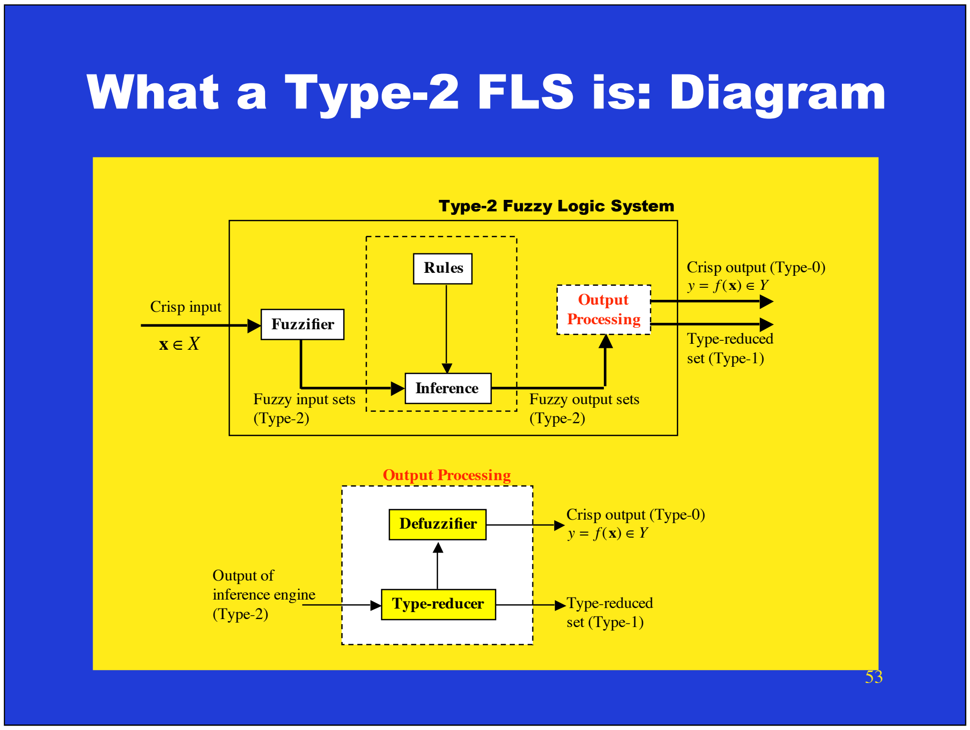
You should compare this diagram with the one given on slide 49. They look very similar.

Rules do not change as we go from using type-1 FSs to type-2 FSs. To paraphrase the famous author Gertrude Stein, a rule is a rule is a rule … . What does change is the way in which we model the words in the rule. We now model them as interval type-2 FSs.

Instead of type-1 FSs propagating from the inputs to the output, as happens in a type-1 FLS, it is interval type-2 FSs that propagate from the inputs to the output of the type-2 FLS.

We do not have the time to present the mathematics of interval type-2 FLSs, but it is pretty simple and can be found in Mendel (2001). Instead, we will illustrate the computations graphically on a later slide.

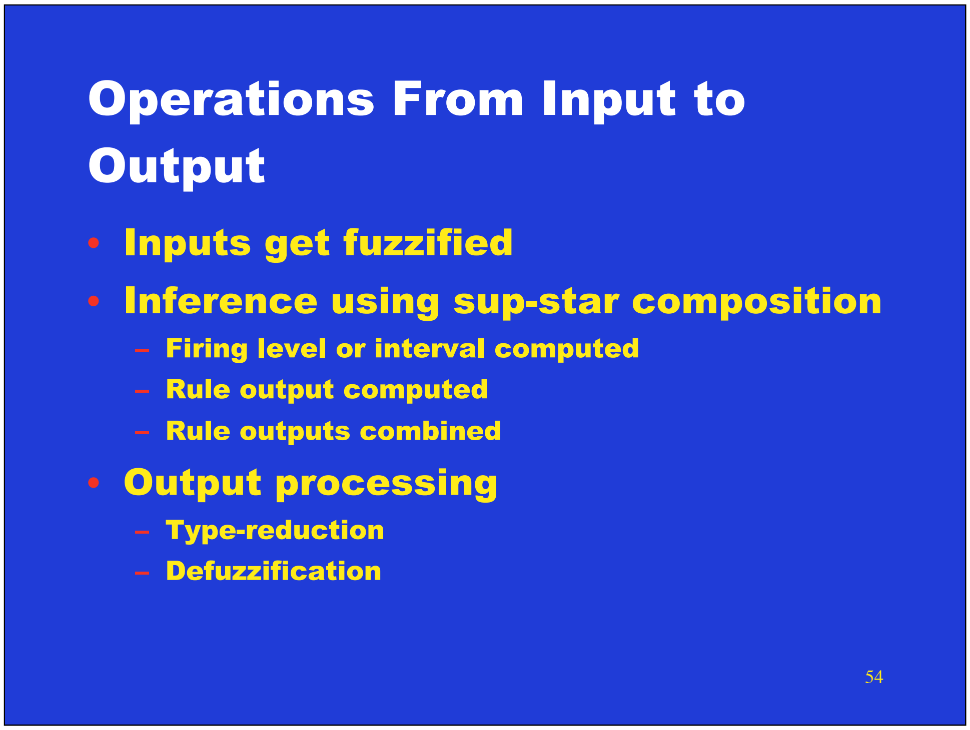
Output processing is explained on the next slide.



Output processing can provide two items: a **type-reduced set**, which is an interval and provides a measure of the uncertainty as it flows through (is transformed in) the type-2 FLS, and a crisp output which is the **defuzzified** output.

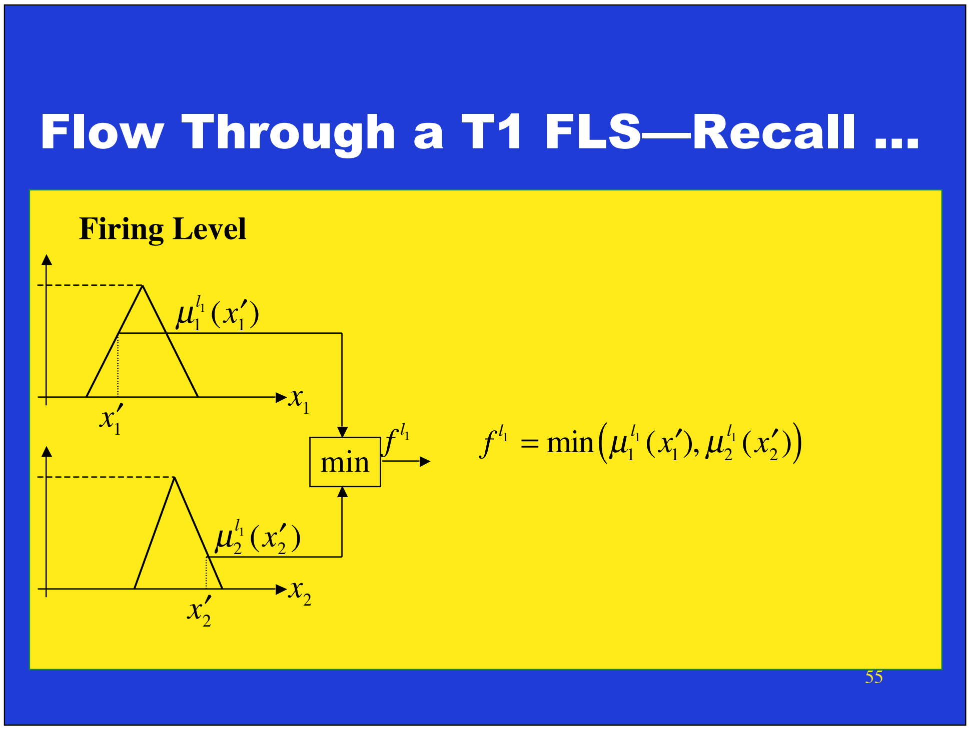
Because the defuzzifed output is a crisp number, it requires two steps to obtain it. Type-reduction maps a type-2 FS into a type-1 FS, and defuzzification maps that type-1 FS into a crisp number (a type-0 FS).

The type-reduced set provides a useful measure of uncertainty that is analogous to a confidence interval in statistics.



It is possible to derive all of the formulas that are needed to implement an interval type-2 FLS using type-1 mathematics! See, for example: J. M. Mendel, R. I. John and F. Liu, "On Using Type-1 Fuzzy Set Mathematics to Derive Interval Type-2 Fuzzy Logic Systems," *Proc. NAFIPS 2005*, Ann Arbor, MI, pp. 528-533.

A journal version of this paper should appear in 2006 in the *IEEE Trans. on Fuzzy Systems*.

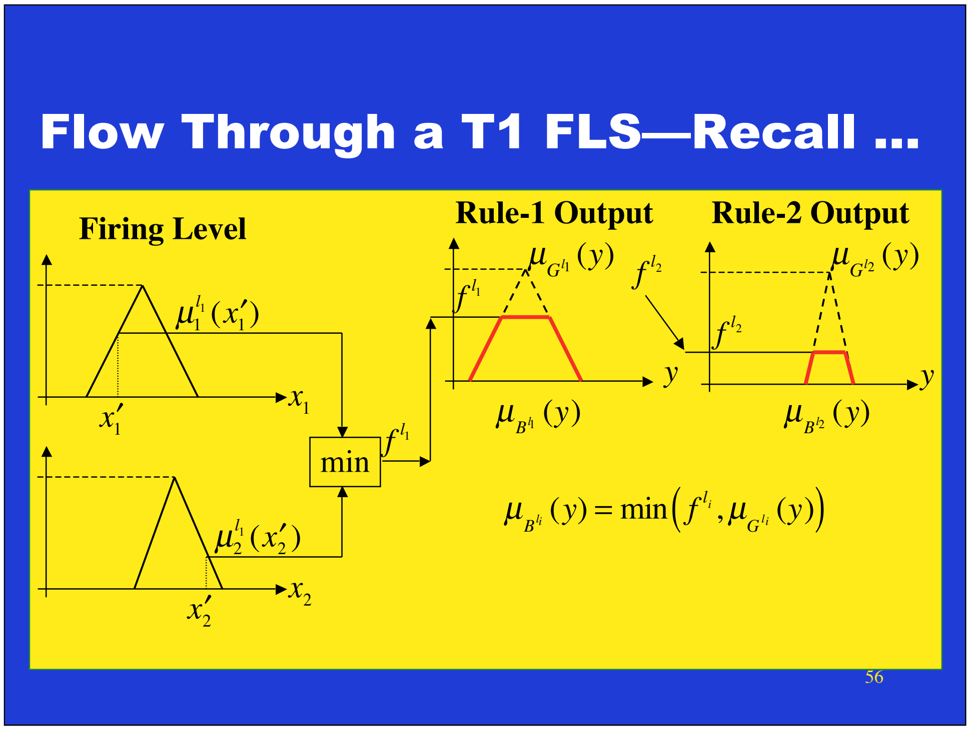


Before we illustrate the interval type-2 FLS calculations graphically, we recall the graphical illustrations of the comparable type-1 FLS calculations.

We are assuming a rule that has two antecedents. In the above figure the primed variables denote their measured values. These values are projected vertically until they intersect the respective MFs.

We have chosen triangle MFs and minimum t-norm. Of course, other shapes could be chosen for the MFs, and we could use other kinds of t-norms, e.g. product.

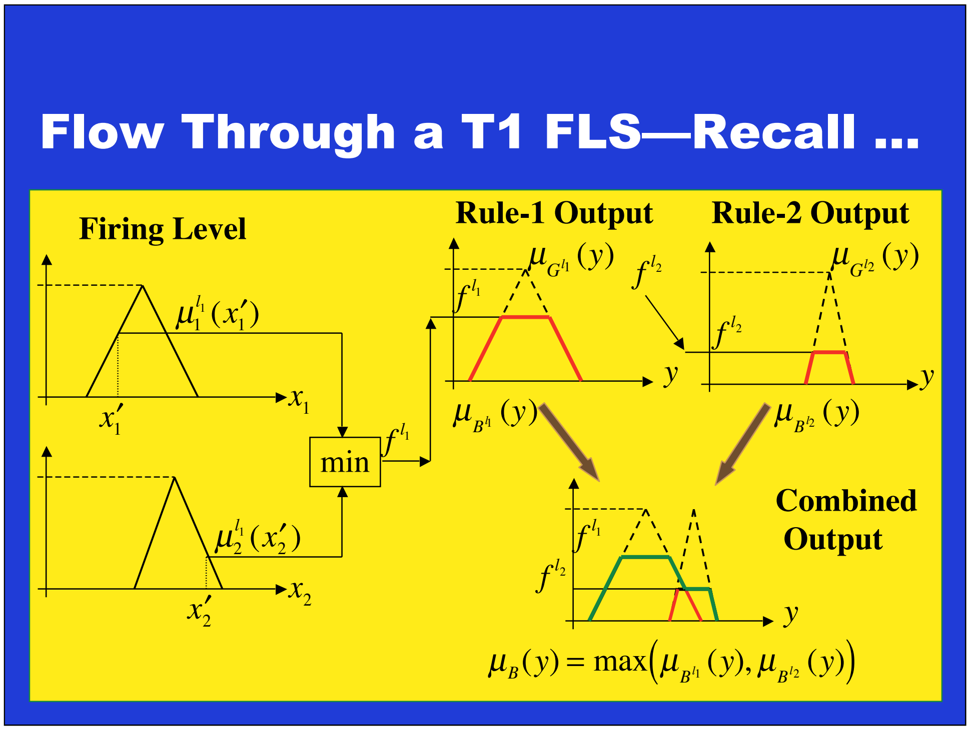
The minimum calculation provides a number called the "firing level." Every rule that "fires" produces such a firing level.



We assume that for the given measurements two rules have fired.

Each fired rule produces a "fired-rule output FS," obtained as the t-norm (minimum) between the firing level for that rule and the consequent FS for that rule. The formula for doing this is shown on this slide.

We show the results of these calculations for the two fired rules in red. They are trapezoids.



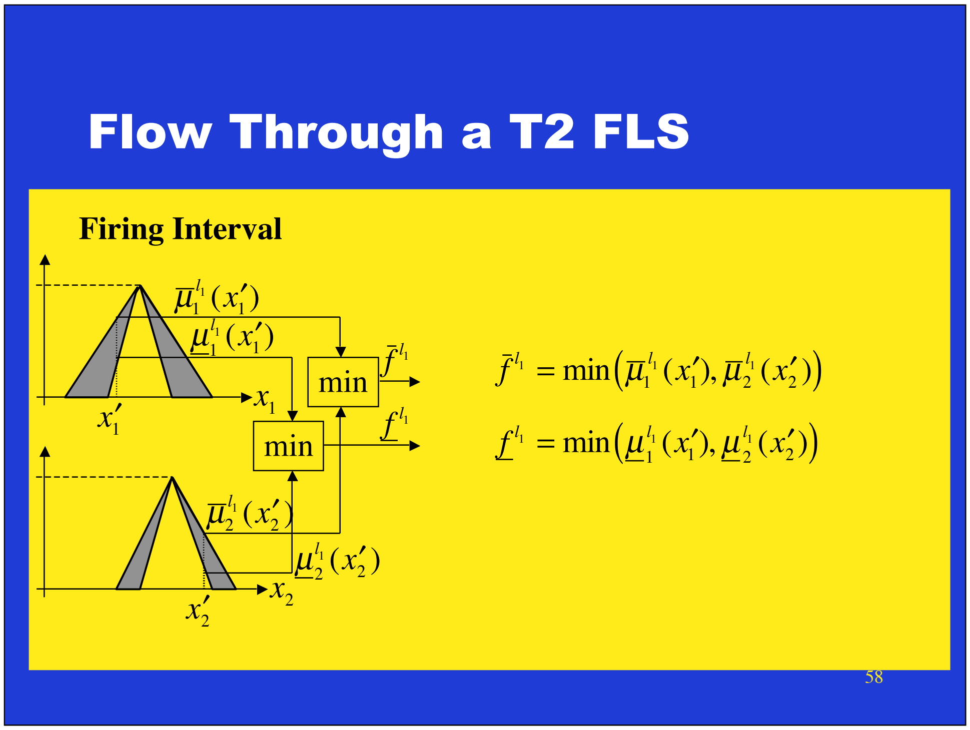
The two fired rule output FSs can then be combined (aggregated), and this can be done in many different ways.

For illustrative purposes, we show a union aggregation, where the union operation is the maximum. A formula for the aggregated output FS is shown on this slide.

The **combined output** is shown as the green MF.

If we want a crisp (defuzzified) output we could compute the center of gravity of this combined output function.

On the next few slides we show the comparable figures for an interval type-2 FLS.



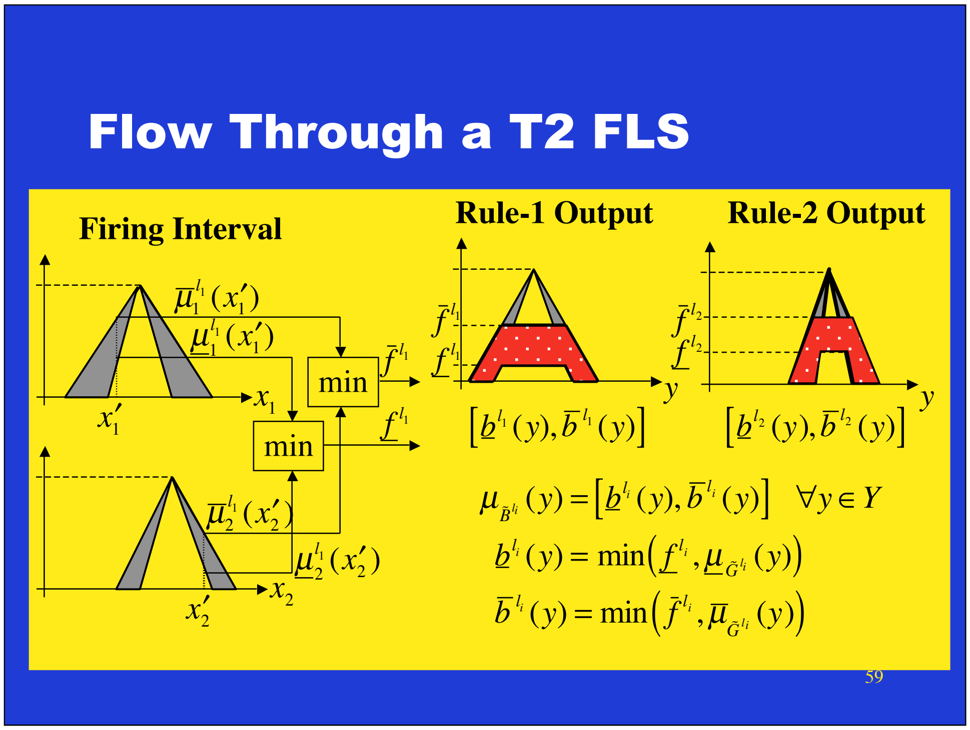
The mathematics of interval type-2 FLSs reveals that instead of computing a firing level for each fired rule, we have to compute a **firing interval**.

The lower bound for the firing interval involves type-1 operations that use only the lower MFs for the activated FSs.

The upper bound for the firing interval involves type-1 operations that use only the upper MFs for the activated FSs.

The firing interval contains all values between the lower and upper bounds.

Formulas for the lower and upper bounds of the firing interval are given on this slide.

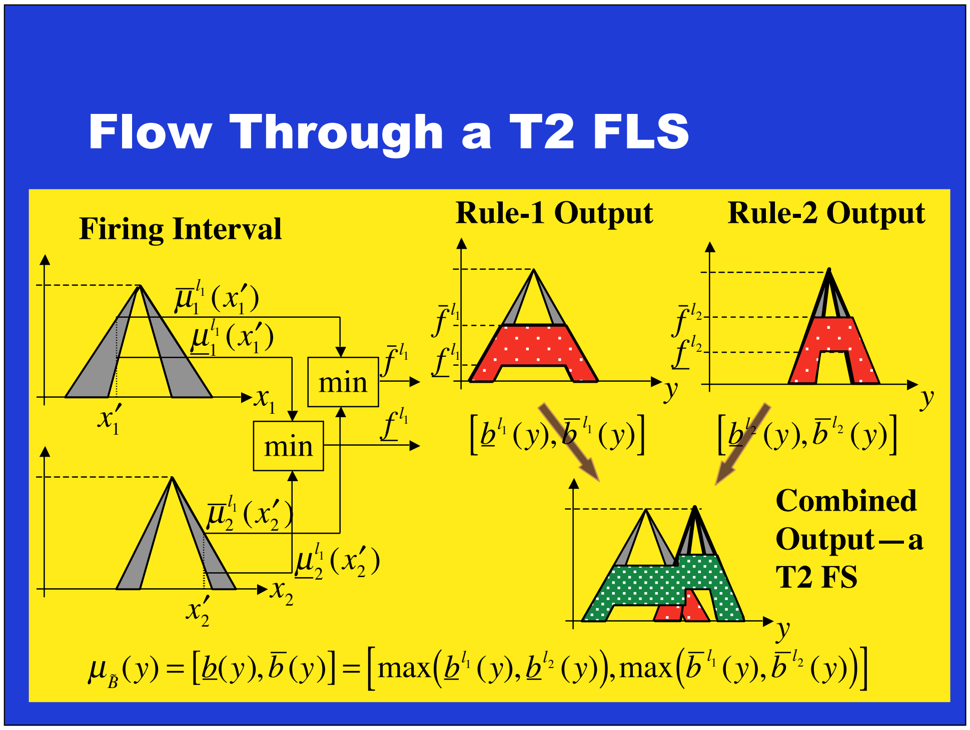


The mathematics of interval type-2 FLSs reveals that instead of computing a single type-1 rule output FS, we have to compute two such FSs for each fired rule—a LMF for the rule-output FS and an UMF for the rule-output FS.

Each of the two calculations again only involves type-1 FS calculations, one for the LMF and one for the UMF of the fired rule-output fuzzy sets. Equations for these calculations are shown on this slide.

The result is a fired-rule-output interval T2 FS, or FOU, shown in red on this slide.

The reason that we obtained a FOU (rather than just the lower and upper MFs) from these calculations is because we began with a firing set, which is a continuous set of points between its LMF and UMF; hence, we must fill in all points between the lower and upper MFs for each fired consequent FS.



The two fired rule output FSs can then be combined (aggregated), and this can also be done in many different ways.

For illustrative purposes, we show a union aggregation, where the union operation is the maximum.

The formula at the bottom of this slide provides the details for how the union is computed. Not too surprisingly, it only involves calculations involving lower and upper MFs.

The **combined output** is shown as the green FOU.

From a Type-2 Output Set to a Number

$$\mu_{\tilde{B}}(y)=\left[\underline{b}(y),\bar{b}(y)\right]=\left[\max\left(\underline{b}^{l_1}(y),\underline{b}^{l_2}(y)\right),\max\left(\bar{b}^{l_1}(y),\bar{b}^{l_2}(y)\right)\right]$$

- **How do we go from this type-2 output set to a number?**
- **First we compute the centroid of this FS—type-reduction**
- **Then we defuzzify the centroid**

61

Referring back to Slide 53, we must now describe **output processing** for an interval type-2 FLS.

The equation shown on this slide is a repeat of the equation on the last slide.

Type-Reduction: 1

- Fuzzifier
- Rules
- Inference
- Type-reduction
- Defuzzification

- **Type-reduction begins with the type-2 output of the inference engine of a type-2 FLS**
- **It reduces a type-2 fuzzy set to a type-1 fuzzy set, which we call the type-reduced set**
- **Type-reduction methods are "extended" versions of type-1 defuzzification methods, and compute the centroid of the type-2 output fuzzy set**

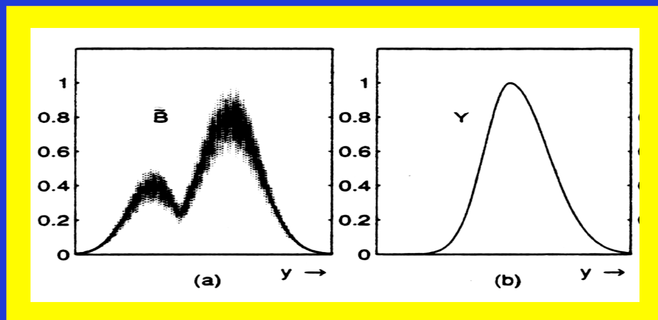
62

The small box in the upper right-hand corner of this slide is meant to remind you where we are in the stream of calculations.

Because there are many type-1 defuzzifcation method, there are a comparable number of type-reduction methods. See Mendel (2001) for discussions about some of them.

Type-Reduction: 2

- Fuzzifier
- Rules
- Inference
- Type-reduction
- Defuzzification

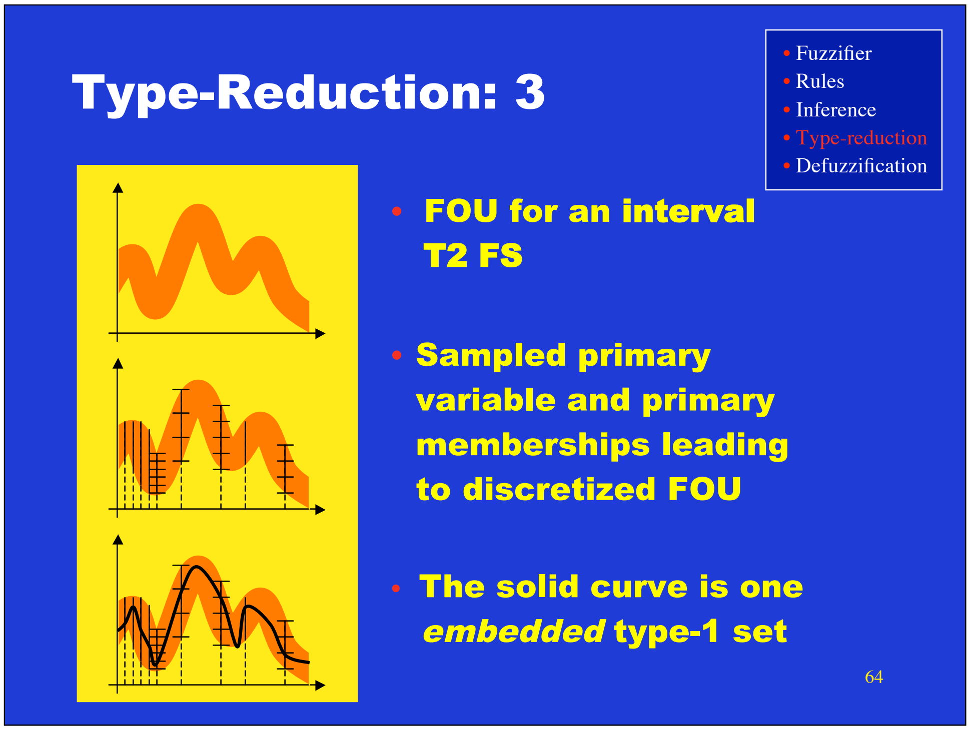


- **Type-reduction represents the transformation from the type-2 fuzzy set in (a) to the type-1 fuzzy set in (b)**
- **Type-reduction is computationally costly to do except for interval type-2 fuzzy sets**

63

Although the MF in the right-hand figure looks Gaussian, its shape depends on the nature of the general type-2 FS in the left-hand figure, meaning it depends on the FOU and the secondary grades of the type-2 FS.

For discussions on how to perform type-reduction for general type-2 FSs, see Mendel (2001).

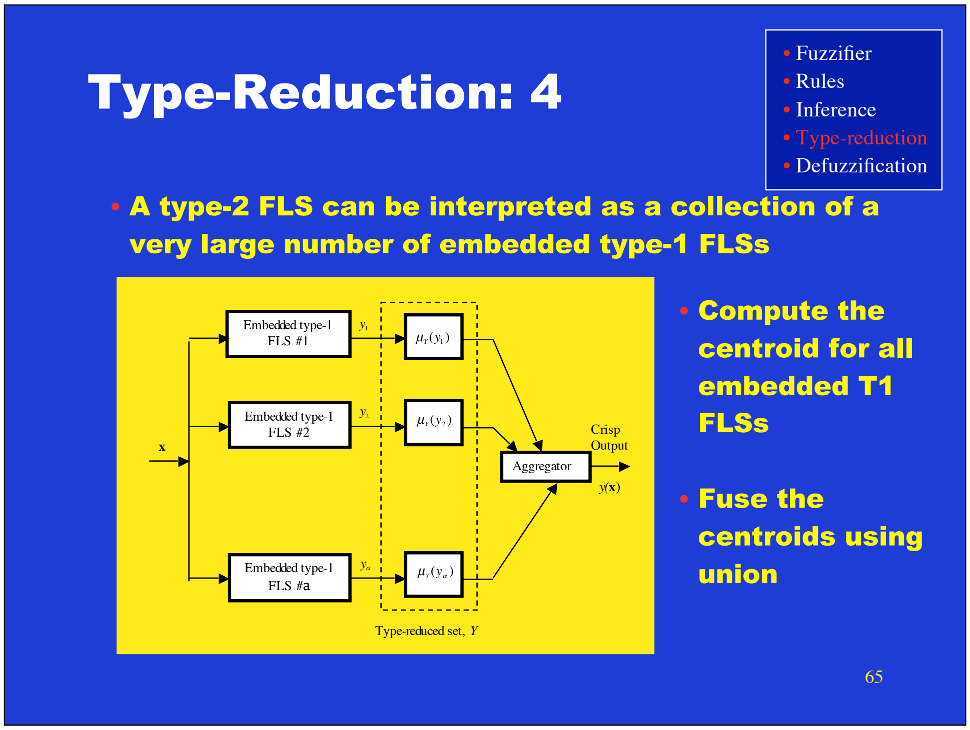


Start with the top figure in this three-part figure.

Whenever we perform computations for a FS using a digital computer we must discretize the primary variable and the primary membership.

Recall that the FOU is the union of all of its embedded type-1 FSs.

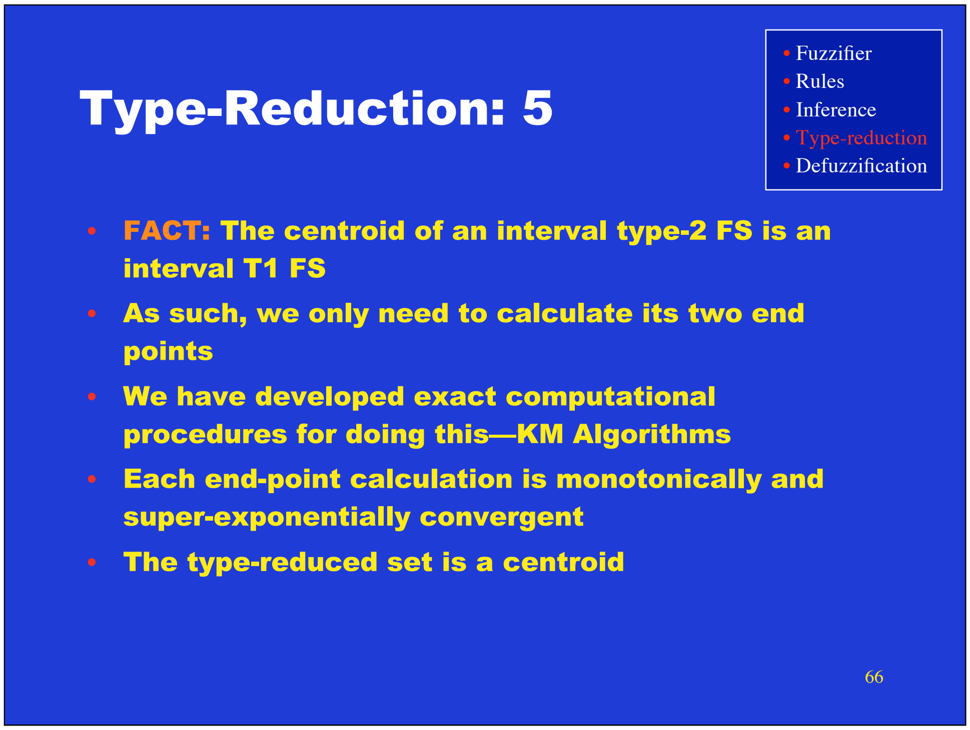
The neat thing about each embedded type-1 FS is it is a type-1 FS, so we can compute its center of gravity using existing type-1 formulas.



This figure provides us with an interpretation for a general type-2 FLS, one that lets us easily communicate what a type-2 FLS is to others—a collection of a very large number of embedded type-1 FLSs.

Of course, because the number of embedded type-1 FSs for the FOU of a general type-2 FS is astronomical, we would never think of actually computing within a general type-2 FLS using all of these embedded type-1 FSs.

Things simplify enormously for an interval type-2 FLS.



The proof of the "Fact" is very simple: even though there may be an astronomical number of embedded type-1 FSs in the FOU, there will be embedded type-1 FSs which have the smallest and largest centroid; hence, the centroid of an interval type-2 FS is an interval type-1 FS. The membership value for all points in this interval equals 1, because the secondary grades for an interval type-2 FS all equal 1.

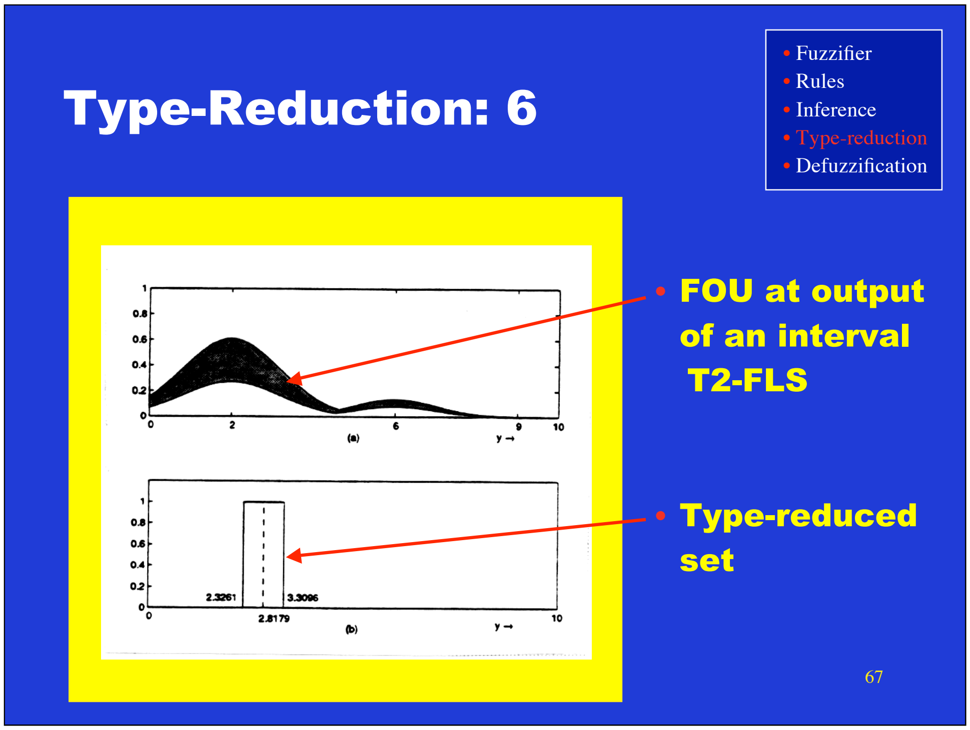
The KM (Karnik-Mendel) algorithms are iterative and each of the two algorithms can be run in parallel. Recently, these algorithms have been proven to converge to their exact answer monotonically and super-exponentially fast (really fast!).

References for the KM algorithms are: N. Karnik and J. M. Mendel, "Centroid of a Type-2 Fuzzy Set," *Information Sciences*, vol. 132, pp. 195-220, 2001; and,

Mendel (2001), Chapter 9.

A type-reduced set is the centroid of a specific T2 FS computed for a T2 FLS.

An example of a type-reduced set is the centroid of the union of fired rule output sets (see the green shaded FOU on slide 60). Other kinds of type-reduced sets can also be computed.



The KM algorithms were used to compute the left and right end-points of the type-reduced set that is shown in the bottom figure on this slide.

Again, the type-reduced set is an interval type-1 FS. Its center of gravity is dashed.

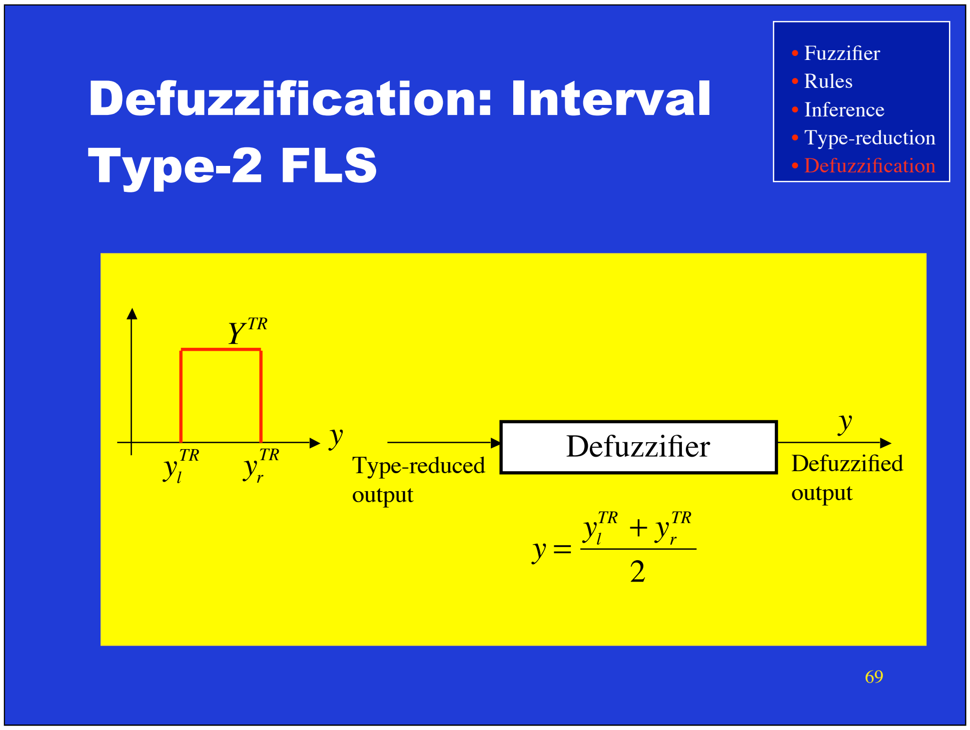
Intuitively, we expect the width of the type-reduced set to increase as the area of the FOU increases and to decrease as the area of the FOU decreases.

In fact, if all uncertainty disappears then the FOU becomes a curve—a type-1 FS—and the type-reduced set is a single number, i.e. the type-2 FLS reduces to a type-1 FLS.

Defuzzification

- Fuzzifier
- Rules
- Inference
- Type-reduction
- Defuzzification

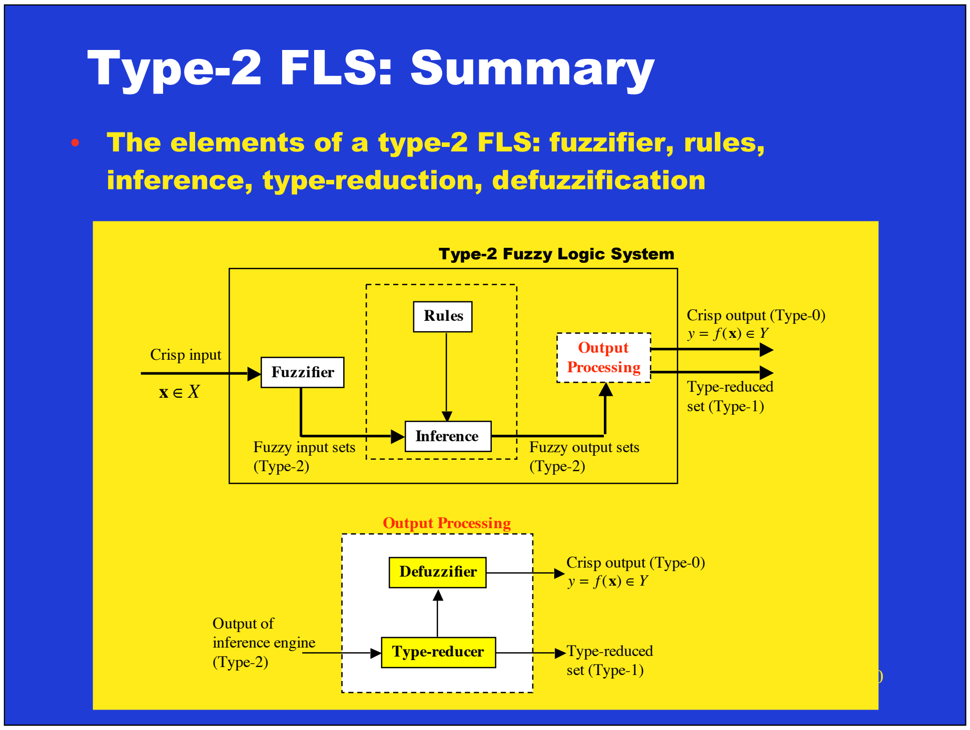
- **We defuzzify the type-reduced set to get a crisp output from a type-2 FLS**
- **The most natural way of doing this seems to be by finding the centroid of the type-reduced set**



TR: Type-reduced

Once the type-reduced set has been computed, defuzzification is trivial.

The formula for the defuzzified output is shown on this slide.



We have now completed all of the computations needed to implement an interval type-2 FLS. They really are quite simple.

These computations let any or all of the four sources of uncertainties (see slide 50) propagate through the interval type-2 FLS.

Having both the defuzzified output and its associated type-reduced set is *analogous* to having both the mean and its associated standard deviation for a random variable.

Comments: 1

- **There are more parameters in an interval T2 FLS than in a T1 FLS; hence, more design degrees of freedom in a type-2 FLS**
- **More design degrees of freedom suggest the potential to obtain better performance with a type-2 FLS than with a type-1 FLS**
- **Methods similar to those used to design a type-1 FLS can be adapted to a type-2 FLS**

71

The **parameters** of an interval type-2 FLS are those of its MFs.

If, for example, we used a Gaussian type-1 MF (in a type-1 FLS) it would be characterized by two parameters—its mean and standard deviation. On the other hand, if that Gaussian MF acted as the primary MF for an interval type-2 FS, and we only assumed uncertainty about its mean value, then the resulting interval type-2 MF would be characterized by three parameters—the two end-points of the interval of uncertainty for the mean and the standard deviation.

The word "potential" means that we cannot guarantee ahead of time that a type-2 design will significantly outperform a type-1 design. This issue is application dependent and also depends on how much uncertainties are present.

By "design of a type-1 FLS," we mean optimizing its MF parameters.

There are a multitude of methods that have been developed to design type-1 FLSs.

Comments: 2

- **Given a training set of data, the parameters of T1 and T2 FLSs can be tuned using steepest descent algorithms—Neural-Fuzzy T2 FLSs**
  - **Greater care is needed for computing the derivatives in a T2 FLS**
- **Rules can be pruned using SVD**
- **Genetic optimization can also be used**

72

For details on how steepest descent algorithms (also called "back-propagation) can be applied to the designs of interval type-2 FLSs, see Mendel (2001). Even though there is nothing "neural" about doing this, it is now customary to refer to such a FLS design as a **neural-fuzzy FLS**.

It is the operation of type-reduction, through the use of the KM algorithms, that makes the computation of derivatives a bit tricky. This is explained fully in Mendel (2001).

SVD: Singular value decomposition. Suppose you start our with a FLS that has, e.g. 100 rules. Are all of these rules needed? SVD is a way to answer this question. It can pick out the really important rules and let you discard the unimportant ones. It is applied after the 100 rule FLS has been designed.

Genetic algorithms can be used to establish which antecedents are really needed in each rule, and even which rules are needed. It is very useful in design situation where variables are not differentiable, e.g. an antecedent variable can be introduced that is 1 if the antecedent should be present and is 0 if it should not be present. Such a 0-1 variable is not differentiable, so a steepest descent algorithm cannot be used for it.

Using a T2 FLS in Real-Time: 1

- **Type reduction is a computational bottleneck because of its iterative computations**
- **Wu and Mendel (*TFS*, 2002) derived inner- and outer-bound sets for the type-reduced set of an interval type-2 FLS, and showed how to use these bounding sets to design an interval type-2 FLS**
- **These bounding sets can be computed *without* having to perform type-reduction**

73

In a real-time applications of a FLS, type-reduction will introduce a computational delay, because the KM algorithms are iterative (even though they are very fast). Research is underway at many places today (2005) on how to overcome this computational bottleneck.

Some of the most promising work is being performed by Prof. Hani Hagras and his students at the Univ. of Essex, in the UK.Their approach is to completely replace the real-time calculation of type-reduction with bounds for the centroid end-points. These bounds are derived in: 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, pp. 622-639, Oct. 2002.

Hagras and his students have shown that very very little performance is lost by using the bounds instead of full-blown type- reduction. A reference for their work is: C. Lynch, H. Hagras and V. Callaghan, "Embedded Type-2 FLC for Real-Time Speed Control of Marine and Traction Diesel Engines." *Proc. IEEE FUZZ Conf.*, pp. 347-352, Reno, NV, May 2005.

Using a T2 FLS in Real-Time: 2

- **Type-reduction is used during the design of the type-2 FLS, when MF parameters are tuned**
- **A new cost function is introduced to do this, one that is a weighted combination of two terms, the usual MSE and a measure of error between using type-reduction and not using it**

74

MSE: Mean-squared-error

Wu and Mendel (2002) have an alternative approach. It may not be a better approach to the Hagras approach—no one is yet sure which approach is the best one—but is still worth explaining.

In their approach, type-reduction is still used during the design of the FLS, but is not used during the real-time operation of the FLS.

To make this possible, a new cost function is used.

Using a T2 FLS in Real-Time: 3

- **Using this design some MSE is traded for reducing the output error of the type-2 FLS with and without type-reduction**
- **Only the bounding sets are used during real-time operation of the type-2 FLS**

75

This new cost function lets the overall design trade-off some MSE in performance so that the defuzzified output based on using only the centroid bounds is so close to the true defuzzified output obtained when type-reduction is used, that type-reduction does not have to be used during the real-time operation of the FLS.

Again, the bounding sets can be computed without having to perform type-reduction.

So, it now appears that an interval type-2 FLS can be used in real-time applications.

This is especially important if such a FLS is to be used as a fuzzy logic controller.

Suitable Applications

- **We have found that T2 FLSs out-perform T1 FLSs in situations where:**
  - **Underlying data-generating system is time-varying (e.g. mobile communication channel)**
  - **Measurement noise is non-stationary (e.g., variable SNR or SINR)**
  - **Features in a pattern recognition problem are associated with quantities whose statistics are non-stationary**
  - **Knowledge is extracted from people—words**

76

This slide addresses the important question "What kinds of applications should we think about using an interval T2 FLS for instead of a type-1 FLS?"

As of this date (2005), many applications are being found for interval type-2 FLSs. Works on such applications are being conducted around the globe. See slide 100 for an Internet link to a comprehensive list of publications about type-2 subjects.

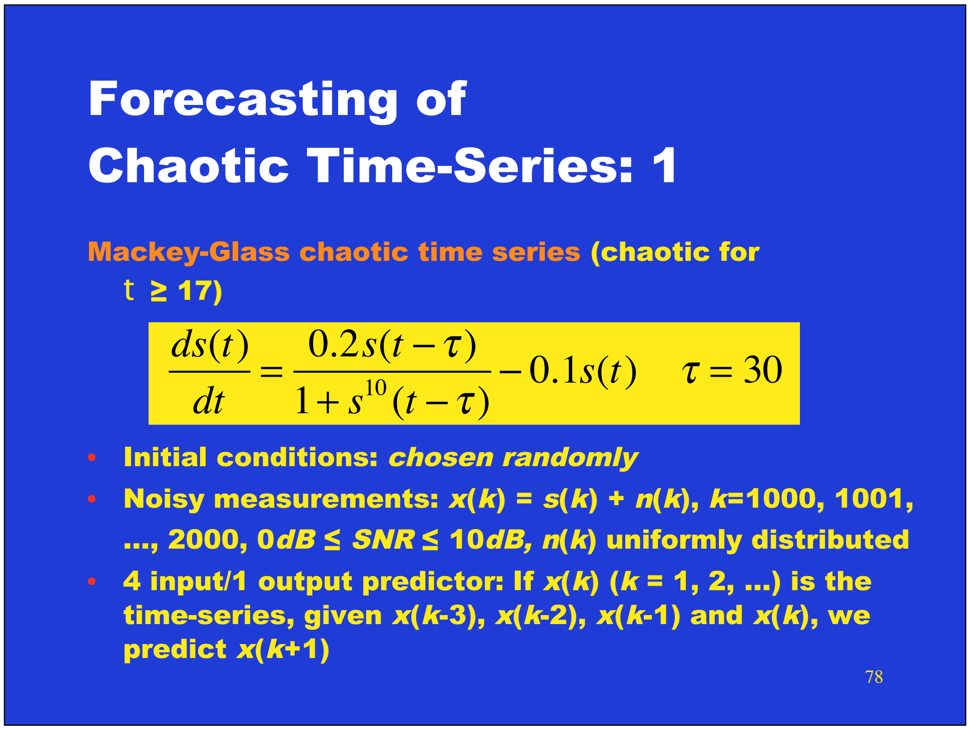
SNR: Signal-to-noise ratio

SINR: Signal-to-interference noise ratio

Part V. Two Applications

77

The two applications are: forecasting of a chaotic time-series and rule-based classification.



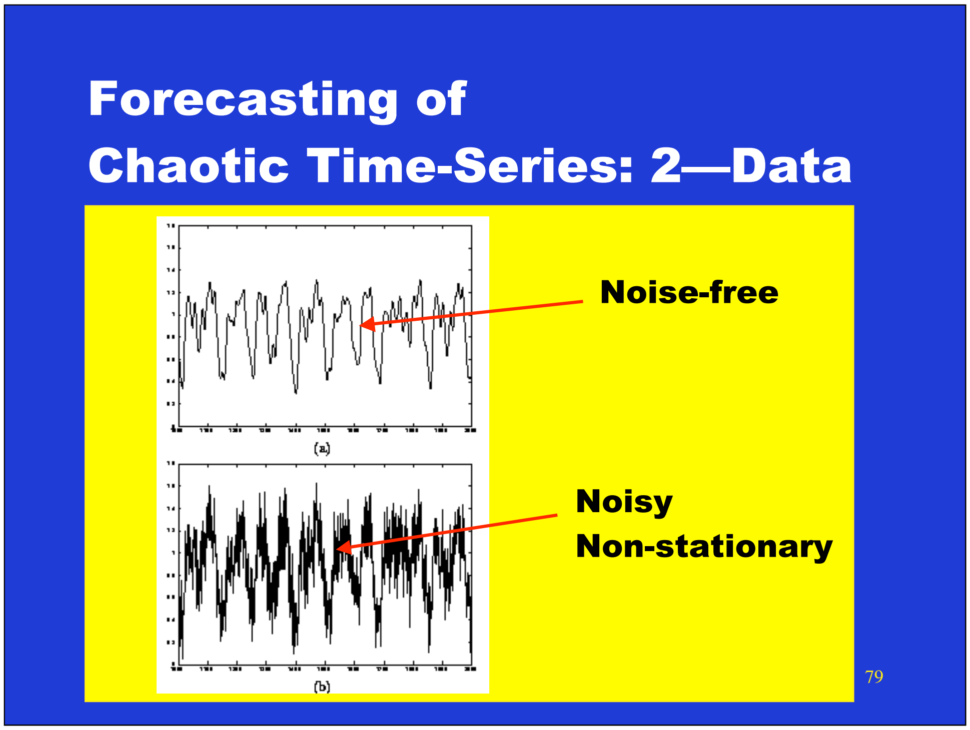
The Mackey-Glass time series is a widely used "test case' in the computational intelligence literature.

In 1977 Mackey and Glass published an important paper in which they "associate the onset of disease with bifurcations in the dynamics of first-order differential-delay equations which model physiological systems." Equation (4b) of that paper has become known as the *Mackey–Glass equation*. It is a non-linear delay differential equation, one form of which is shown on this slide.

What distinguishes our application of this time series from many others is we begin with noisy measurements whereas most other studies do not. Our noise is non-Gaussian, and at each time point its SNR varies.

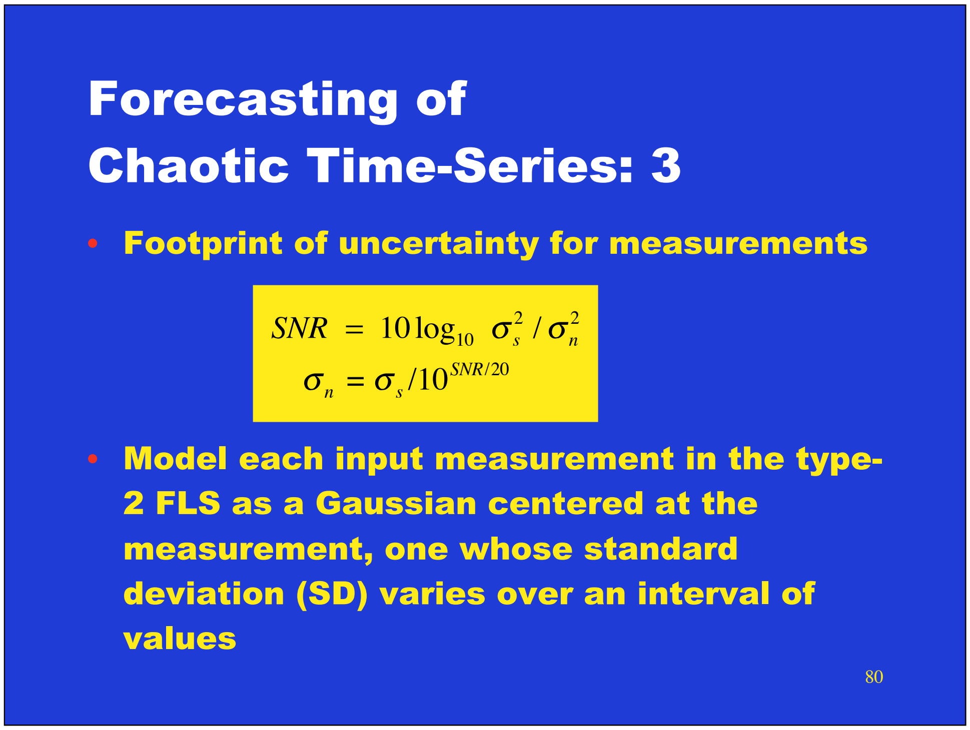
We chose the initial conditions of this equation randomly until we got a time series that "looked interesting". It is shown on the next slide. This was the time-series we used for this forecasting application.

We arbitrarily decided to design a FL predictor of $x(k)$ using only four previous values of x.



The top figure is a noise-free Mackey-Glass time series. Although it looks random it is not random (it is chaotic), i.e. for its known initial conditions, every repeated simulation of it will look exactly the same.

The bottom figure is the top figure with uniformly-distributed variable-SNR measurement noise added to it.



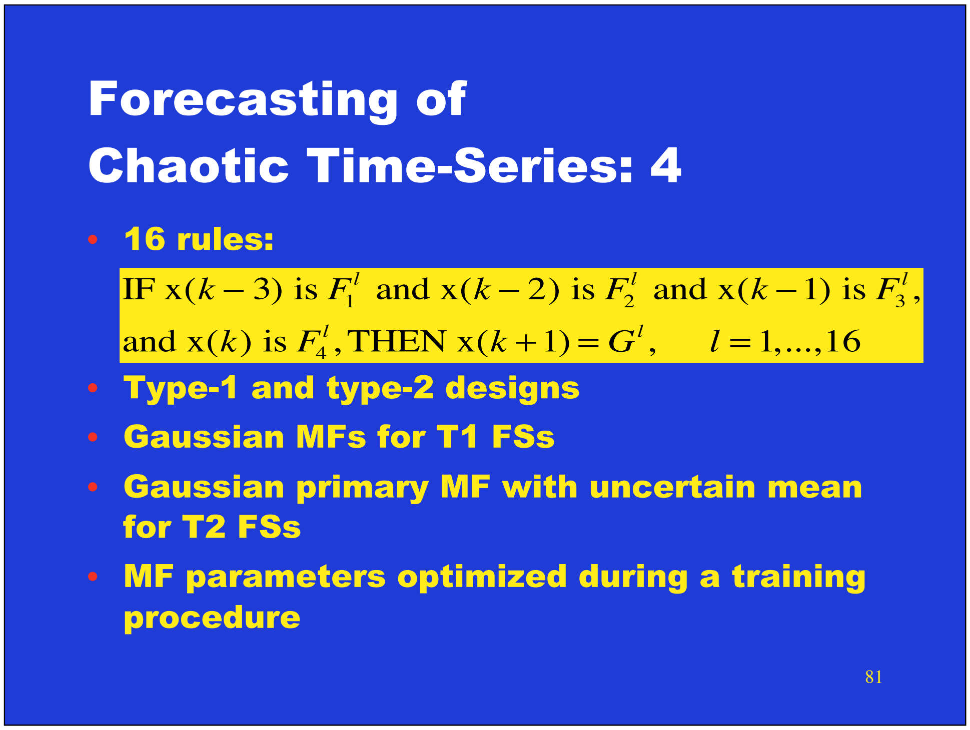
When we use probability-based methods to solve a problem, one of the first decisions we have to make is what to choose for the underlying probability distributions of the sources of randomness.

Analogously, when we use interval type-2 FLSs, one of the first decisions we have to make is what to choose for the FOUs of the MFs.

To do this we must understand the natures of the uncertainties that are present in our application. As shown on this slide, SNR depends on the variance of the signal and the variance of the additive noise.

Solving the SNR equation for the standard deviation of the additive noise, we see that it can be interpreted as varying over some unknown interval of values, because of the variable but unknown SNR and the unknown signal standard deviation.

It is this kind of analysis that suggests we choose the input measurement FOU that is stated on this slide.



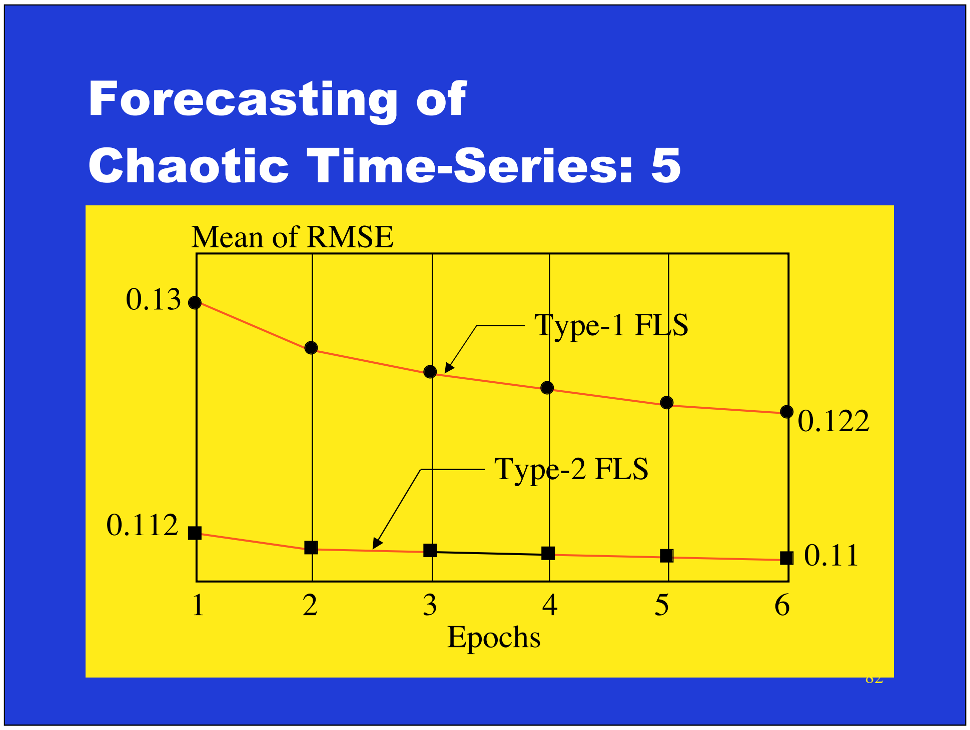
This is a very simple fuzzy forecaster. Since there are four antecedents, the smallest number of fuzzy sets that one can assume for each antecedent is two; hence, there can be 2^4=16 rules.

We always baseline our type-2 FLS designs against a comparable type-1 FLS design.

We used Gaussian type-1 MFs for the antecedents of the type-1 FLS and Gaussian primary MFs with uncertain means for the antecedents of the interval type-2 FLS.

From the previous slide, we also used Gaussian primary MFs with uncertain standard deviations for the measurements that activated the type-2 FLS. In the parlance of a FLS, this means we are using a **non-singleton FLS** (Mendel, 2001) in this application, in order to correctly model the noisy measurements.

First we designed the type-1 FLS, and then we used its MF parameters to initialize the design of the type-2 FLS. All designs were accomplished using steepest descent algorithms.



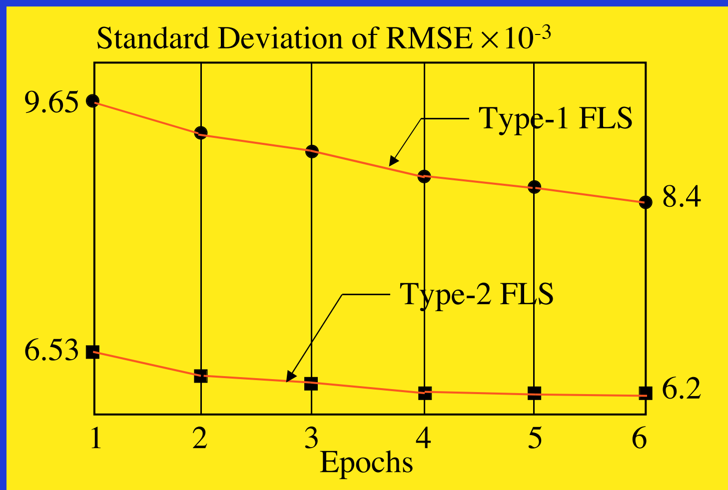
Epoch: We used 500 training data and updated the MF parameters for each of the datum. Then we tested the forecaster on another set of data. Each of the testing RMSE values shown here is at such an epoch point. Training and testing were done for six epochs.

Because our data is random, we performed Monte-Carlo simulations, i.e. we repeated the entire training and testing process 50 times. We then averaged the RMSE performance over those 50 runs.

The mean values of the RMSE for both the type-1 and interval type-2 FLSs are shown on this slide.

There really is not a great deal of difference between the two mean-value curves (see the scale on the figure), however …

Forecasting of Chaotic Time-Series: 6

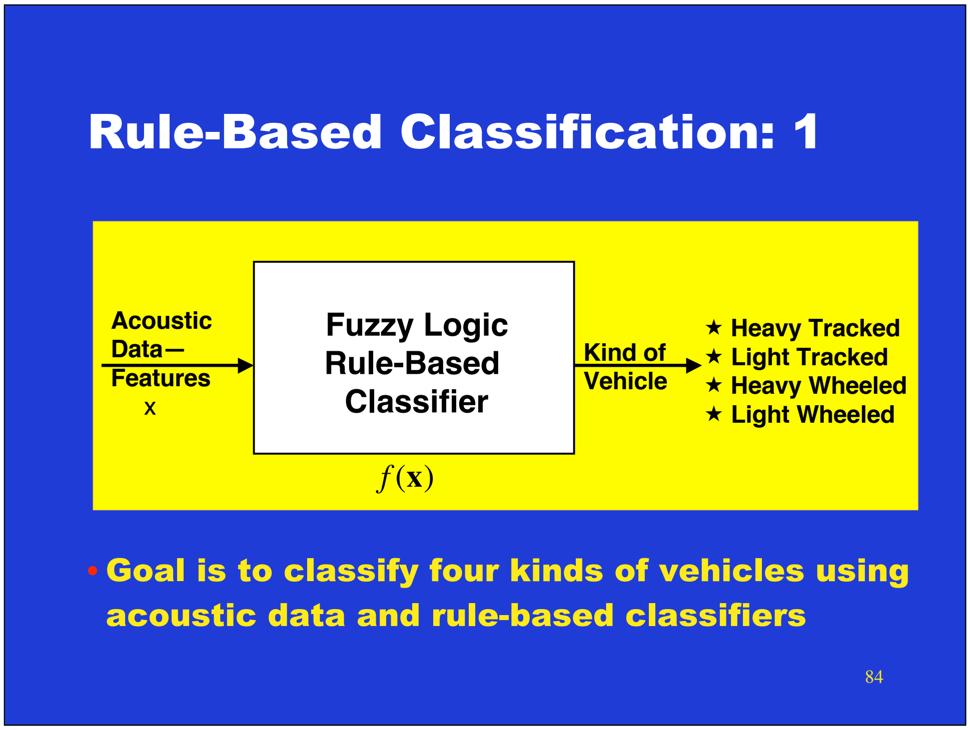


83

There is a significant difference between the RMSE standard deviations— 6.2×10^{-3} for the interval type-2 FLS versus 8.4×10^{-3} for the type-1 FLS.

We can say, therefore, that the interval type-2 FLS is more robust to the non-stationary noise than is the type-1 FLS.

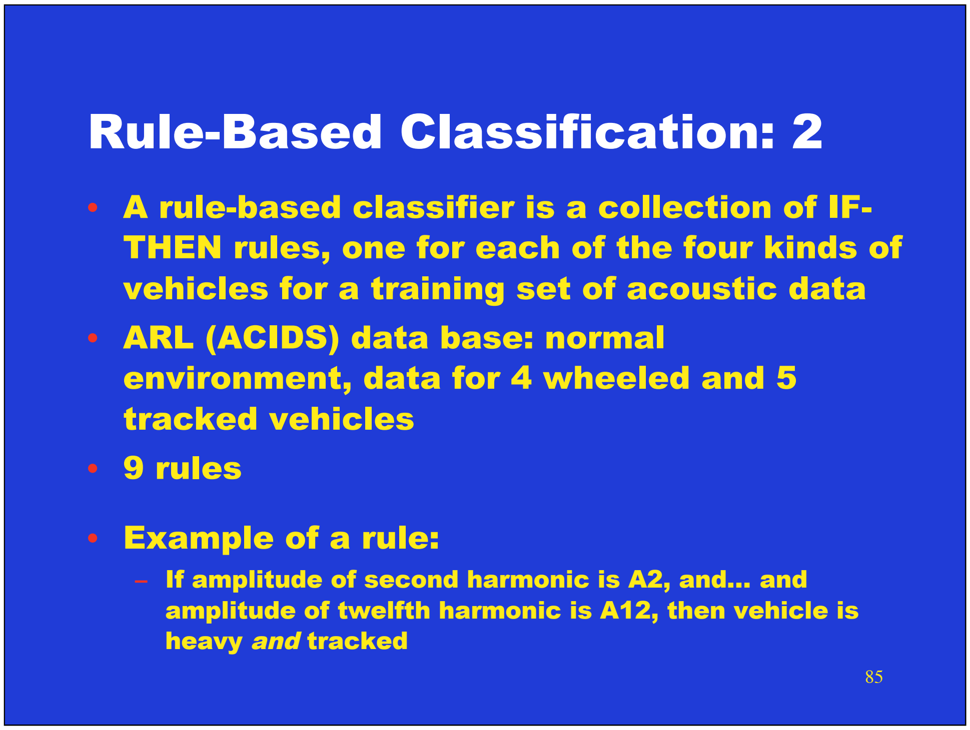
Take another look at the figure on the previous slide as well as the one on this slide. Observe that the interval type-2 FLS achieves close to its optimal performance almost at the first epoch of training. This shows that the interval type-2 FLS (as compared to the type-1 FLS) is very promising for real-time signal processing where more than one epoch of training is not possible.



This is a pattern recognition application.

Features (to be described) are extracted from acoustic measurements (also to be described).

Those features activate a fuzzy logic rule-based classifier (FL RBC) that classifies them into one of the four kinds of ground vehicles (e.g., trucks, tanks) that are listed on this slide.



ARL: Army Research Laboratory

ACIDS: Acoustic-Seismic Classification/Identification Data Set

Normal terrain: a grassy terrain (data is also available in ACIDS for a desert terrain and two arctic terrains)

Our approach was to use one rule for each of the nine vehicles for which data had been collected, so our FL RBC used nine rules.

We used the same features that others had used in prior studies, namely the amplitudes of the second through the 12th harmonics of a one-second window of acoustic data.

This means that each of our nine rules, an example of which is shown on this slide, had 11 antecedents.

Rule-Based Classification: 3

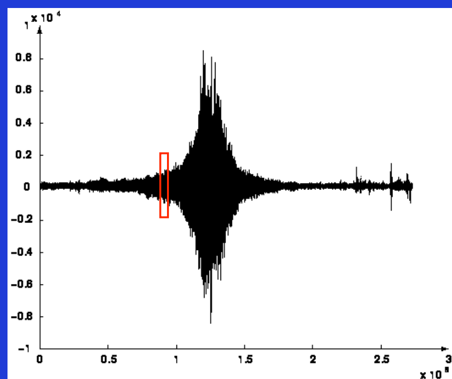
- **Causes of uncertainties in features:**
  - **Terrain conditions**
  - **Traveling speed**
  - **Signal-to-noise ratio**
  - **Windowing of measurements during feature extraction**
- **Uncertainties change with time, i.e. they are non-stationary**
- **Exact temporal behaviors of uncertainties unknown ahead of time**

86

There are four causes of uncertainties in the features, and they are listed on this slide.

For discussions about these uncertainties, see: H. Wu and J. M. Mendel, "Multi-Category Classification of Ground Vehicles Using Fuzzy Logic Rule-Based Classifiers: Early Results," *Proc. of the 7th IASTED Int'l. Conf. Artificial Intelligence and Soft Computing*, Banff, Canada, July 2003, pp. 52-57.

Rule-Based Classification: 4



- One-second window of data is used to extract features

- Typical measurement
- As vehicle gets close to three-element microphone array, amplitude of signal increases
- Maximum amplitude occurs at "closest point of approach" (CPA)

87

Data was collected using a three-element triangular array of microphones that were placed on the ground.

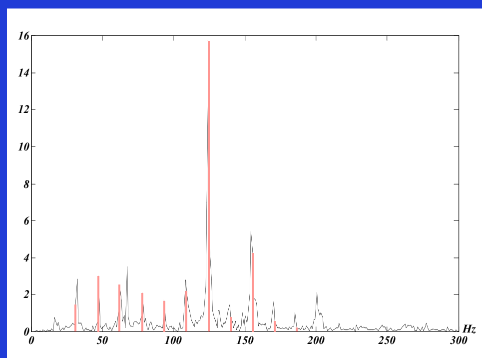
In this figure we see a recording of sound amplitude as a vehicle approaches a microphone and then moves away from it.

Recorded data length varied for each of the nine vehicles, as did the number of independent runs recorded for each vehicle.

The small red box denotes a one-second window of data. For an available run, there are many such windows of data. Each window was used to extract the 11 harmonic amplitude features.

Because we began with recorded data for which the CPA could be established, all windows of data were chosen with respect to the CPA.

Rule-Based Classification: 5

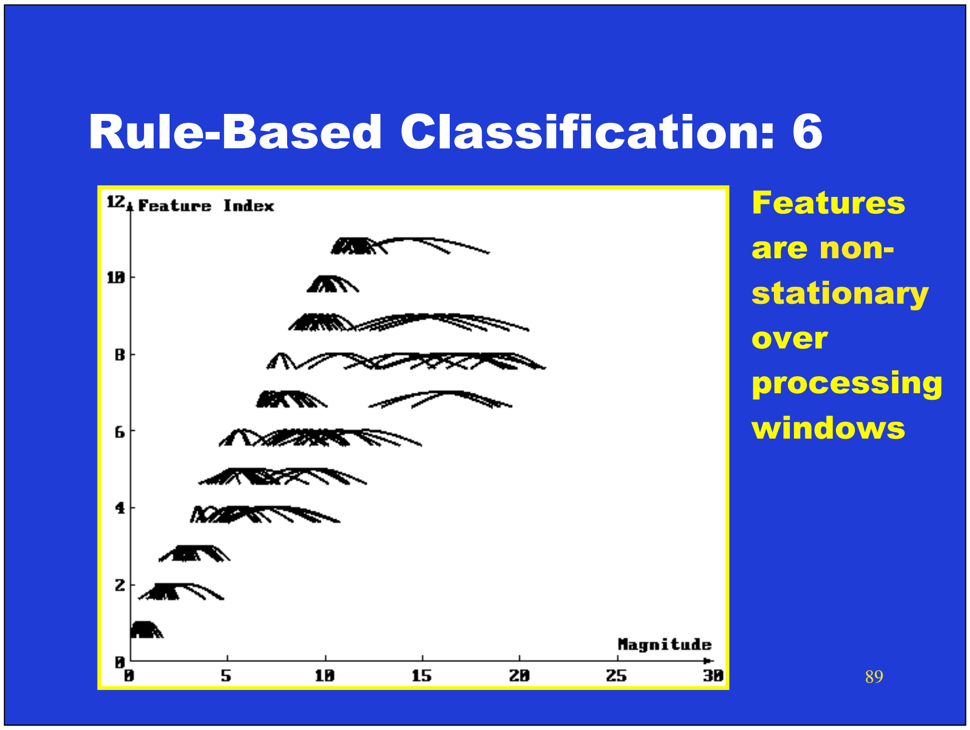


Features are amplitudes of second through twelfth harmonics

88

Shown in this figure is the FFT of a one-second window of data (black) and the 11 harmonic amplitudes that have been "picked" off of it.

The Harmonic Line Association Algorithm for doing this is explained in: M. C. Wellman, N. Srour and D. B. Hills, "Acoustic Feature Extraction for a Neural Network Classifier," Army Research Laboratory Report, ARL-TR-1166, Jan. 1997.



Shown in this slide are the 11 feature-distributions for one of the nine vehicles.

Each curve corresponds to a feature distribution within one of the vehicle's 11 available runs.

Each curve is characterized by a Gaussian function that is centered at the mean value for that feature (remember, each run has many one-second windows for which each feature has been extracted),

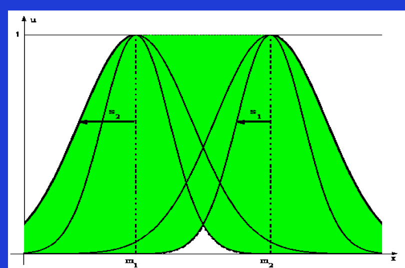
Each Gaussian is shown extending ±1 standard deviation about the mean value. The standard deviation was also computed from the collection of feature values computed for all of the one-second windows for a specific run.,

The reason I am showing this figure is to make it clear that all of the 11 features vary, some not so much, but most quite a bit.

It is the variability of the classifier features that motivates the use of a type-2 FL RBC.

Rule-Based Classification: 7

- **Based on extracting 11 features from real data for the 9 vehicles, we established that the uncertainty for the mean and standard deviation (SD) of variations of all features are of same order of magnitude**
- **So, the FOU for each feature must allow for simultaneous variations of mean and SD, e.g**



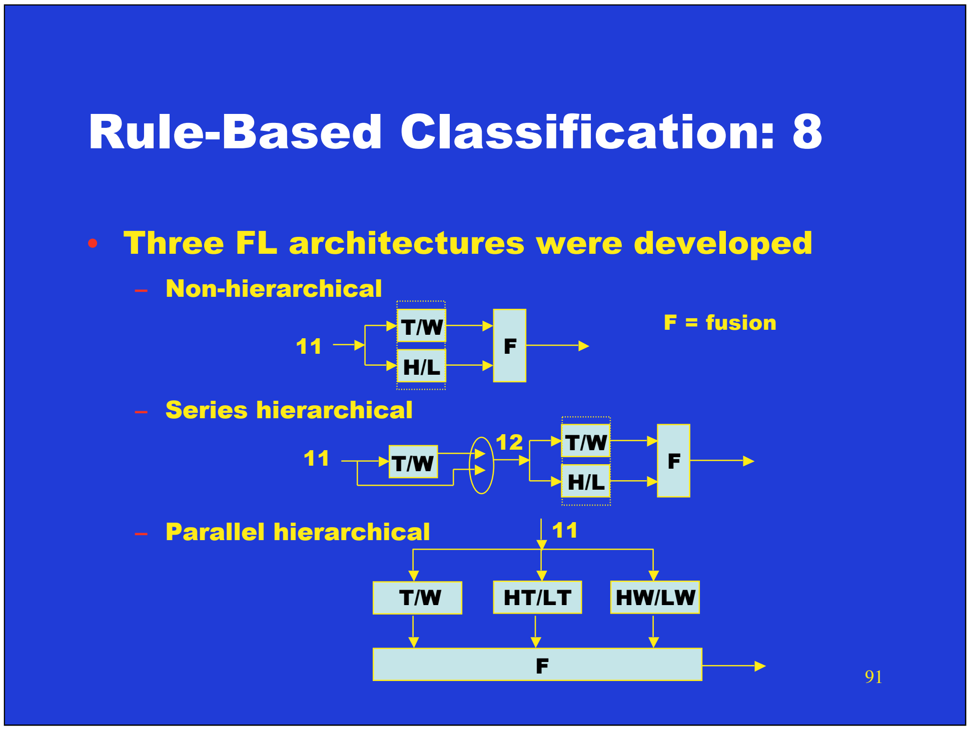
- **Gaussian primary MF with uncertain mean and standard deviation**
- **Each MF is characterized by four parameters**

90

To see that there is about the same order of magnitude of variability for both the mean and standard deviation of each feature, see the previous slide.

The green FOU is for a Gaussian primary MF that has both an uncertain mean and an uncertain standard deviation.

The four MF (FOU) parameters are: the two end-points for the uncertain mean and the two end-points for the uncertain standard deviation.



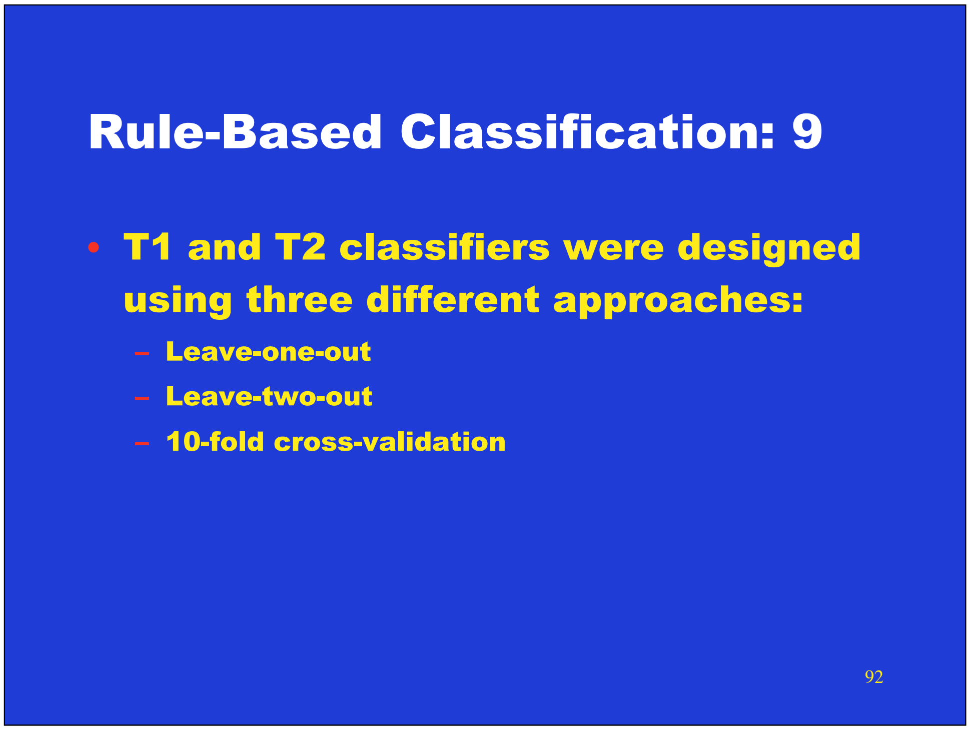
T: Tracked vehicle, W: Wheeled vehicle, H: Heavy vehicle, L: Light vehicle

The **non-hierarchical** classifier is comprised of two binary classifiers (T/W and H/L, each with 11 antecedent rules) whose outputs are then fused to provide a final classification.

The **series hierarchical** classifier is comprised first of a binary classifier for T/W (with 11 antecedent rules), and then of two other binary classifiers (T/W and H/L, each with 12 antecedent rules, one of which is the output of the first binary classifier). The outputs of the last two binary classifiers are then fused to provide a final classification.

The parallel; hierarchical classifier is comprised of three binary classifiers (T/W, HT/LT, HW/LW, each with 11 antecedent rules) whose outputs are then fused to provide a final classification.

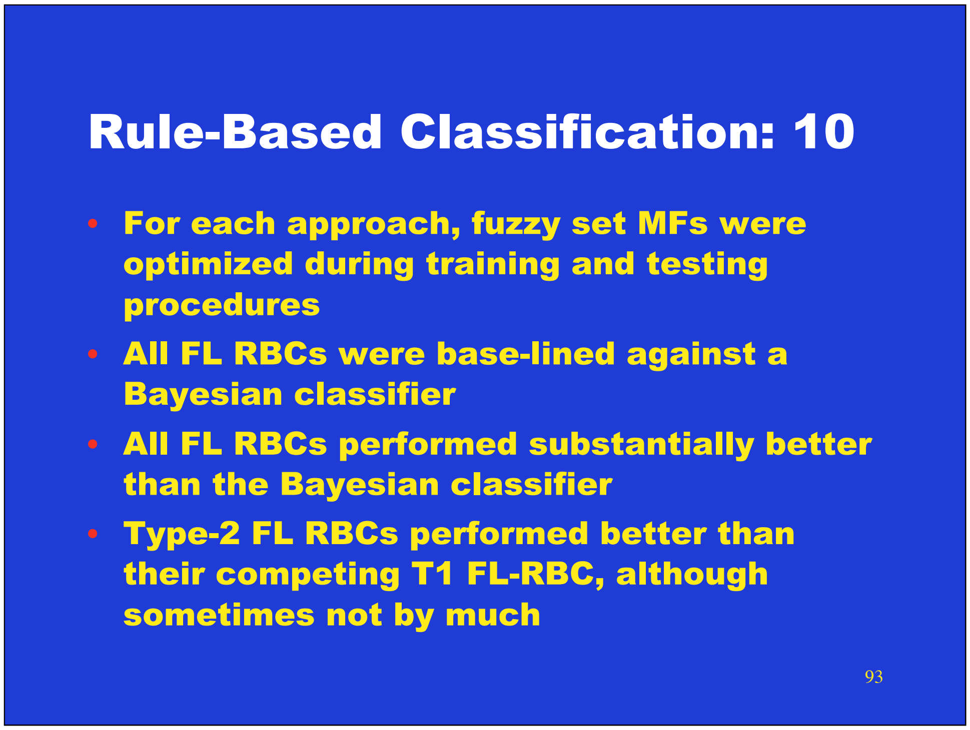
For details about these classifiers, see: H. Wu and J. M. Mendel, "Multi-Category Classification of Ground Vehicles Based on Their Acoustic Emissions," *Proc. SPIE Conf. on Unattended/Unmanned Ground, Ocean, and Air Sensor Technologies VI*, part of Defense and Security Conf., Orlando. FL, April 2004.



Leave-one-out: This is a design and testing method where each classifier within a collection of classifiers is designed by leaving one vehicle run out for testing. Performance (percent of correct classification) is averaged across all of the designs. There were 89 such designs.

Leave-two-out: This is a design and testing method where each classifier within a collection of classifiers is designed by leaving two vehicle runs out for testing. Performance (percent of correct classification) is averaged across all of the designs. We used 200 such designs.

10-fold cross-validation: This is a design and testing method where all of the available one-second windows of data are grouped into 10 sets. Ten classifiers are designed by using nine of the 10 groups for training and the one left-out group for testing. Performance (percent of correct classification) is averaged across the 10 designs.



The Bayesian classifier is a standard one, without any bells or whistles, since the FL RBCs also do not have any bells or whistles. See the article referred to in the notes for slide 91 for a description of this classifier.

See the next slide for a table that compares the different classifier performances. The last two statements on the present slide are based on the performance numbers on that slide.

In a 2005 study, where we designed one classifier to operate in any of four environments, the additional uncertainties about not being able to specify a specific environment ahead of time introduced so much extra uncertainty that the type-2 FL RBCs showed significant performance improvements over the type-1 FL RBCs. The following is a reference for the 2005 study: H. Wu and J. M. Mendel, "Multi-Category Classification of Ground Vehicles Based on the Acoustic Data of Multiple Terrains Using Fuzzy Logic Rule-Based Classifiers," *Unattended Ground Vehicle Technologies and Applications VII, Proc. of SPIE*, vol. 5796. Pp. 28-39, 2005.

Rule-Based Classification: 11

Classifier	Avg. Error	SD of Error
Bayesian	27.87%	0.262%
Non-H T1	6.95%	0.081%
Non-H T2	**3.19%**	**0.045%**
H-P T1	5.28%	0.053%
H-P T2	4.93%	0.054%
H-S T1	3.83%	0.047%
H-S T2	3.24%	0.041%

H: Hierarchical, P: Parallel, S: Series

H: Hierarchical

P: Parallel

S: Series

The results that are shown an this slide are for the leave-one-out designs. Comparable results were obtained for the leave-two-out and 10-fold cross-validation designs.

The best results are highlighted in red.

A computational analysis of the three kinds of classifiers revealed that the non-hierarchical classifier was much simpler to implement than the others; hence, based on its performance and computational complexity it is the classifier of choice.

Part VI. Conclusions

95

Conclusions: 1

- When we use fuzzy sets in engineering applications of rule-based systems, the words used in the rules are a means to an end
- In such applications, we use the mathematical description of the FLS and tune the FS MFs using “input-output” training data, which are quite different data than are MF data
- There is nothing wrong with doing this!
- An important role exists for interval T2 FSs in engineering applications, because such FSs can model uncertainties

96

We distinguish using fuzzy sets for engineering applications, where we optimize MF parameters using application-based training data, from using fuzzy sets to "compute with words," where MF parameters must be determined from word-data that are collected from people.

Both kinds of applications can make good use of interval type-2 fuzzy sets in order to directly model and minimize the effect of uncertainties.

Conclusions: 2

- **On the other hand, if the new field of computing with words is really about using rules whose words mean something to people, then we need to extract MF data from people**
- **Because words mean different things to different people, I have demonstrated that we need to use interval T2 FSs to model them**

97

The first paper about computing with words is: L. A. Zadeh, "Fuzzy Logic = Computing With Words,"*IEEE Trans. on Fuzzy Systems*, vol. 4, pp. 103-111, 1996.

Since that paper, many papers and books have been published on this subject.

Conclusions: 3

- **How to collect person MF data from people is an open issue**
- **Collecting word interval data from people is manageable**
- **How to go from such data to a FOU is an interesting "inverse" problem—Fuzzistics**

Part VII. References

99

References

- J. M. Mendel, ***Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions***, Prentice-Hall, NJ, 2001
- I have also created a bibliography for T2 FSs and FLSs
- It can be accessed at my web-site:
  – http://sipi.usc.edu/~mendel
- Free software is on-line
  – http://sipi.usc.edu/~mendel/software

100

Another entry-level article about type-2 FSs is: J. M. Mendel, "Type-2 Fuzzy Sets: Some Questions and Answers," *IEEE Connections*, *Newsletter of the IEEE Neural Networks Society*, vol. 1, Aug. 2003, pp. 10-13.