Global
Top 100
University

The University of Nottingham
UNITED KINGDOM · CHINA · MALAYSIA

ASAP research group

horizon DIGITAL ECONOMY RESEARCH

ima Intelligent Modelling & Analysis

Tutorial on Type-2 Fuzzy Sets and Systems
WCCI 2016, Vancouver

Jon Garibaldi, Robert John and Christian Wagner

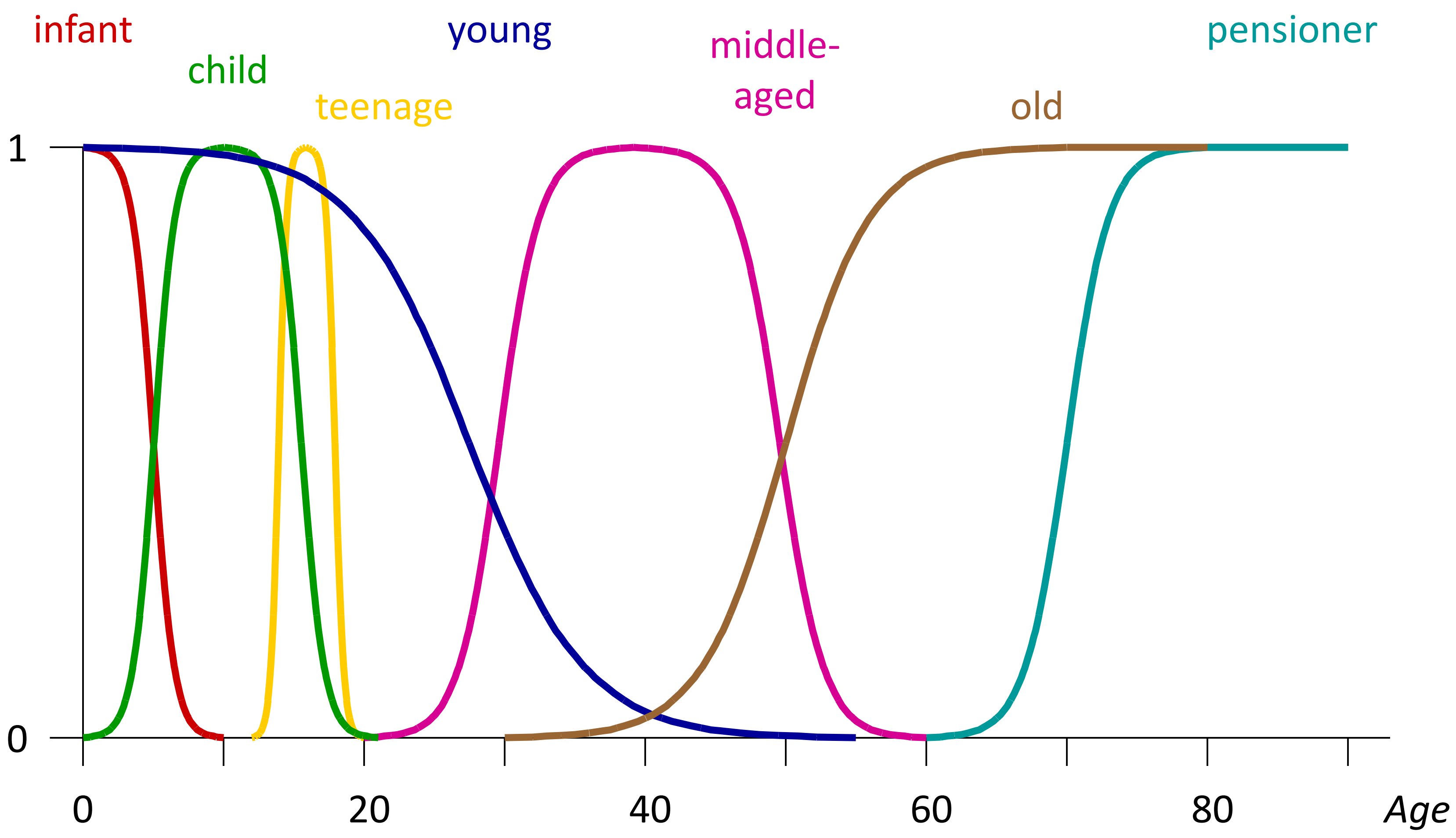
LUCID
Lab for
Uncertainty in Data
and Decision Making

http://lucidresearch.org

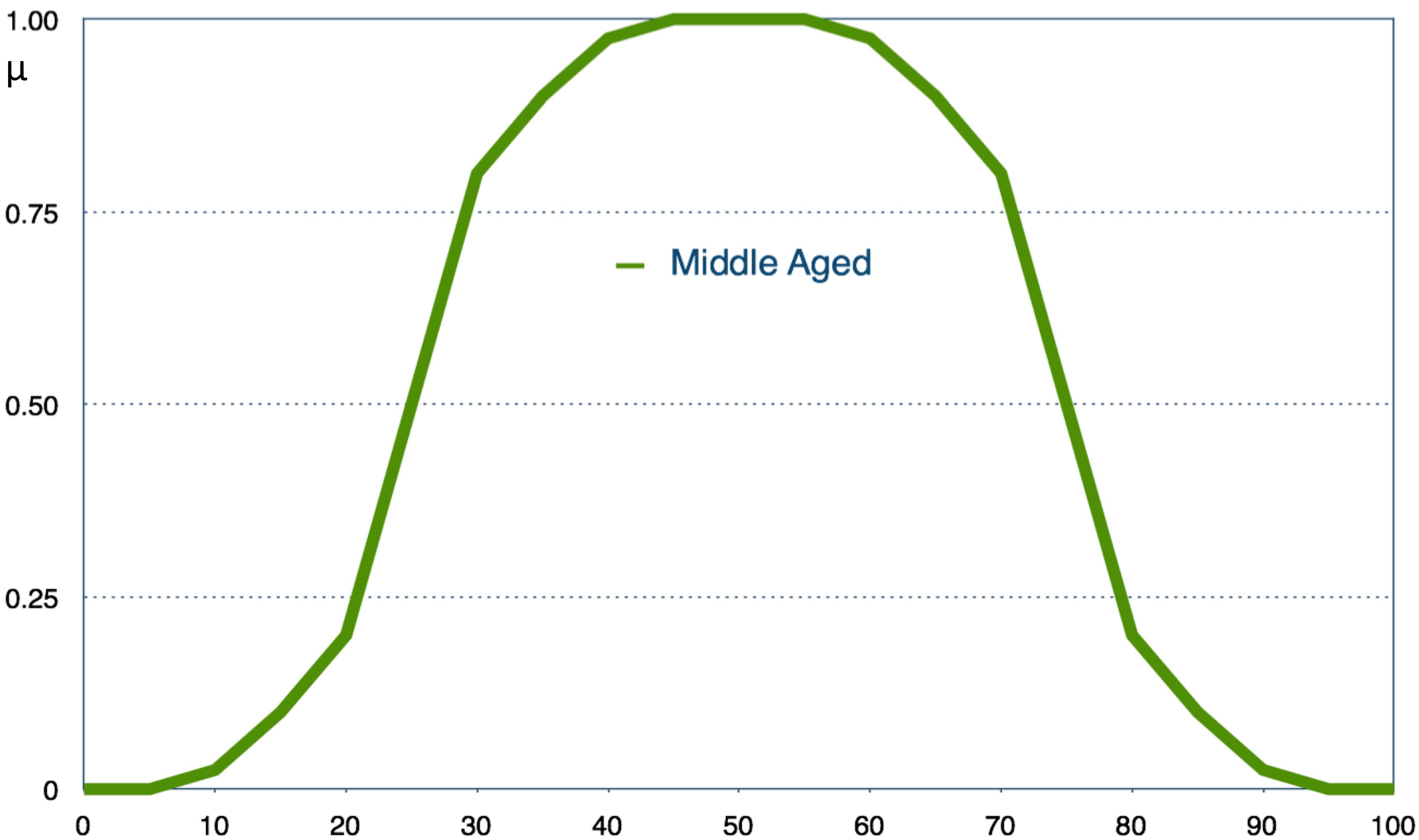
Outline

- Type-2 Fuzzy sets – what are they?
- Interval Type-2
  - Fuzzy Sets
  - Inference
  - Real World Example
- General Type-2
  - Fuzzy Sets
  - Inference
  - Computational Aspects
  - Control Example
  - Design you own Type-2 FLSs
- Constructing Type-2 Fuzzy Sets from Data
  - Building Type-2 FSs from Survey Data
  - Application Example

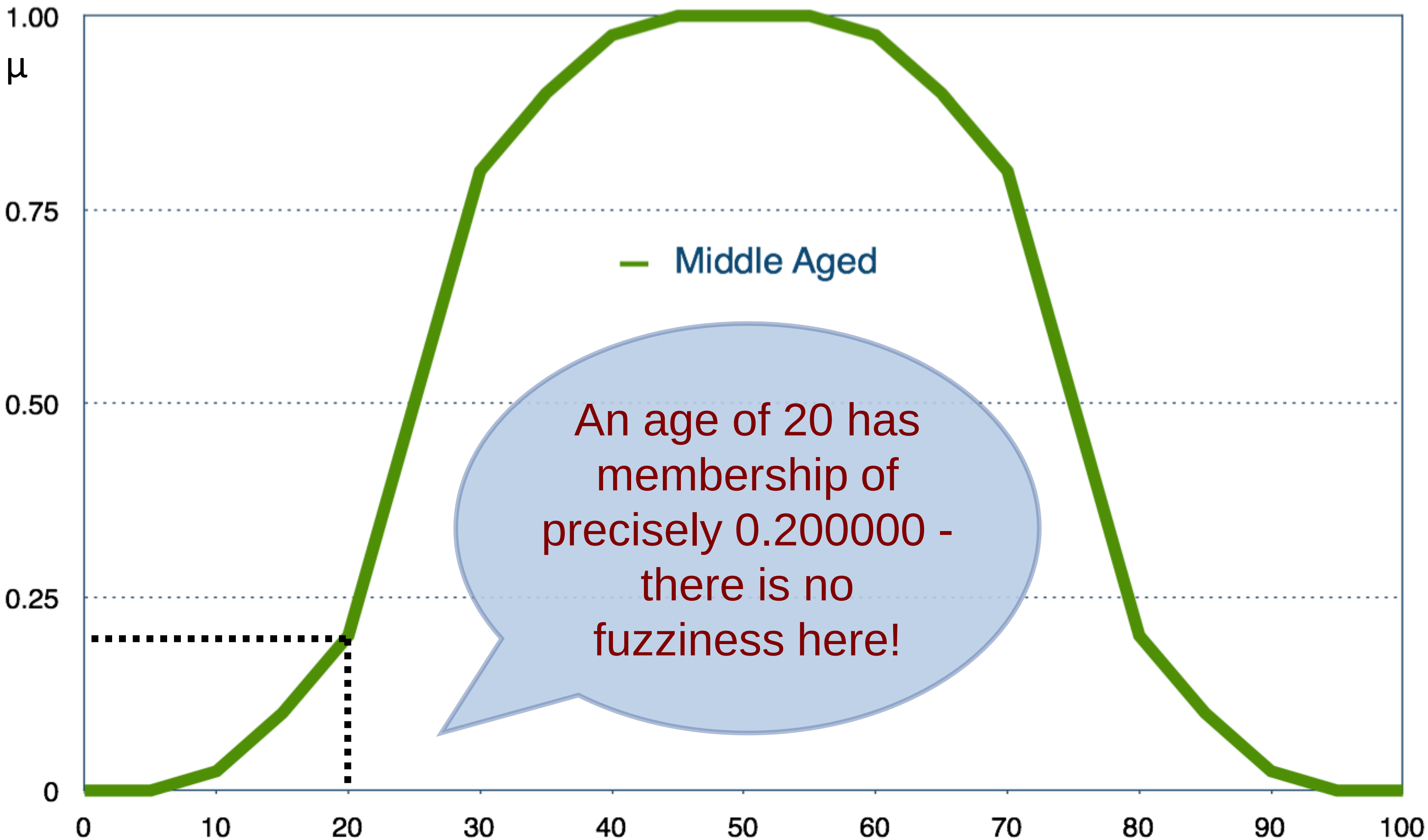
Linguistic Variable - *Age*



Standard (Type-1) Fuzzy Set

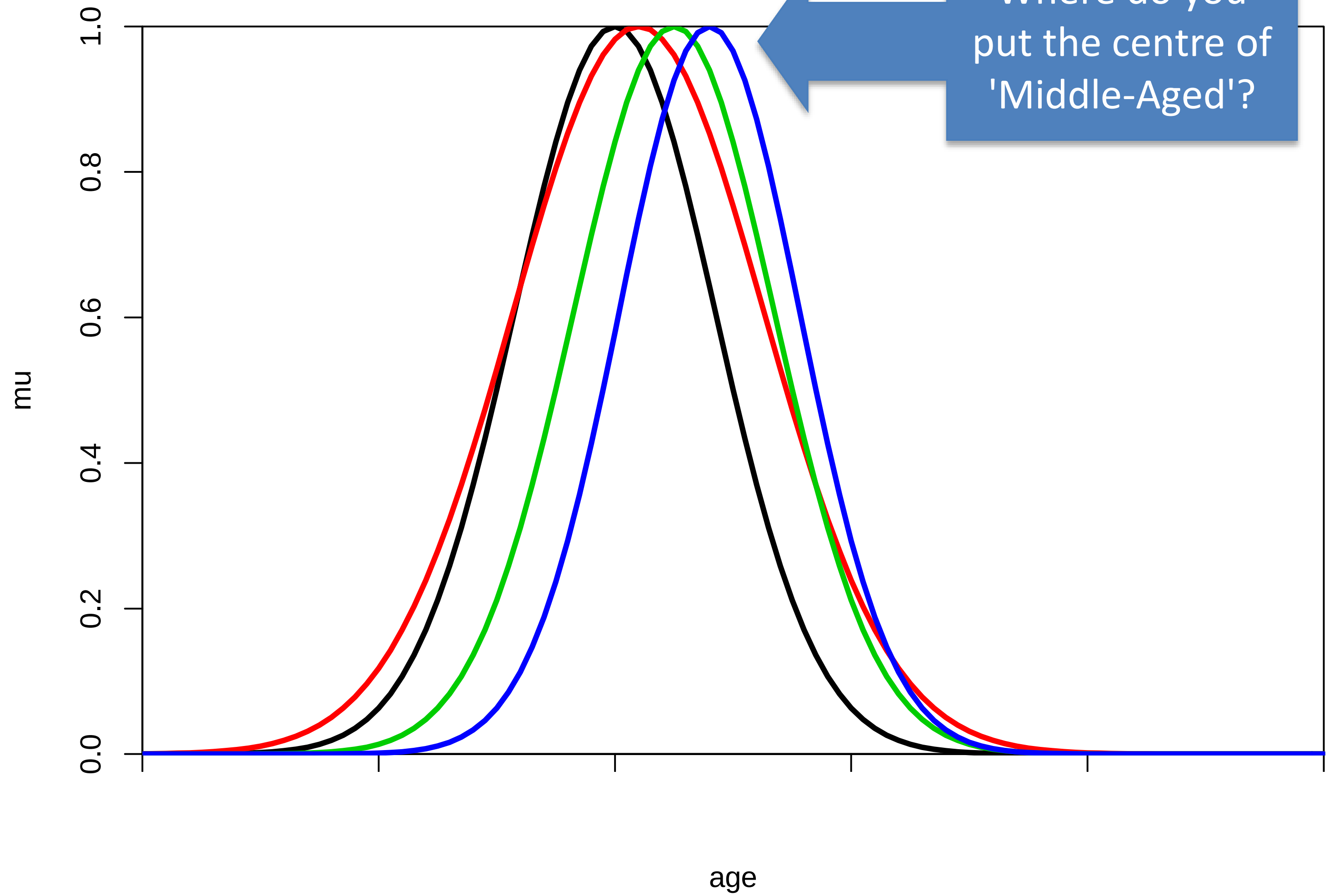


However …

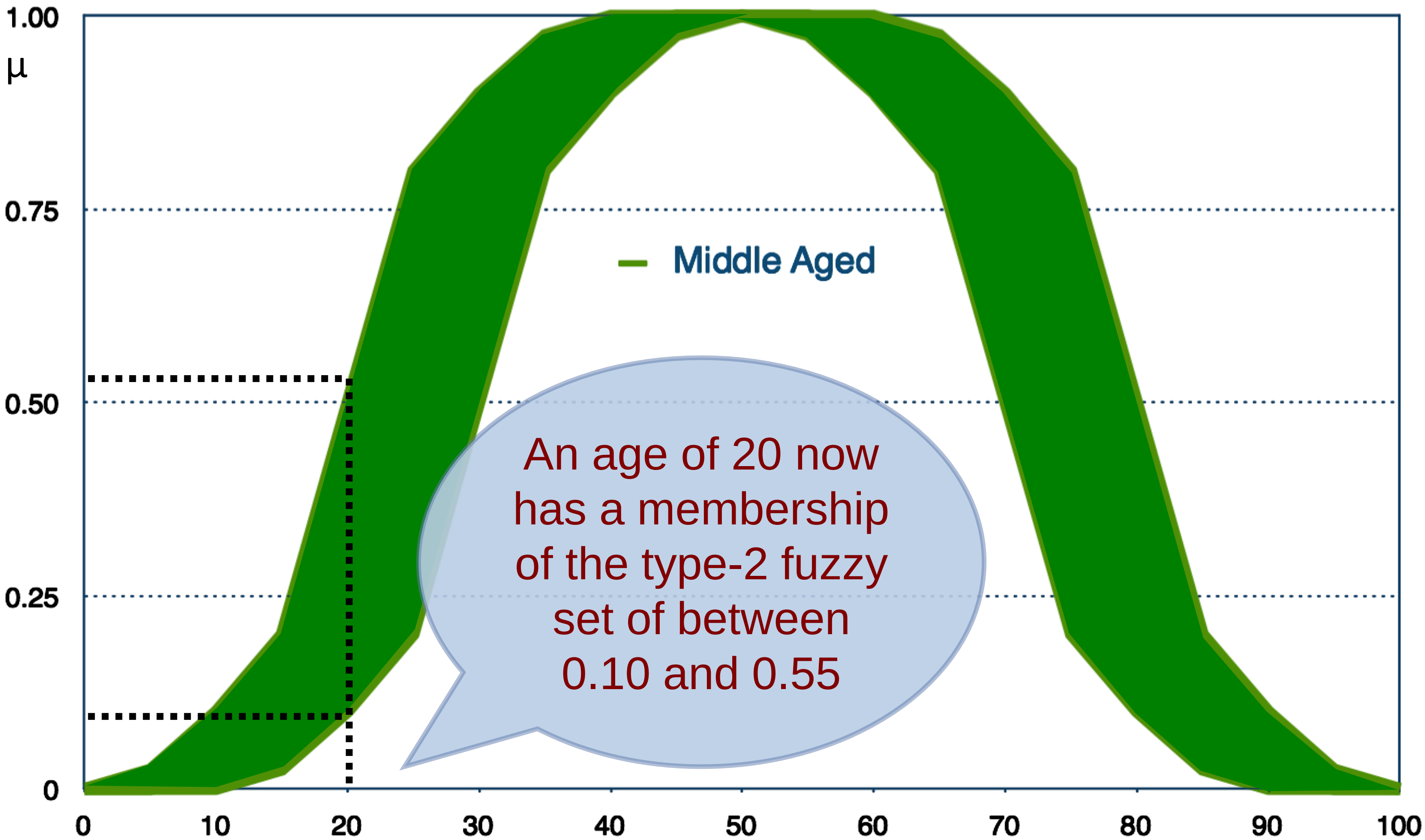

1.00
μ
0.75
0.50
0.25
0
0
10
20
30
40
50
60
70
80
90
100
Middle Aged
An age of 20 has membership of precisely 0.200000 - there is no fuzziness here!

Exercise – ‘Middle-Aged’?

Where do you put the centre of 'Middle-Aged'?



Type-2 Fuzzy Sets



Basics on Interval Type-2

J.M. Mendel, R.I. John, F.Liu,
"Interval Type-2 Fuzzy Logic Systems Made Simple",
IEEE Transactions Fuzzy Systems, 14(6):808-21, 2006

Basic Definitions

Definition 1: A *T2 FS*, denoted $\tilde{A}$, is characterized by a *type-2 MF* $\mu_{\tilde{A}}(x, u)$, where $x \in X$ and $u \in J_x \subseteq [0, 1]$, i.e.,

$$\tilde{A} = \{((x, u), \mu_{\tilde{A}}(x, u)) \mid \forall\, x \in X,\ \forall\, u \in J_x \subseteq [0, 1]\} \quad (1)$$

in which $0 \leq \mu_{\tilde{A}}(x, u) \leq 1$. $\tilde{A}$ can also be expressed as

$$\tilde{A} = \int_{x \in X} \int_{u \in J_x} \mu_{\tilde{A}}(x, u)/(x, u) \quad J_x \subseteq [0, 1] \qquad (2)$$

where $\int \int$ denotes union[1] over all admissible x and u. For discrete universes of discourse, $\int$ is replaced by $\sum$. ■

Basic Definitions

Definition 2: When all $\mu_{\tilde{A}}(x,u) = 1$ then A is an *interval T2 FS* (IT2 FS). ■

$$\tilde{A} = \int_{x \in X} \int_{u \in J_x} 1/(x,u), \qquad J_x \subseteq [0,1]. \tag{3}$$

Example Discrete IT2 Set

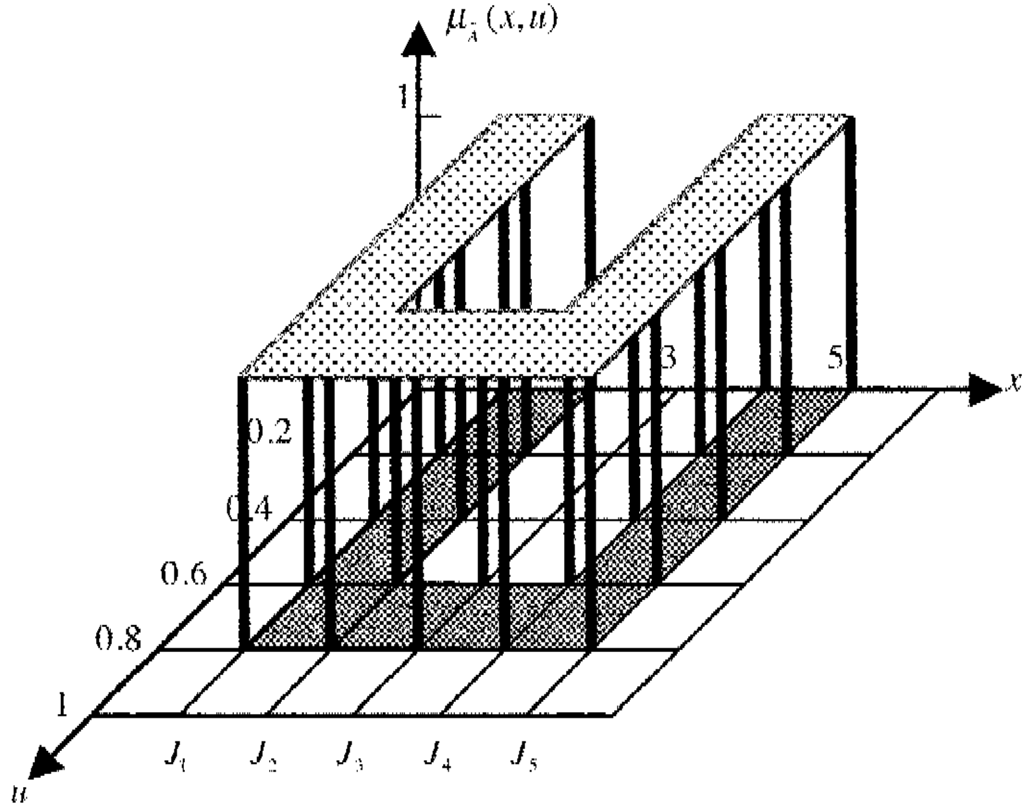
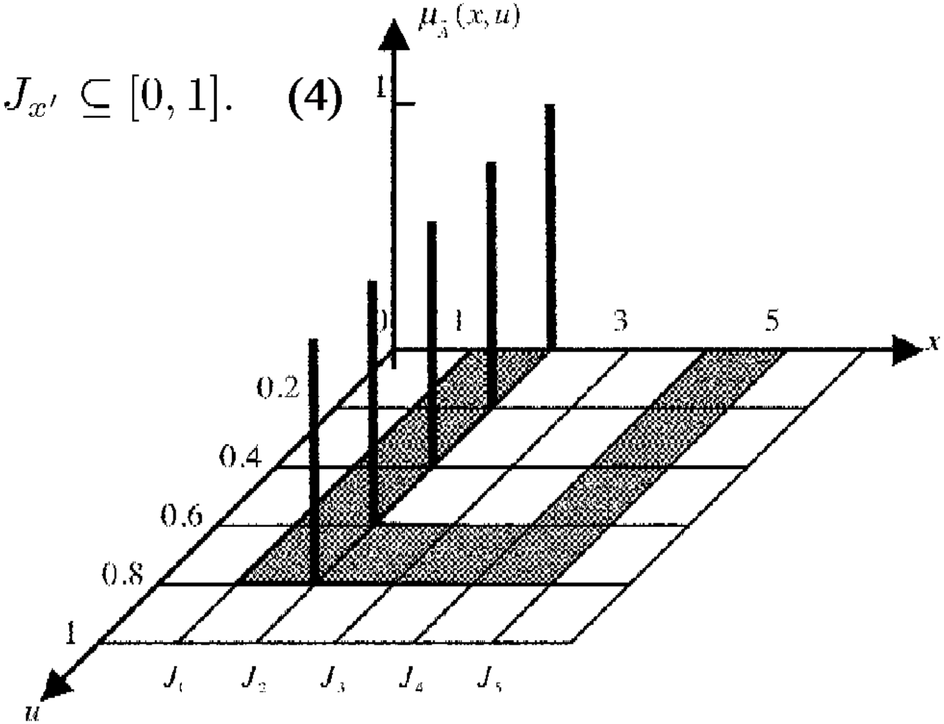


Fig. 2. Example of an interval T2 MF for discrete universes of discourse. The shaded area in the $x - u$ plane is the FOU.

Vertical Slice

Definition 3: At each value of x, say $x = x'$, the 2-D plane whose axes are u and $\mu_{\tilde{A}}(x', u)$ is called a *vertical slice* of $\mu_{\tilde{A}}(x, u)$. A *secondary MF* is a vertical slice of $\mu_{\tilde{A}}(x, u)$. It is $\mu_{\tilde{A}}(x = x', u)$ for $x' \in X$ and $\forall\, u \in J_{x'} \subseteq [0, 1]$, i.e.,

$$\mu_{\tilde{A}}(x = x', u) \equiv \mu_{\tilde{A}}(x') = \int_{u \in J_{x'}} 1/u \quad J_{x'} \subseteq [0, 1]. \quad (4)$$



FOU

Definition 6: Uncertainty in the primary memberships of an IT2 FS, $\tilde{A}$, consists of a bounded region that we call the *footprint of uncertainty* (FOU). It is the union of all primary memberships, i.e.,

$$\mathrm{FOU}(\tilde{A}) = \bigcup_{x \in X} J_x. \tag{8}$$

This is a *vertical-slice representation of the FOU*, because each of the primary memberships is a vertical slice. ■

Upper and Lower MF

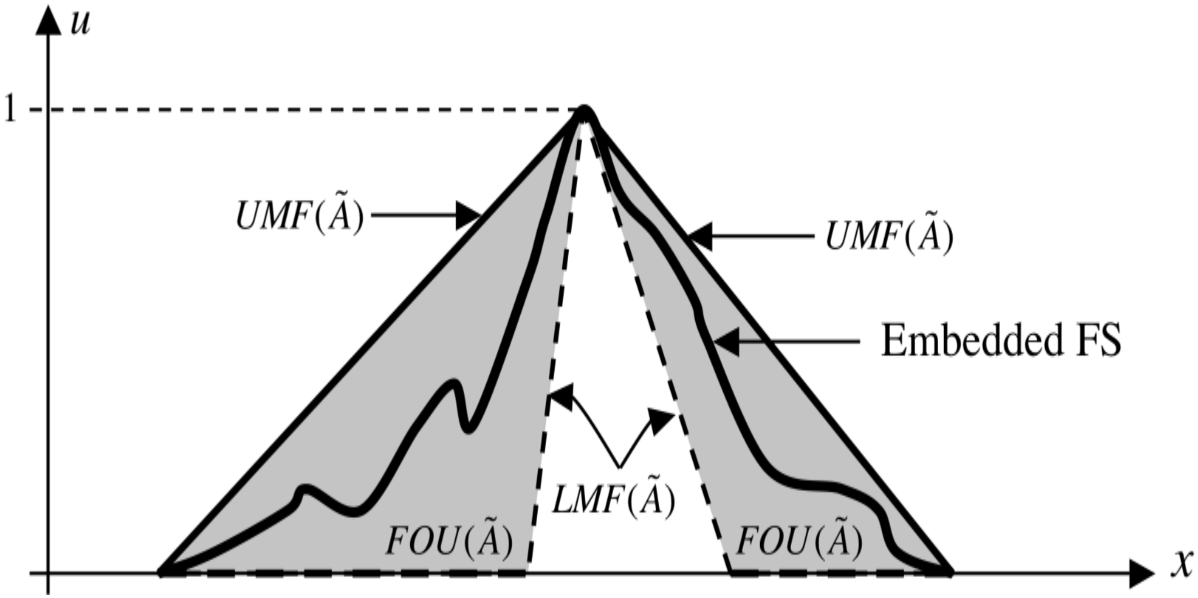
Definition 7: The *upper membership function* (UMF) and *lower membership function* (LMF) of $\tilde{A}$ are two T1 MFs that bound the FOU (e.g., see Fig. 4). The UMF is associated with the upper bound of $\mathrm{FOU}(\tilde{A})$ and is denoted $\bar{\mu}_{\tilde{A}}(x), \forall\, x \in X$, and the LMF is associated with the lower bound of $\mathrm{FOU}(\tilde{A})$ and is denoted $\underline{\mu}_{\tilde{A}}(x), \forall\, x \in X$, i.e.,

$$\bar{\mu}_{\tilde{A}}(x) \equiv \overline{\mathrm{FOU}(\tilde{A})} \qquad \forall\, x \in X \tag{9}$$

$$\underline{\mu}_{\tilde{A}}(x) \equiv \underline{\mathrm{FOU}(\tilde{A})} \qquad \forall\, x \in X. \tag{10}$$

$\bar{\mu}_{\tilde{A}}(x) =$

Concepts Illustrated



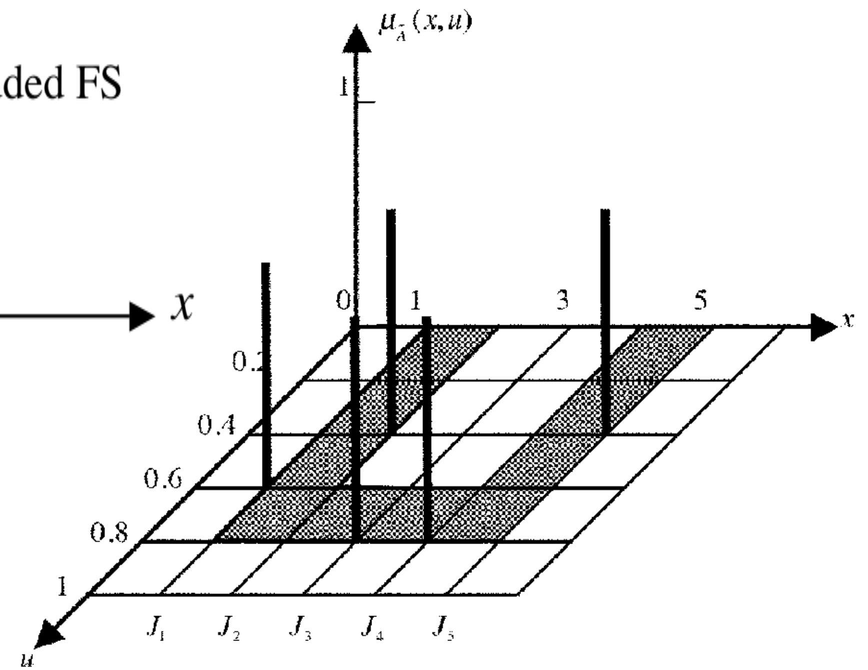


Fig. 5. Example of an embedded IT2 FS associated with the T2 MF depicted in Fig. 2.

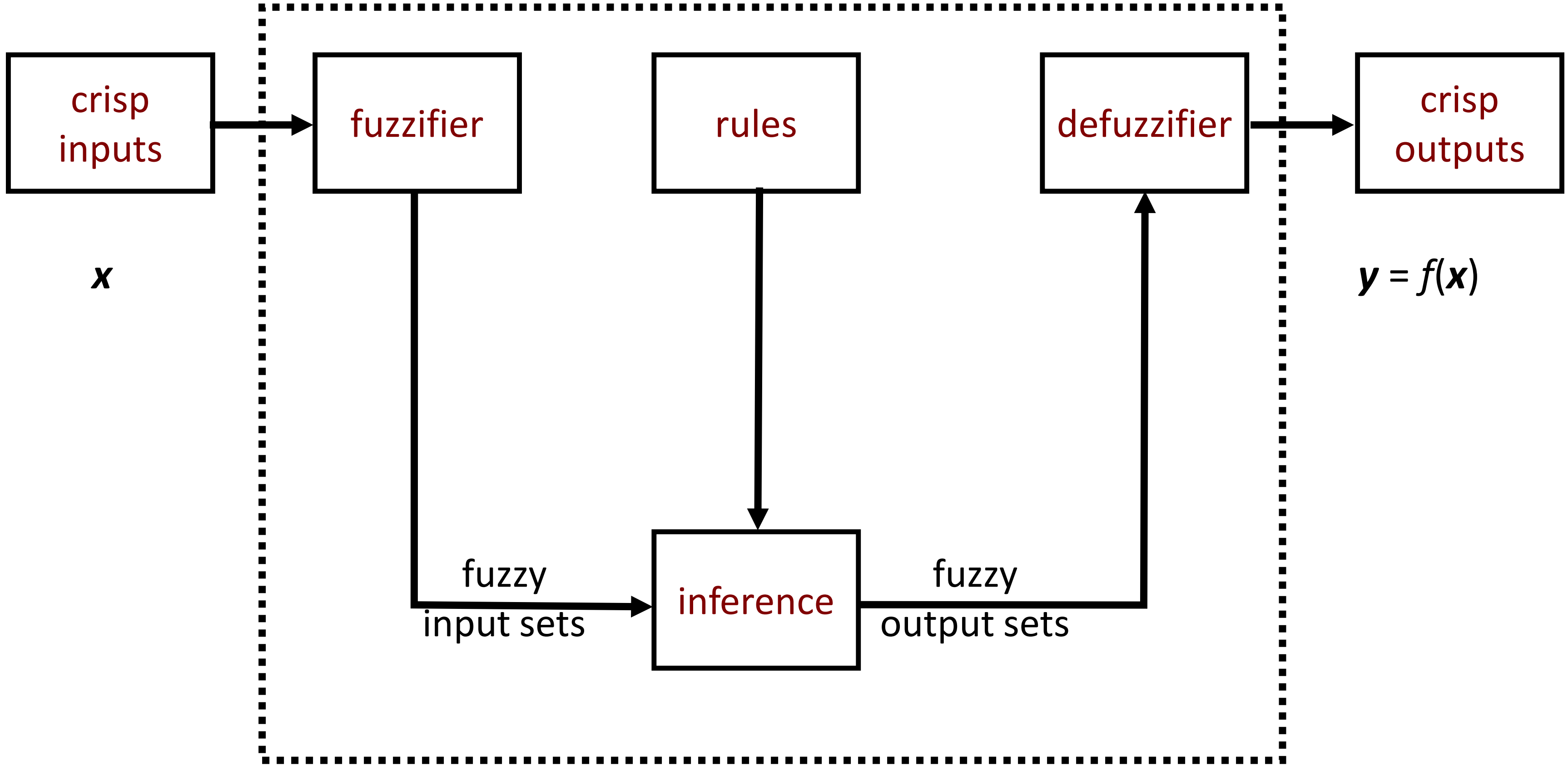
Representing Type-2 Sets

- Vertical slice representation
  - union of all vertical slices
- Horizontal slice representation
  - union of all alpha planes (‘horizontal slices’)
- Wavy slice representation
  - union of all embedded sets (‘wavy slices’)
- zSliced based representation
  - used for general type-2: next lecture

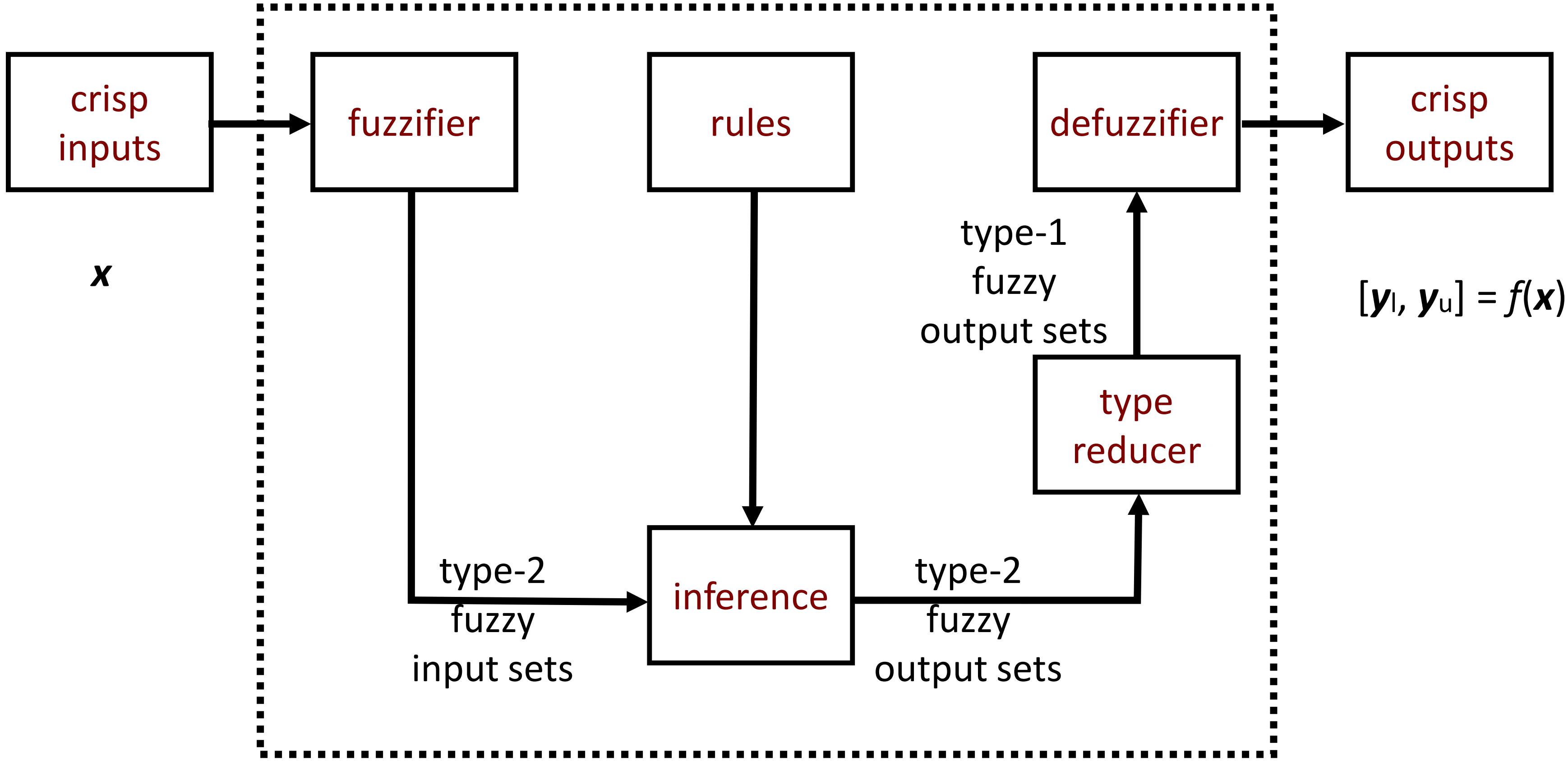
Interval Type-2 Inference

Some material is taken from:
J.M. Mendel
"Type-2 Fuzzy Sets and Systems: An Overview"
Computational Intelligence Magazine
2(1):20-29, 2007

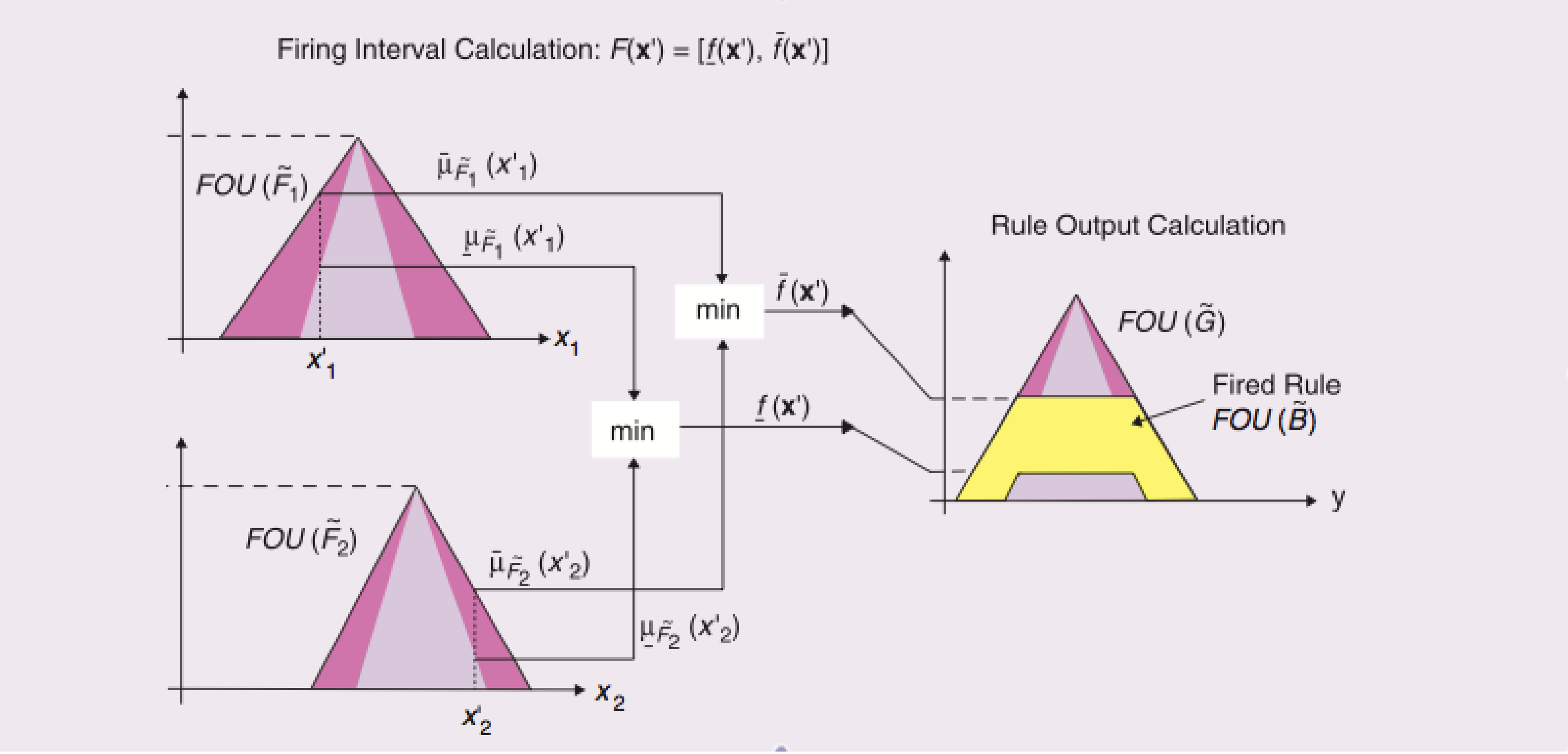
Type-1 Fuzzy Inference Systems



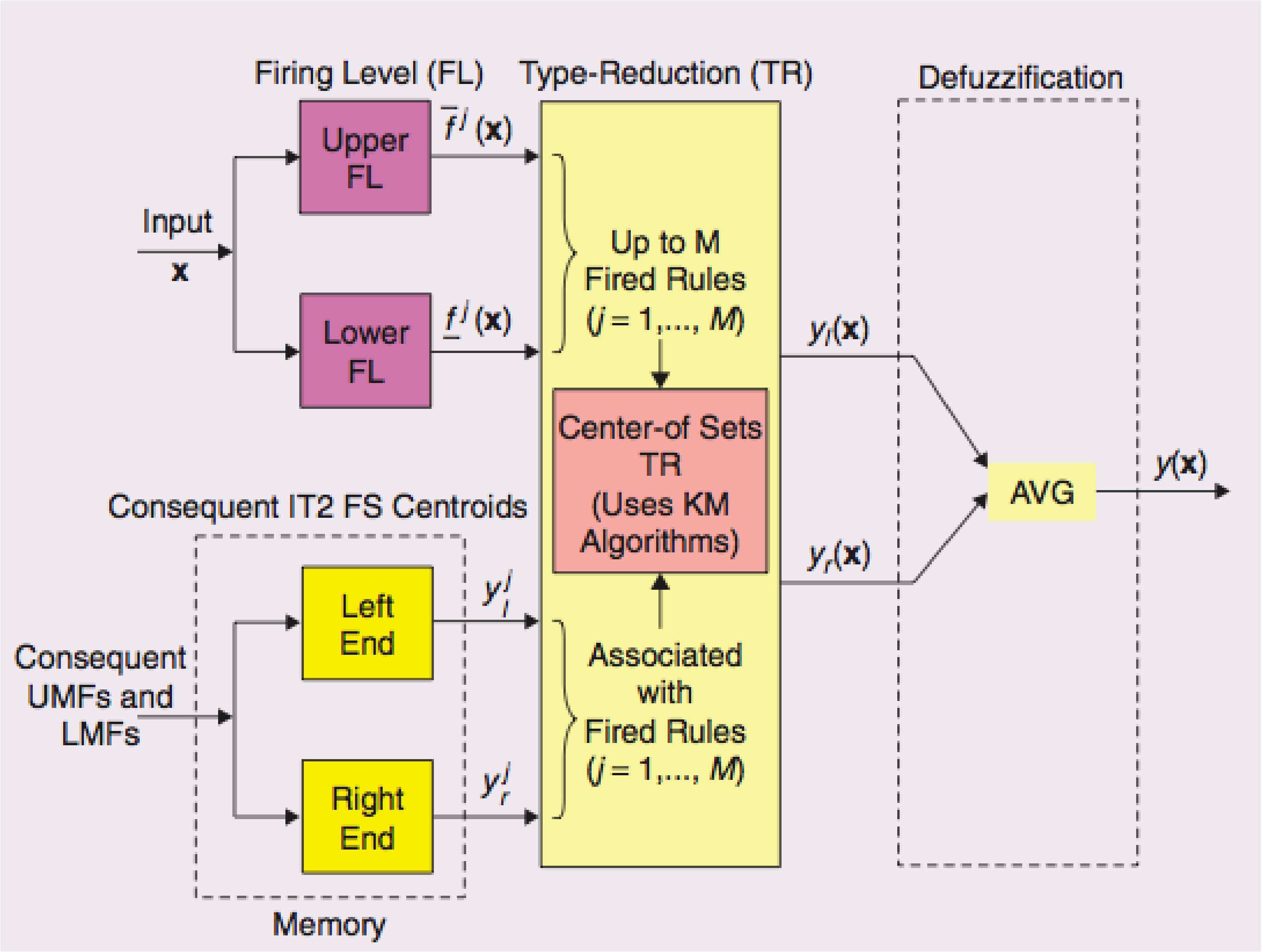
Type-2 Fuzzy Inference Systems



Single Rule Inference



Rule Combination



Rule Combination

Center-of-Sets (COS) Defuzzification, $y_{cos}(\mathbf{x})$

1. Compute the centroid of each rule's consequent T1 FS. Call it c^l ($l = 1, ..., M$)
2. Compute the firing level for each (fired) rule. Call it f^l ($l = 1, ..., M$)
3. Compute $y_{\cos}(\mathbf{x}) = \sum_{l=1}^{M} c^l f^l \Big/ \sum_{l=1}^{M} f^l$

Center-of-Sets (COS) TR, $Y_{cos}(\mathbf{x})$

1. Compute the centroid of each rule's consequent IT2 FS, using the KM algorithms (see Box 6). Call it $[y_l^l, y_r^l]$ ($l = 1, ..., M$)
2. Compute the firing interval for each (fired) rule. Call it $\left[\underline{f}^l, \bar{f}^l\right]$ ($l = 1, ..., M$)
3. Compute $Y_{\cos}(\mathbf{x}) = [y_l(\mathbf{x}), y_r(\mathbf{x})]$, where $y_l(\mathbf{x})$ is the solution to the following minimization problem, $y_l(\mathbf{x}) = \min_{\forall f^l \in [\underline{f}^l, \bar{f}^l]} \left[\sum_{l=1}^{M} y_l^l f^l \Big/ \sum_{l=1}^{M} f^l \right]$, that is solved using a KM Algorithm (Box 6), and $y_r(\mathbf{x})$ is the solution to the following maximization problem, $y_r(\mathbf{x}) = \max_{\forall f^l \in [\underline{f}^l, \bar{f}^l]} \left[\sum_{l=1}^{M} y_r^l f^l \Big/ \sum_{l=1}^{M} f^l \right]$, that is solved using the other KM Algorithm (Box 6).

Type Reduction

Consider the FOU shown in Figure 8. Using the RT, compute the centroids of all of its embedded T1 FSs, examples of which are shown as the colored functions. Because each of the centroids is a finite number, this set of calculations leads to a set of centroids that is called the centroid of $\tilde{B}$, $C(\tilde{B})$. $C(\tilde{B})$ has a smallest value c_l and a largest value c_r, i.e. $C(\tilde{B}) = [c_l(\tilde{B}), c_r(\tilde{B})]$. So, to compute $C(\tilde{B})$ it is only necessary to compute c_l and c_r. It is not possible to do this in closed form. Instead, it is possible to compute c_l and c_r using two iterative algorithms that are called the *Karnik-Mendel (KM) algorithms*.

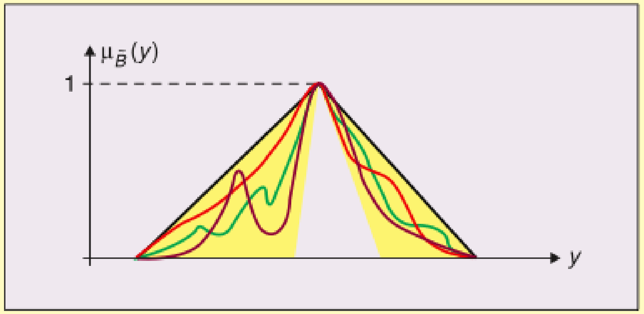


FIGURE 8 FOU and some embedded T1 FSs.

Note that c_l = min (centroid of all embedded T1 FSs in $FOU(\tilde{B})$). Analysis shows that:

$$c_l = c_l(L) = \frac{\sum_{i=1}^{L} y_i UMF(\tilde{B}|y_i) + \sum_{i=L+1}^{N} y_i LMF(\tilde{B}|y_i)}{\sum_{i=1}^{L} UMF(\tilde{B}|y_i) + \sum_{i=L+1}^{N} LMF(\tilde{B}|y_i)}$$

One of the KM algorithms computes switch point L (see Figure 9). Note also that c_r = max (centroid of all embedded T1 FSs in $FOU(\tilde{B})$). Analysis shows that:

$$c_r = c_r(R) = \frac{\sum_{i=1}^{R} y_i LMF(\tilde{B}|y_i) + \sum_{i=R+1}^{N} y_i UMF(\tilde{B}|y_i)}{\sum_{i=1}^{R} LMF(\tilde{B}|y_i) + \sum_{i=R+1}^{N} UMF(\tilde{B}|y_i)}$$

The other KM algorithm computes switch point R (see Figure 10).

Derivations and statements of the KM algorithms are found, e.g., in [2, pp. 204–207] and [5, pp. 258–259 or pp. 308–311].

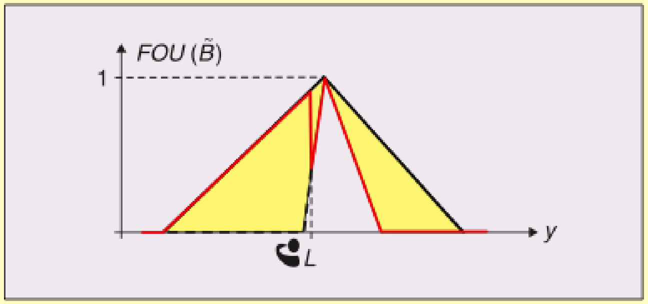


FIGURE 9 The red embedded T1 FS is used to compute $c_l(L)$.

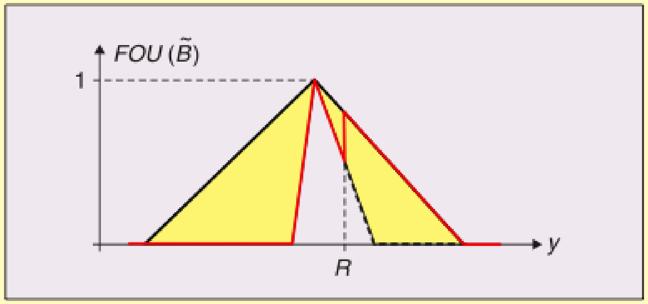


FIGURE 10 The red embedded T1 FS is used to compute $c_r(R)$.

Interval Type-2 Fuzzy Logic and Supply Chain Management

S Miller, M Gongora, JM Garibaldi, RI John

"Interval type-2 fuzzy modelling and stochastic search for real-world inventory management"

Soft Computing, 16(8), 1447-1459, 2012

The Problem

- Managing the supply chain is (surprisingly still) difficult:
  - There are large amounts of uncertainties
  - There are conflicting objectives
  - Different stake holders have different needs

Supply Chain Management (SCM)

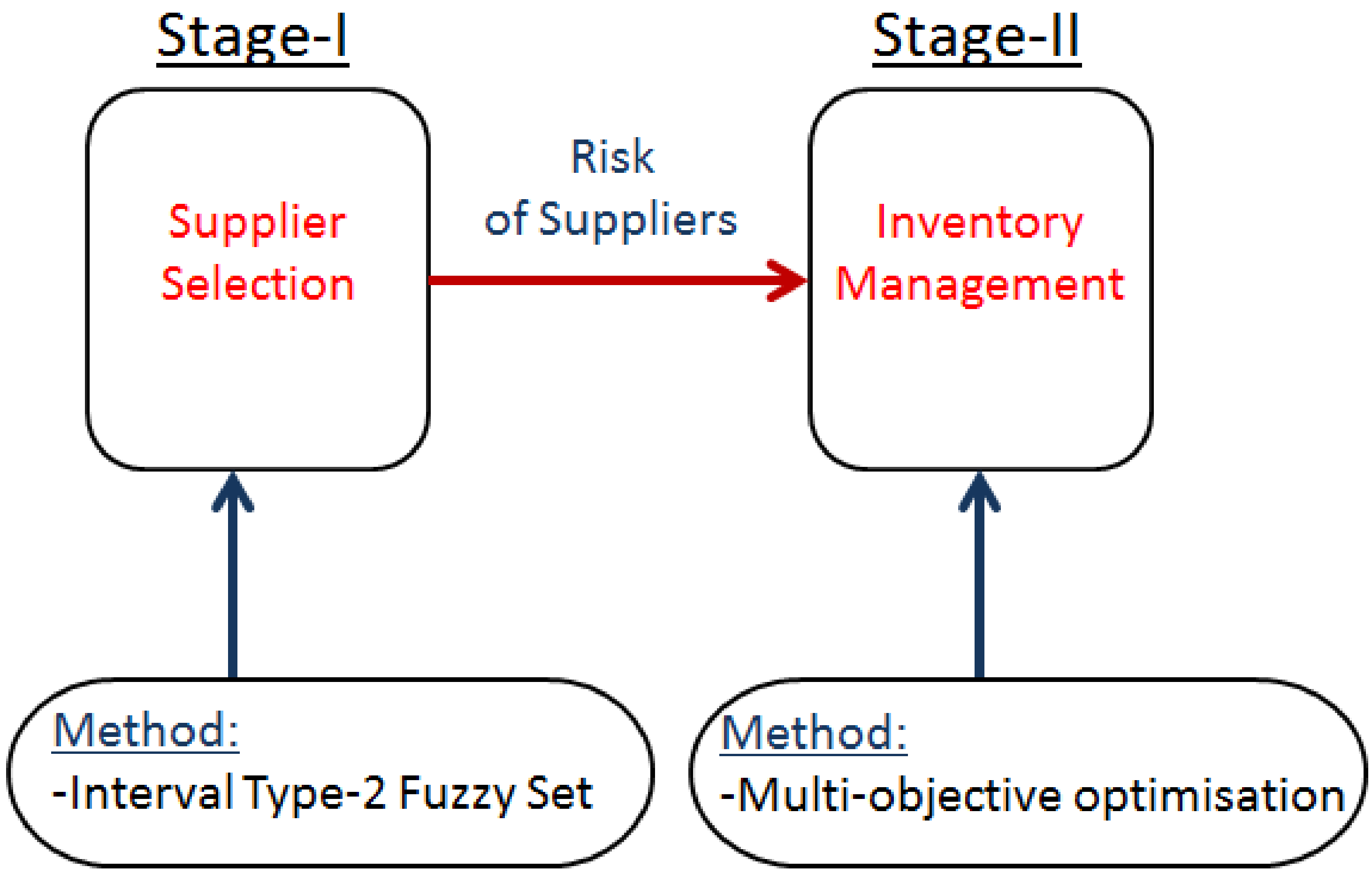
- SCM is the management of material flows from the procurement of basic raw materials to final product delivery considering;
  - information flows among whole processes of supply chains,
  - material flows,
  - long-term relations between customers and suppliers.

Typical Supply Chain

- A typical supply chain consists of five main components
  - Customers,
  - Retailers,
  - Wholesalers/Distributors,
  - Manufacturers,
  - Component/Raw material suppliers.

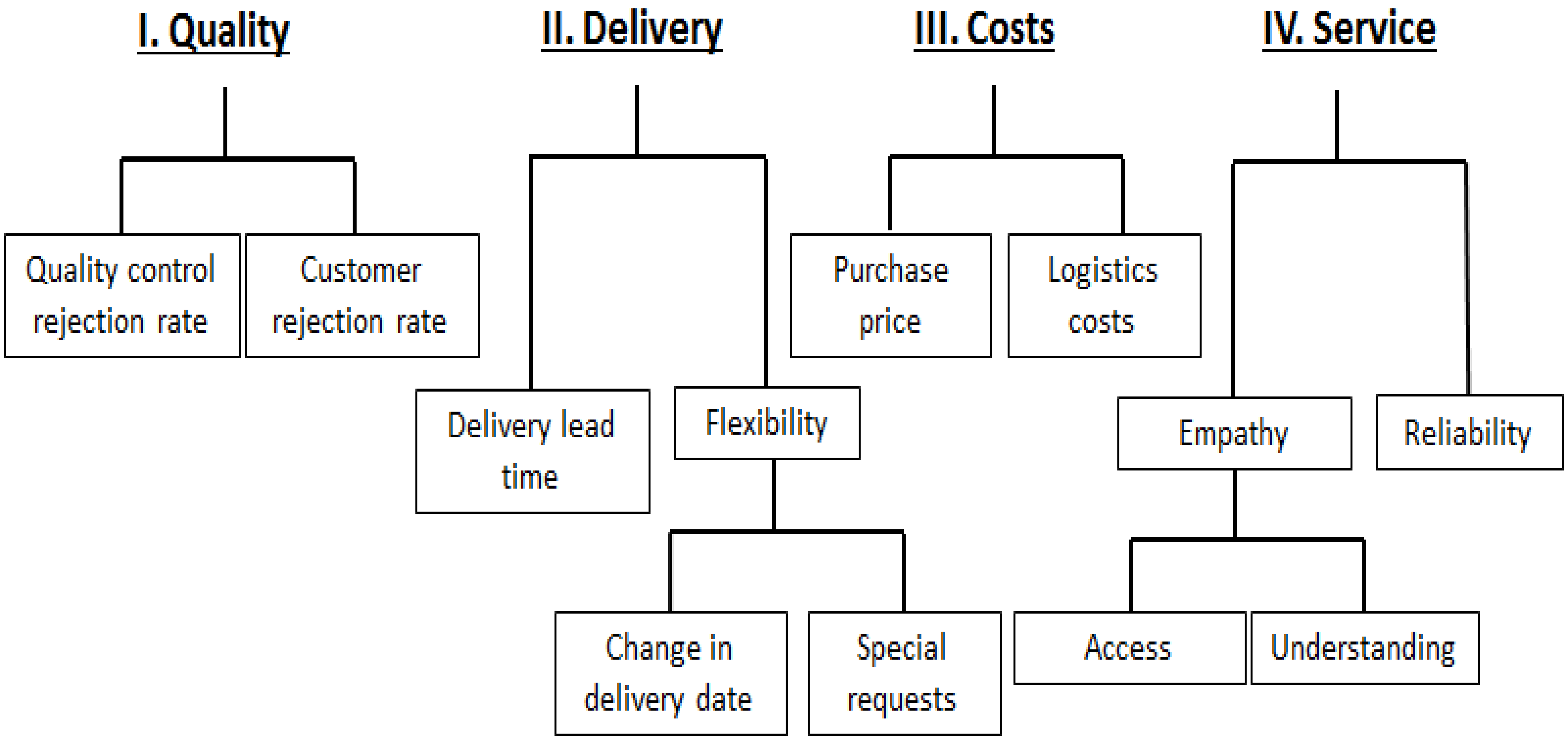
Inventory Management

- Inventory management is an integrated approach to plan and control inventory while considering the whole network from suppliers to end users.
- It is essential for good inventory management:
  – to avoid stock outs,
  – to manage surplus stock.
- Purpose of inventory management is to find out:}
  – How many units to order?
  – When to order?


Stage-I
Stage-II
Supplier Selection
Risk of Suppliers
Inventory Management
Method:
-Interval Type-2 Fuzzy Set
Method:
-Multi-objective optimisation

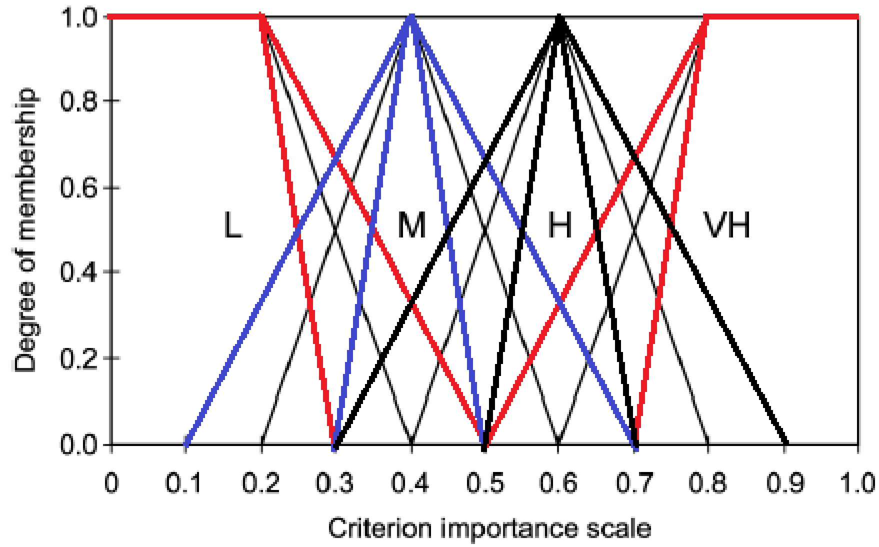
Stage-I: Ranking of Suppliers

- The decision makers selected the criteria relevant to the circumstance at hand from a list of criteria.
- They were evaluated in a linguistic way such as `low', `moderate', `high', `very high' to generate trapezoidal fuzzy sets for the importance of each criterion based on thoughts of the decision maker.
- The linguistic terms were converted into fuzzy weights using fuzzy membership functions.

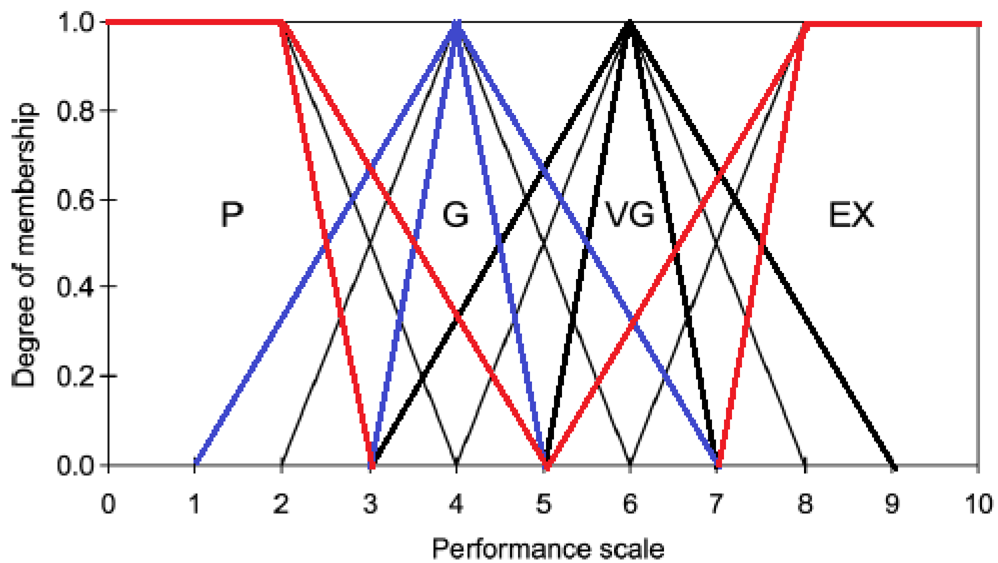

I. Quality
II. Delivery
III. Costs
IV. Service
Quality control rejection rate
Customer rejection rate
Delivery lead time
Flexibility
Change in delivery date
Special requests
Purchase price
Logistics costs
Empathy
Reliability
Access
Understanding

Method Summary

- Performances were determined in the same manner of the criteria using linguistic terms such as \`excellent', \`very good', \`good', \`poor'
  - the linguistic terms were converted into fuzzy weights using fuzzy membership functions
- The aggregate fuzzy scores for each supplier was calculated by aggregating all the pertinent criteria
  - each of them was converted into crisp scores using centroid type-reduction and defuzzification methods
- Suppliers were ranked according to their crisp scores
- The risk values are calculated based on their scores



1.0
0.8
0.6
0.4
0.2
0.0
Degree of membership
L
M
H
VH
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
Criterion importance scale



1.0
0.8
0.6
0.4
0.2
0.0
Degree of membership
P
G
VG
EX
0
1
2
3
4
5
6
7
8
9
10
Performance scale

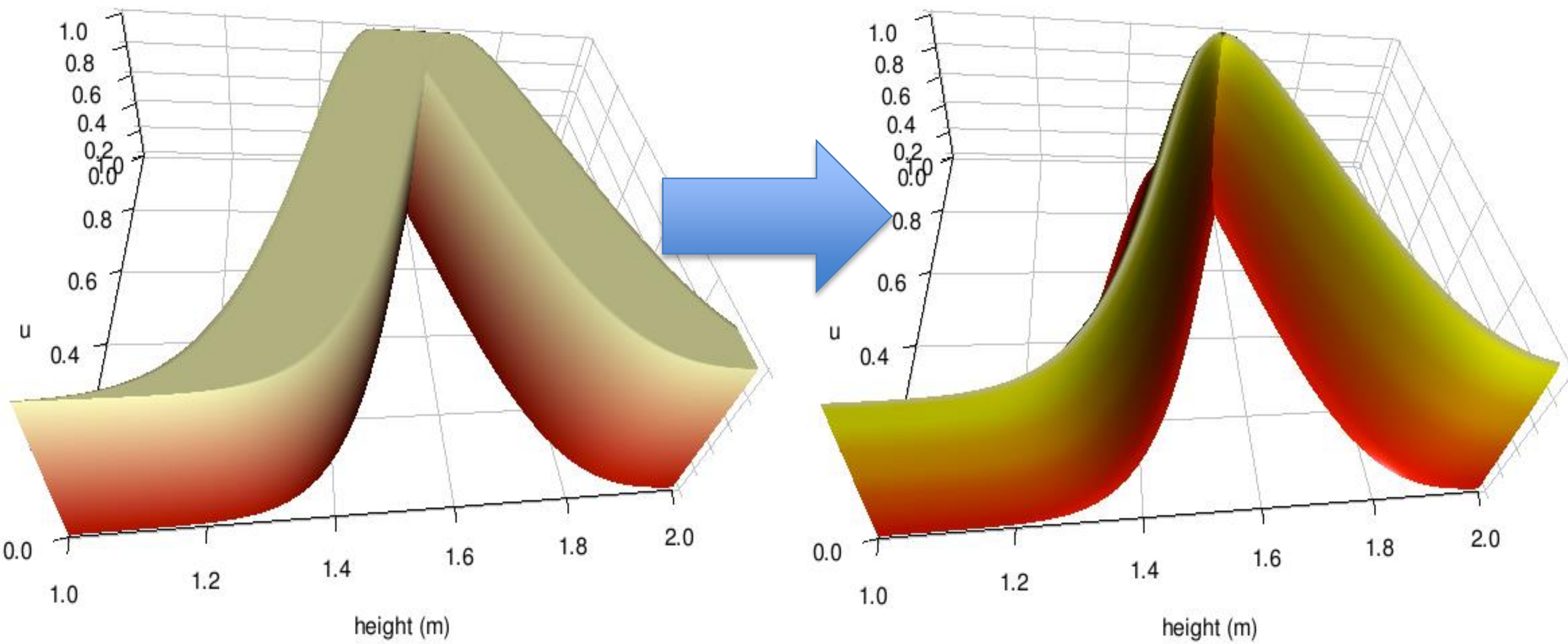
Inventory Planning with Consideration of Supplier Risk

- Several assumptions are made to determine the form of problem
- The problem is formulated defining objectives, constraints and decision variables
- Objectives are scaled into one objective using two scalarisation method
  - weighted sum and Tchebycheff approaches
- Six different scenario are generated using different weight settings

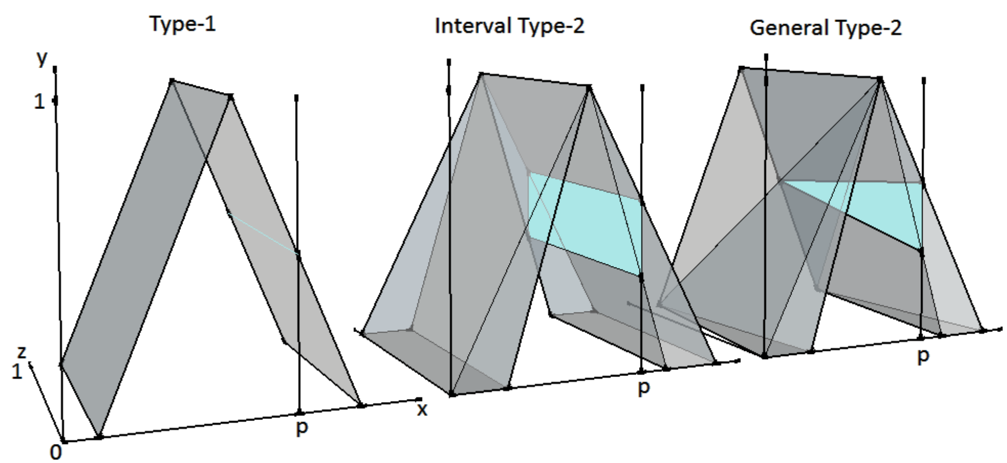
General Type-2 Fuzzy Sets

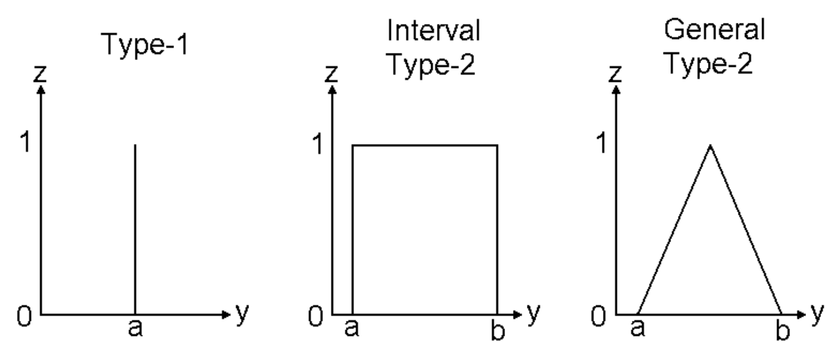
C.Wagner, H.Hagras,
"Towards General Type-2
Fuzzy Logic Systems based on zSlices",
IEEE Trans Fuzzy Systems, 18(4):637-60, 2010

General Type-2



Secondary View



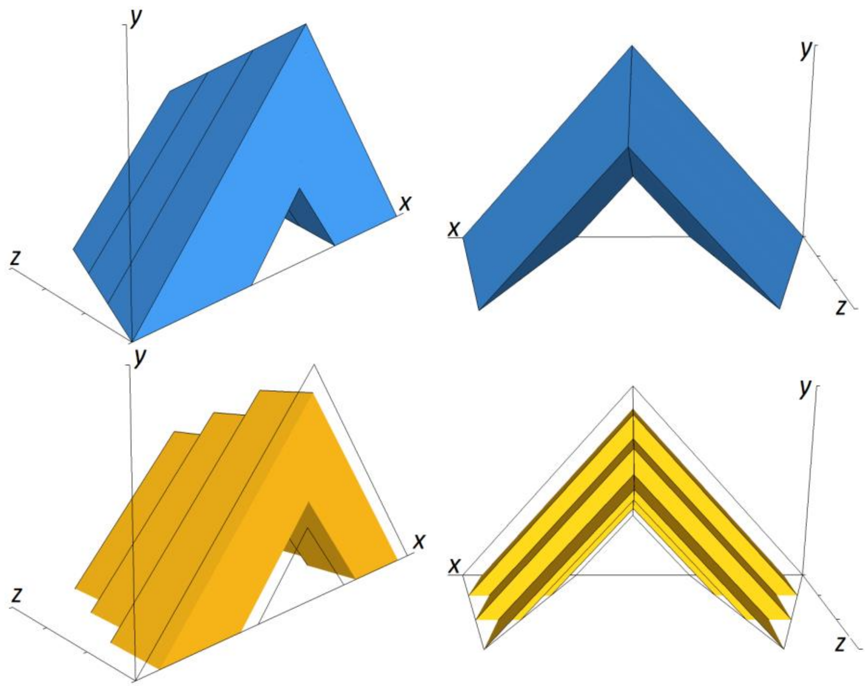


zSlices

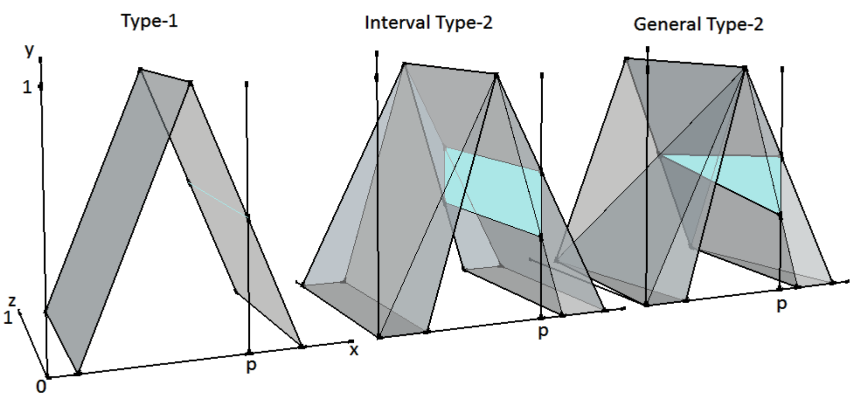
A zSlice is formed by slicing a general type-2 fuzzy set in the third dimension (z) at level z_i. This slicing action will result in an interval set in the third dimension with height z_i. As such, a zSlice $\tilde{Z}_i$ is equivalent to an interval type-2 fuzzy set with the exception that its membership grade $\mu_{\tilde{Z}_i\,(x,u)}$ in the third dimension is not fixed to 1 but is equal to z_i, where $0 \leq z_i \leq 1$. Thus, the zSlice $\tilde{Z}_i$ can be written as follows:

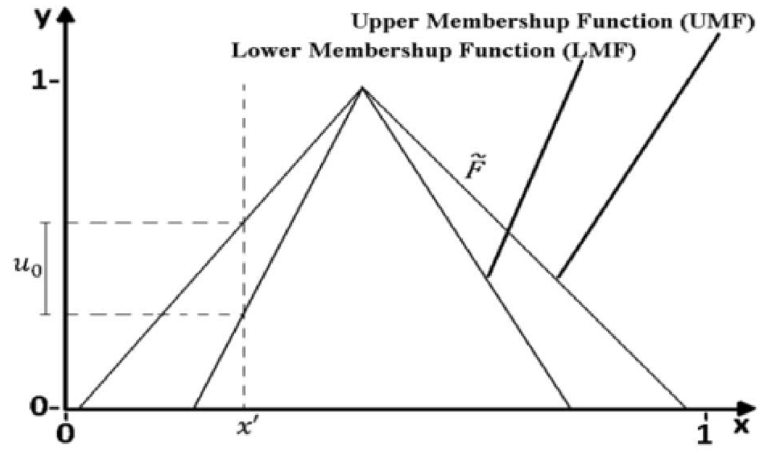
$$\tilde{Z}_i = \int_{x \in X} \int_{u_i \in J_{i\,x}} z_i/(x, u_i) \tag{6}$$

zSlices-based General Type-2 FSs

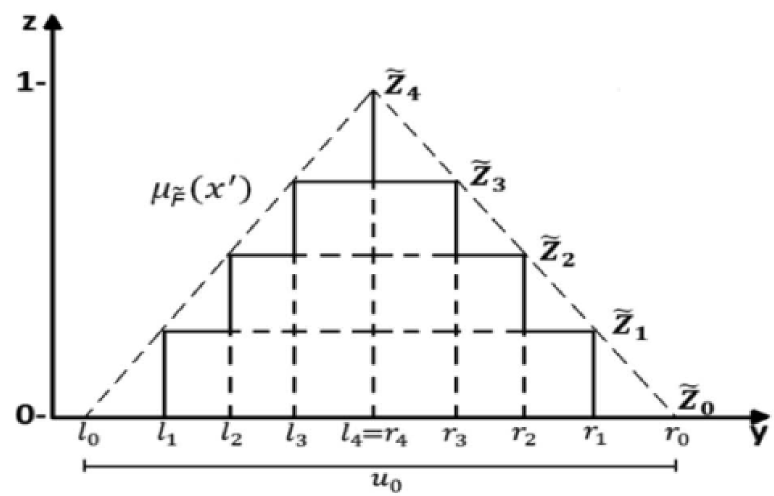


zSlices Illustration





(a)



(b)

From zSlices to General

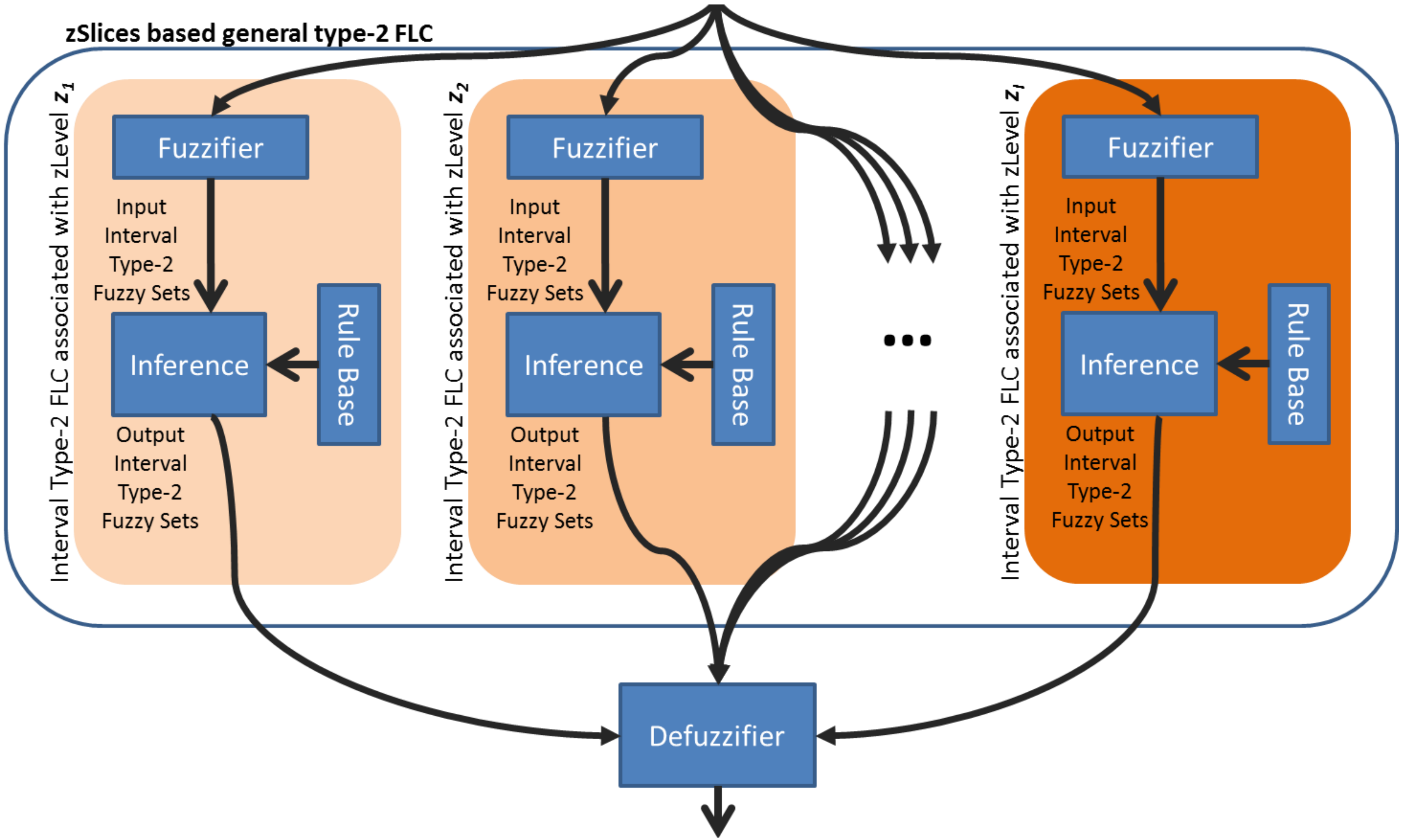
A general type-2 fuzzy set F can be seen equivalent to the collection of an infinite number of zSlices

$$\tilde{F} = \int_{0 \le i \le I} \tilde{Z}_i \qquad I \to \infty. \tag{10}$$

In a discrete universe of discourse, (10) can be rewritten as follows:

$$\tilde{F} = \sum_{i=0}^{I} \tilde{Z}_i. \tag{11}$$

zSlices based Inference



A Robotics Example

Three FLCs

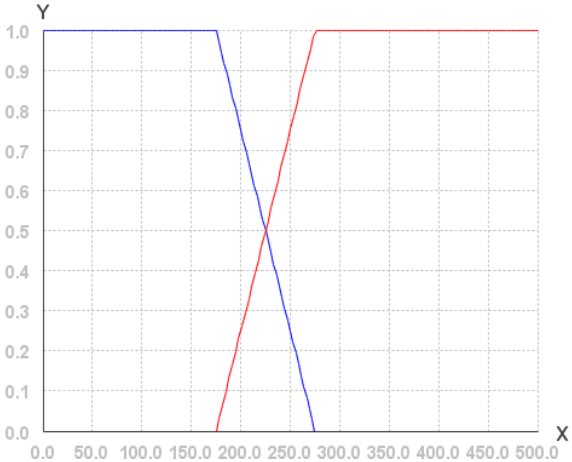
- Type-1
- Interval Type-2
- zSlices based General Type-2

- All implemented using Juzzy, an open source Java library for T1, IT2, and zGT2 FLSs.
- Two inputs, one output.
- Same rule base for all:

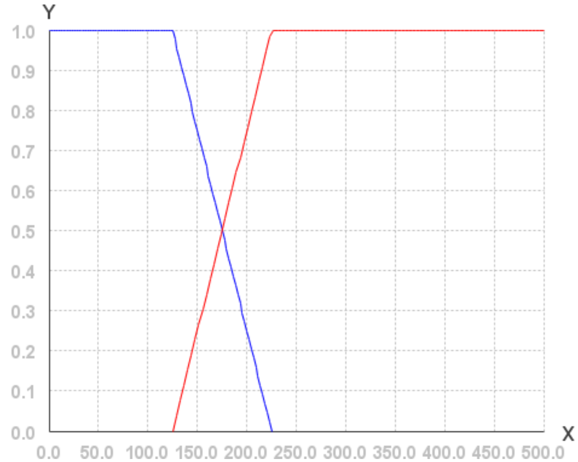
```
rulebase = new T1_Rulebase(4);
rulebase.addRule(new T1_Rule(new T1_Antecedent[]{rfsNear, rbsNear}, left));
rulebase.addRule(new T1_Rule(new T1_Antecedent[]{rfsNear, rbsFar}, left));
rulebase.addRule(new T1_Rule(new T1_Antecedent[]{rfsFar, rbsNear}, zero));
rulebase.addRule(new T1_Rule(new T1_Antecedent[]{rfsFar, rbsFar}, right));
```

Type-1 MFs

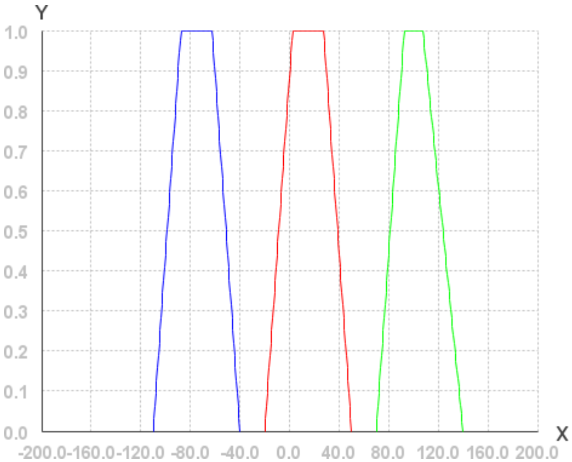
Input: Front Sonar
Near and Far



Input: Side Sonar
Near and Far

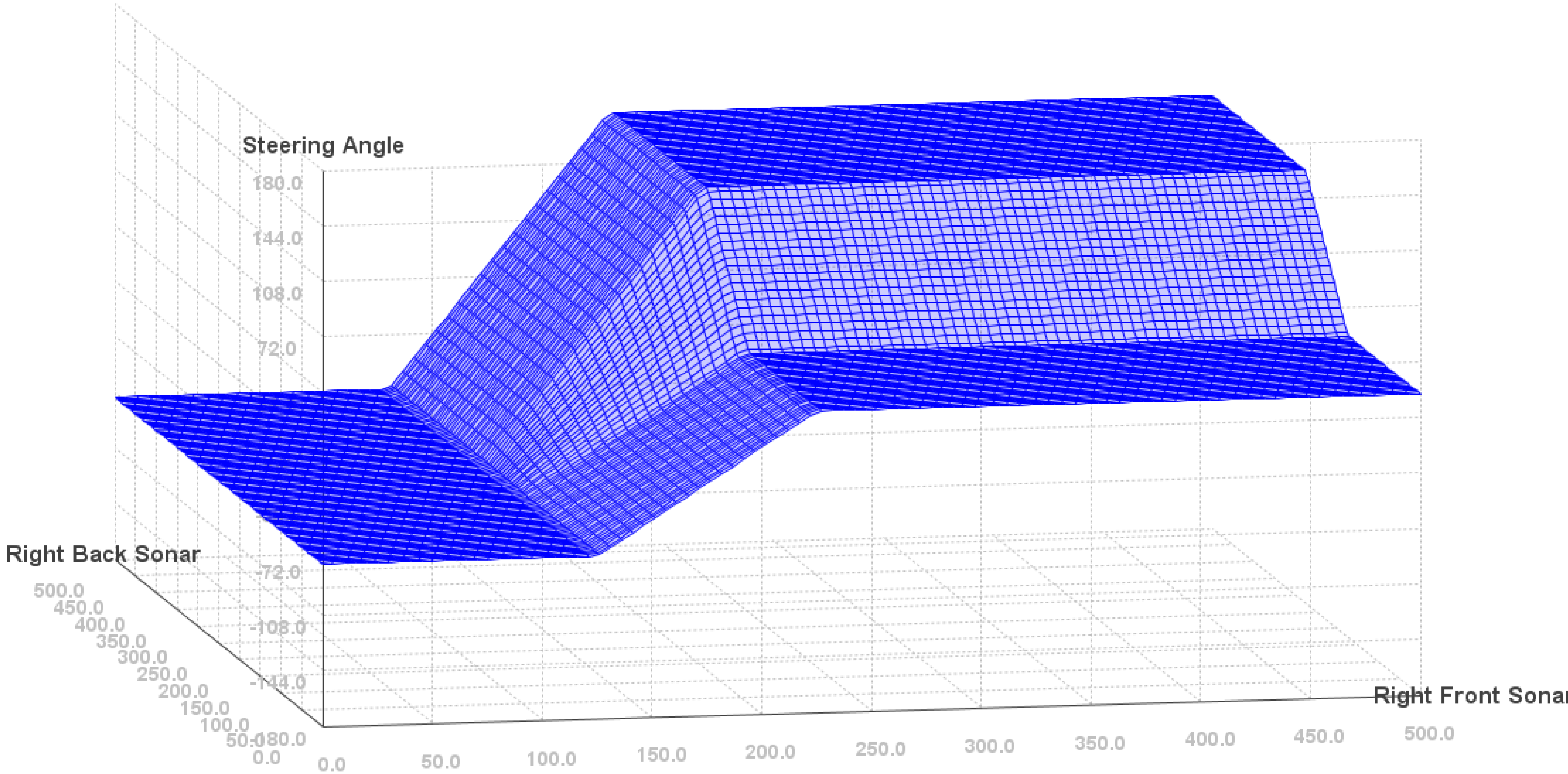


Output: Steering
Left, Straight, Right



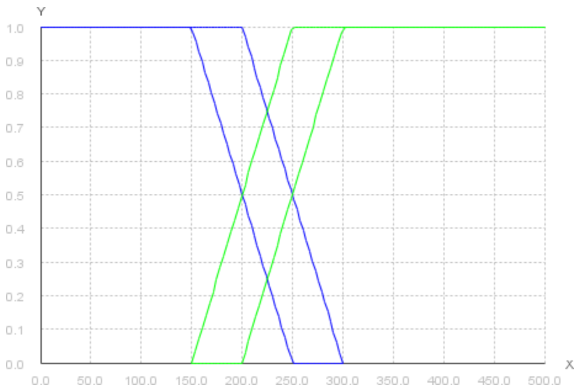

```
T1MF_Trapezoidal rfsNearMF = new T1MF_Trapezoidal("MF for Near distance of the RFS",new double[]{0.0, 0.0, 125.0, 225.0});
T1MF_Trapezoidal rfsFarMF = new T1MF_Trapezoidal("MF for Far distance for the RFS",new double[]{125.0, 225.0, 500.0, 500.0});

T1MF_Trapezoidal rbsNearMF = new T1MF_Trapezoidal("MF for Near distance of the RBS",new double[]{0.0, 0.0, 175.0, 275.0});
T1MF_Trapezoidal rbsFarMF = new T1MF_Trapezoidal("MF for Far distance for the RBS",new double[]{175.0, 275.0, 500.0, 500.0});

T1MF_Trapezoidal leftMF = new T1MF_Trapezoidal("MF for Left Steering Angle",new double[]{-110.0, -87.5, -62.5, -40.0});
T1MF_Trapezoidal zeroMF = new T1MF_Trapezoidal("MF for Zero Steering Angle",new double[]{-20.0, 2.5, 27.5, 50.0});
T1MF_Trapezoidal rightMF = new T1MF_Trapezoidal("MF for Right Steering Angle",new double[]{70.0, 92.5, 107.5, 140.0});
```

Type-1 Control Surface

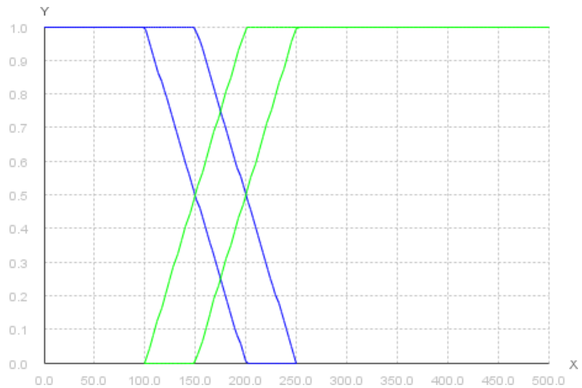


Interval Type-2 MFs

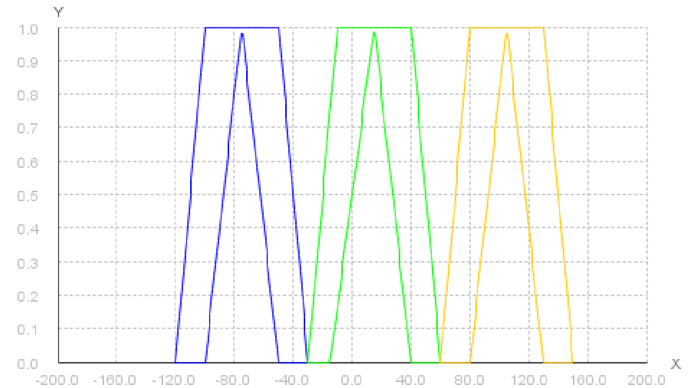
Input: Front Sonar
Near and Far



Input: Side Sonar
Near and Far



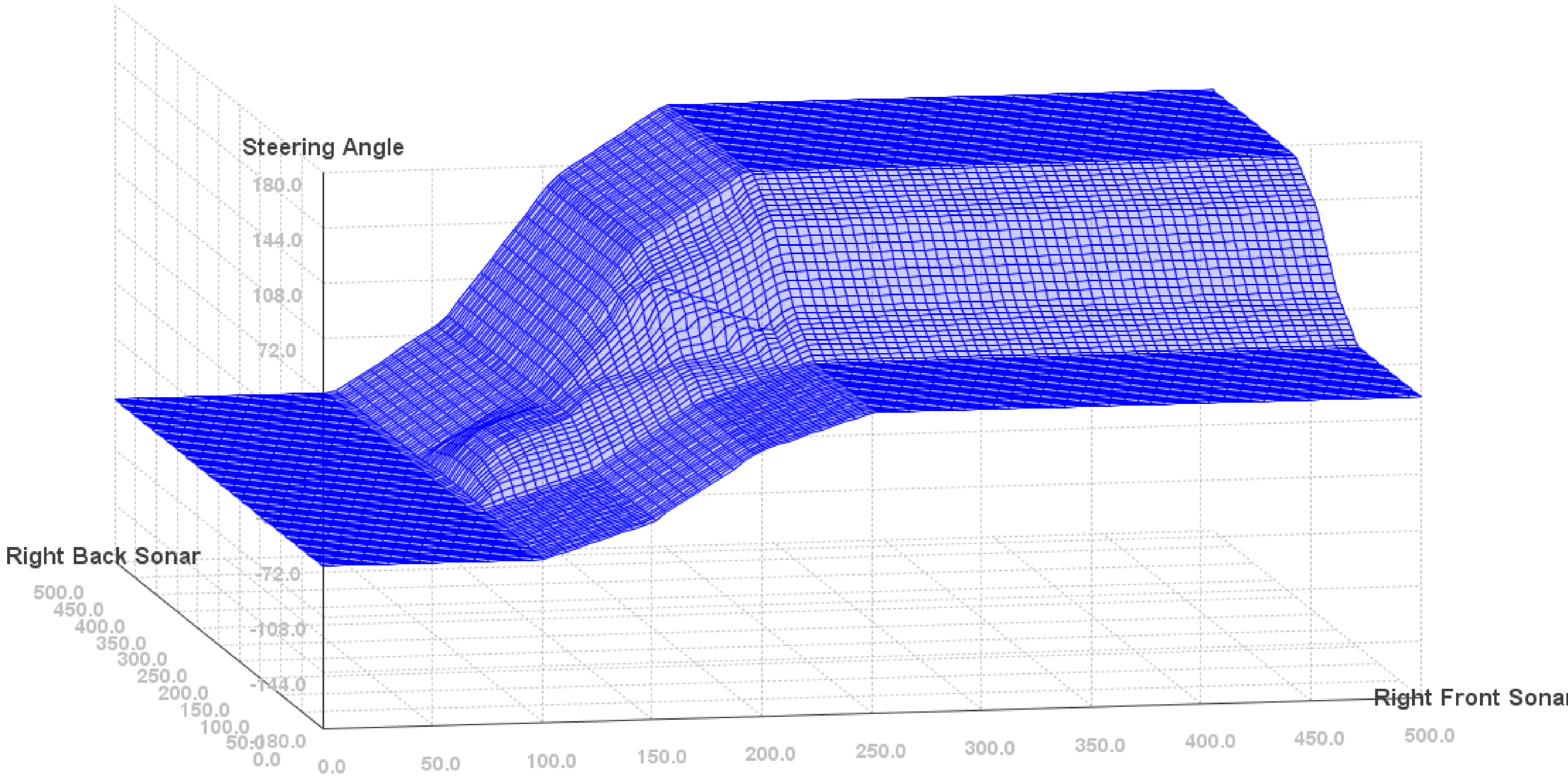
Output: Steering
Left, Straight, Right



```
T1MF_Trapezoidal rfsNearUMF = new T1MF_Trapezoidal("Upper MF for Near distance of the RFS",new double[]{0.0, 0.0, 150.0, 250.0});
T1MF_Trapezoidal rfsNearLMF = new T1MF_Trapezoidal("Lower MF for Near distance of the RFS",new double[]{0.0, 0.0, 100.0, 200.0});
IntervalT2MF_Trapezoidal rfsNearMF = new IntervalT2MF_Trapezoidal("IT2MF for Near distance of the RFS",rfsNearUMF,rfsNearLMF);

T1MF_Trapezoidal rfsFarUMF = new T1MF_Trapezoidal("Upper MF for Far distance of the RFS",new double[]{100.0, 200.0, 500.0, 500.0});
T1MF_Trapezoidal rfsFarLMF = new T1MF_Trapezoidal("Lower MF for Far distance of the RFS",new double[]{150.0, 250.0, 500.0, 500.0});
IntervalT2MF_Trapezoidal rfsFarMF = new IntervalT2MF_Trapezoidal("IT2MF for Far distance of the RFS",rfsFarUMF,rfsFarLMF);

T1MF_Trapezoidal rbsNearUMF = new T1MF_Trapezoidal("Upper MF for Near distance of the RBS",new double[]{0.0, 0.0, 200.0, 300.0});
T1MF_Trapezoidal rbsNearLMF = new T1MF_Trapezoidal("Lower MF for Near distance of the RBS",new double[]{0.0, 0.0, 150.0, 250.0});
IntervalT2MF_Trapezoidal rbsNearMF = new IntervalT2MF_Trapezoidal("IT2MF for Near distance of the RBS",rbsNearUMF,rbsNearLMF);
```

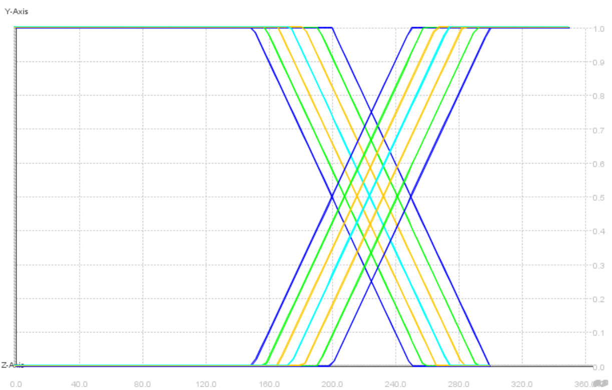
...

Interval Type-2 Control Surface

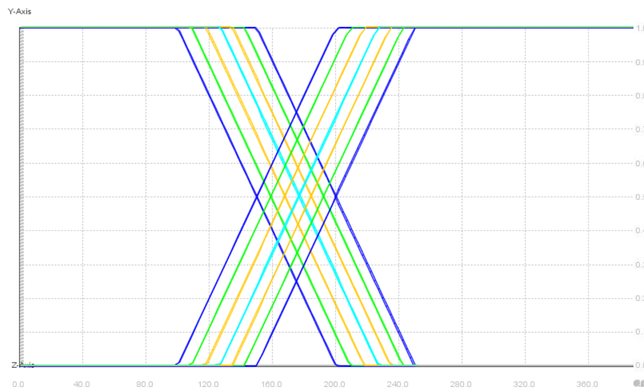


zSlices based General Type-2 MFs

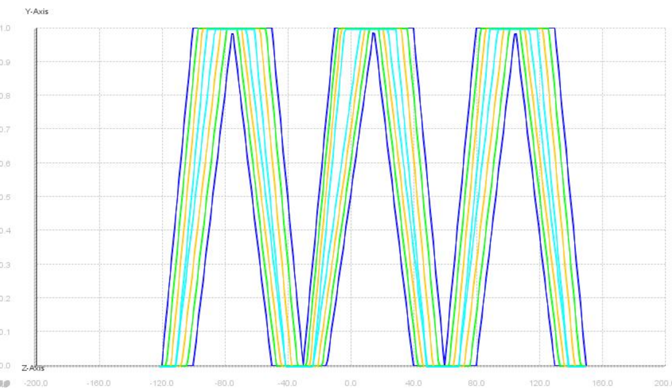
Input: Front Sonar
Near and Far



Input: Side Sonar
Near and Far



Output: Steering
Left, Straight, Right

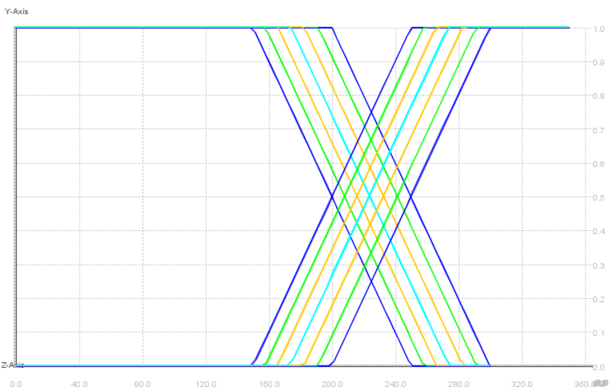


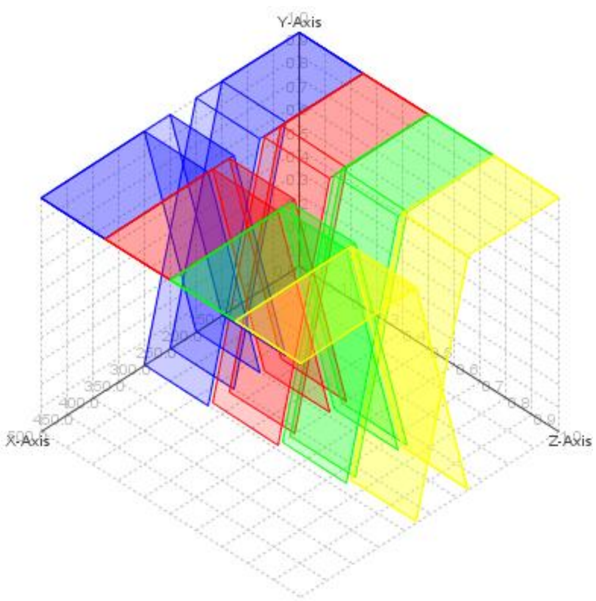
```
//Set up the lower and upper membership functions (MFs) making up the
//overall Interval Type-2 Fuzzy Sets for each input and output
T1MF_Trapezoidal rfsNearUMF = new T1MF_Trapezoidal("Upper MF for Near distance of the RFS",new double[]{0.0, 0.0, 150.0, 250.0});
T1MF_Trapezoidal rfsNearLMF = new T1MF_Trapezoidal("Lower MF for Near distance of the RFS",new double[]{0.0, 0.0, 100.0, 200.0});
IntervalT2MF_Trapezoidal rfsNearIT2MF = new IntervalT2MF_Trapezoidal("IT2MF for Near distance of the RFS",rfsNearUMF,rfsNearLMF);
//now spawn a basic zSlices-based set with 4 zLevels
GenT2zMF_Trapezoidal rfsNearMF = new GenT2zMF_Trapezoidal("zGT2MF for Near distance of the RFS", rfsNearIT2MF, numberOfzLevels);

T1MF_Trapezoidal rfsFarUMF = new T1MF_Trapezoidal("Upper MF for Far distance of the RFS",new double[]{100.0, 200.0, 500.0, 500.0});
T1MF_Trapezoidal rfsFarLMF = new T1MF_Trapezoidal("Lower MF for Far distance of the RFS",new double[]{150.0, 250.0, 500.0, 500.0});
IntervalT2MF_Trapezoidal rfsFarIT2MF = new IntervalT2MF_Trapezoidal("IT2MF for Far distance of the RFS",rfsFarUMF,rfsFarLMF);
GenT2zMF_Trapezoidal rfsFarMF = new GenT2zMF_Trapezoidal("zGT2MF for Far distance of the RFS", rfsFarIT2MF, numberOfzLevels);
```

…

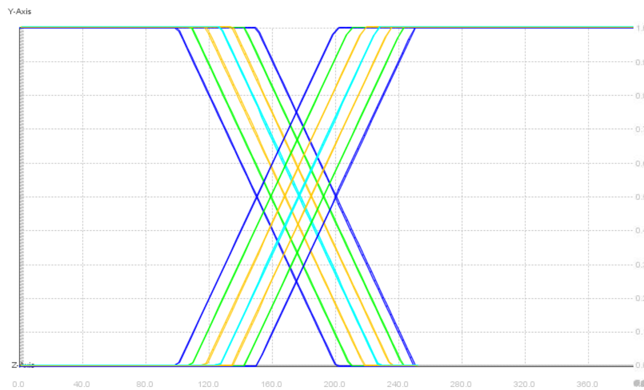
zSlices based General Type-2 MFs

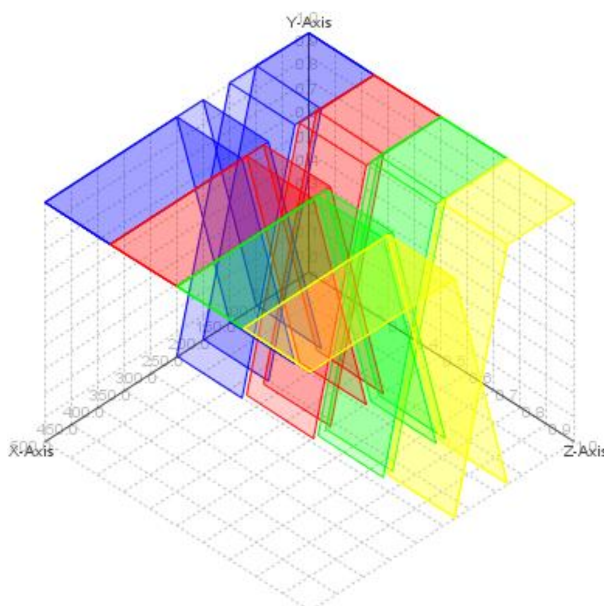
Input: Front Sonar
Near and Far



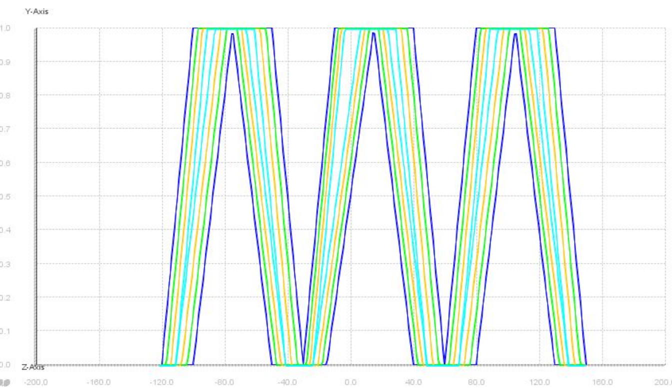


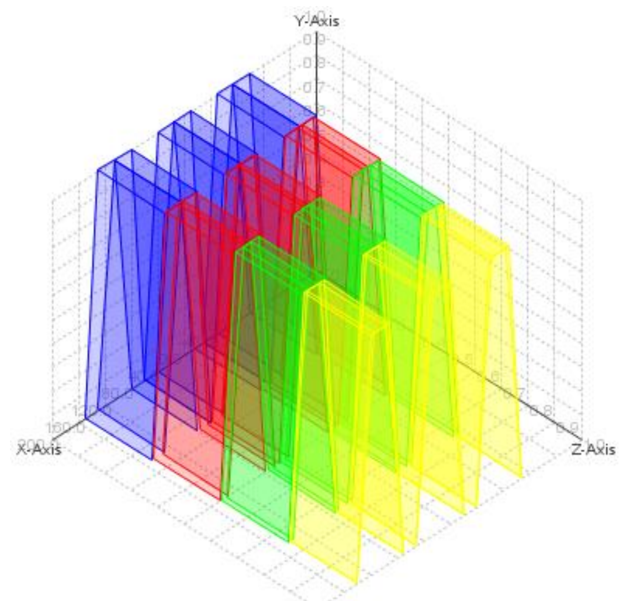
Input: Side Sonar
Near and Far



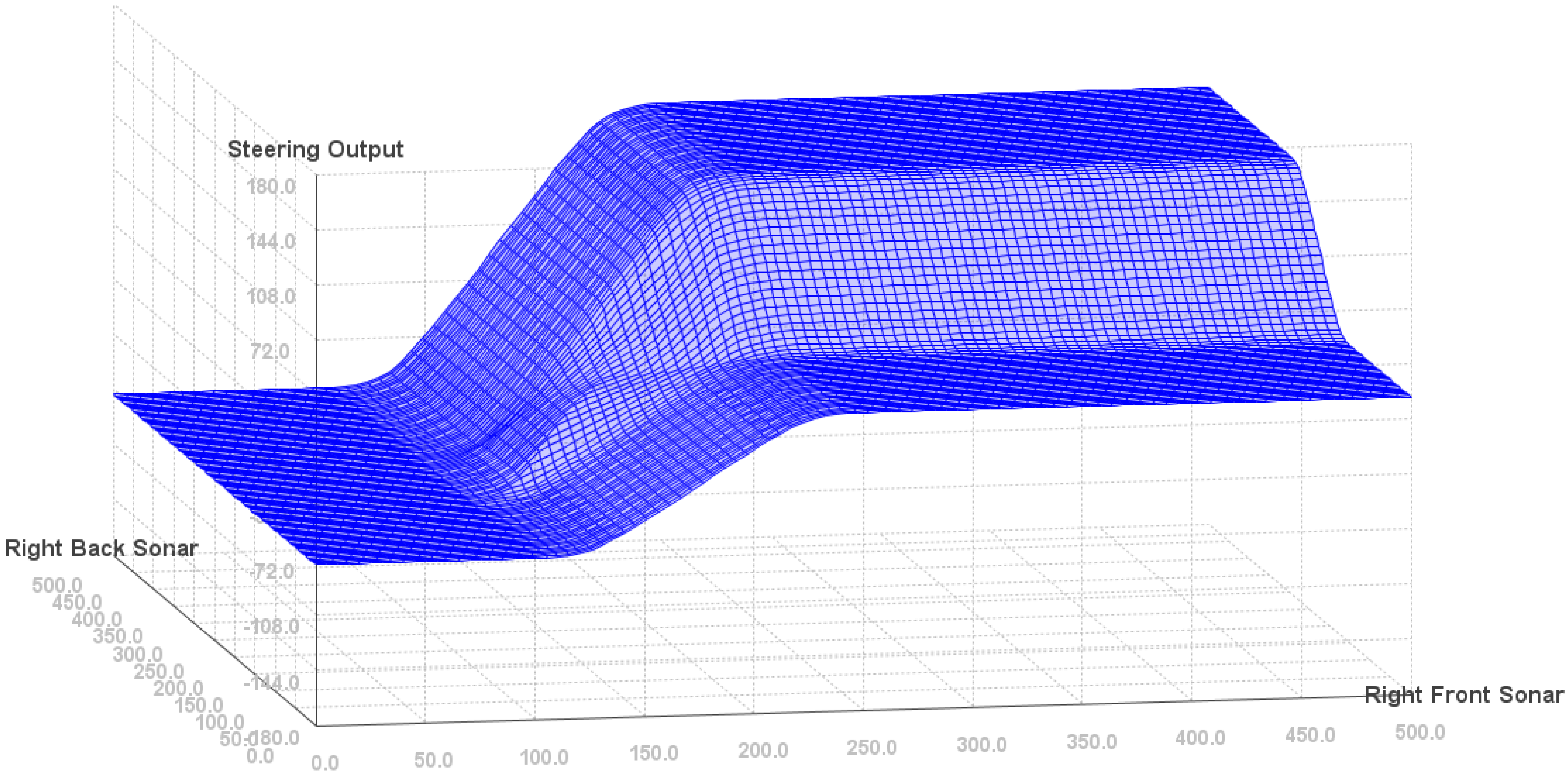


Output: Steer Left,
Straight, Right



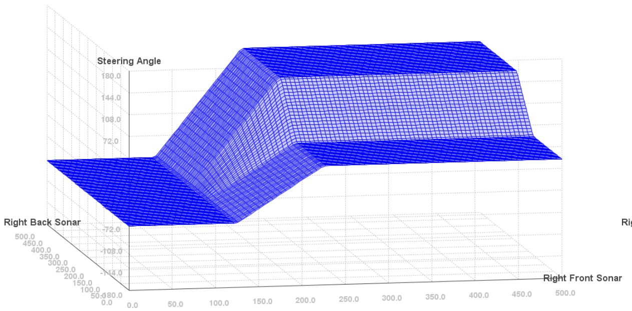


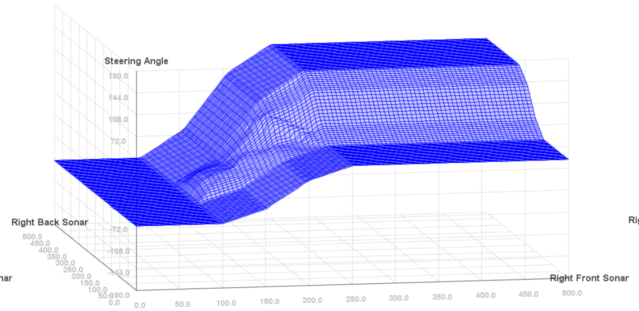
zSlices based general Type-2 CS

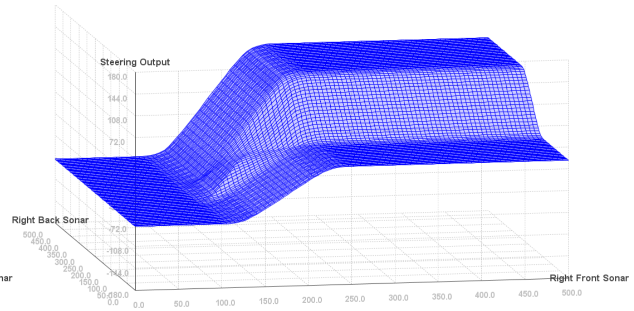


FLCs in comparison

Type-1 Interval Type-2 zGen. Type-2







Generating Control Surfaces

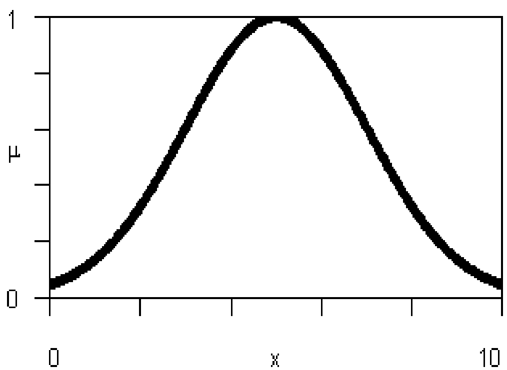
```
private void plotControlSurfaceMC(int input1Discs, int input2Discs)
{
    double output;
    double[] x = new double[input1Discs];
    double[] y = new double[input2Discs];

    double[][] z = new double[y.length][x.length];
    double incrX, incrY;
    incrX = rfs.getDomain().getSize()/(input1Discs-1.0);
    incrY = rbs.getDomain().getSize()/(input2Discs-1.0);

    //first, get the values
    for(int currentX=0; currentX<input1Discs; currentX++)
    {
        x[currentX] = currentX * incrX;
    }
    for(int currentY=0; currentY<input2Discs; currentY++)
    {
        y[currentY] = currentY * incrY;
    }

    for(int currentX=0; currentX<input1Discs; currentX++)
    {
        rfs.setInput(x[currentX]); //set the front sonar input
        for(int currentY=0; currentY<input2Discs; currentY++)
        {
            rbs.setInput(y[currentY]); //set the back sonar input
            output = factory.runFactory();

            z[currentY][currentX] = output;
        }
    }

    //now do the plotting (relies on JMathPlot library: http://code.google.com/p/jmathplot/ (Sep. 2012))
    JMathPlotter plotter = new JMathPlotter();
    plotter.plotControlSurface("Control Surface",
            new String[]{rfs.getName(), rbs.getName(), "Steering Output"}, x, y, z, new Tuple(-180.0, 180.0));
    plotter.show("zSlices based General Type-2 Fuzzy Logic System Control Surface for Tipping Example");
}
```

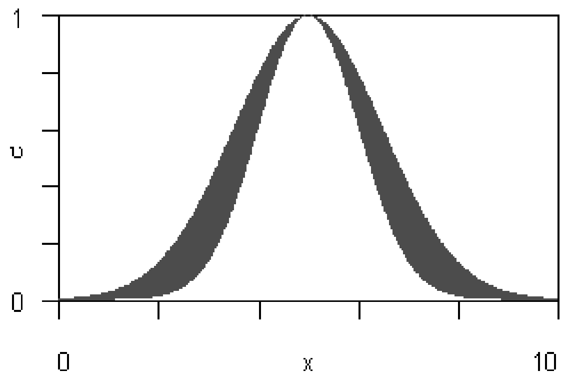
Juzzy Online

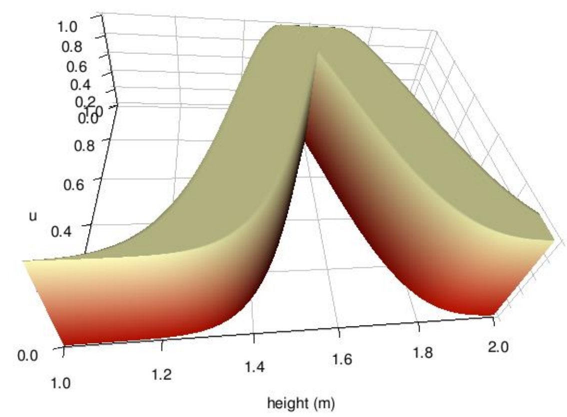
An Online System for Type-1, Interval Type-2 and General-Type-2 Fuzzy Systems

Different Types of Fuzzy Sets

- Type-1, interval type-2 and general type-2 fuzzy sets
- Modelling capability and complexity increase
  - access, in particular to higher-order fuzzy sets and systems, is hampered by the need for programming skills and familiarity with theory



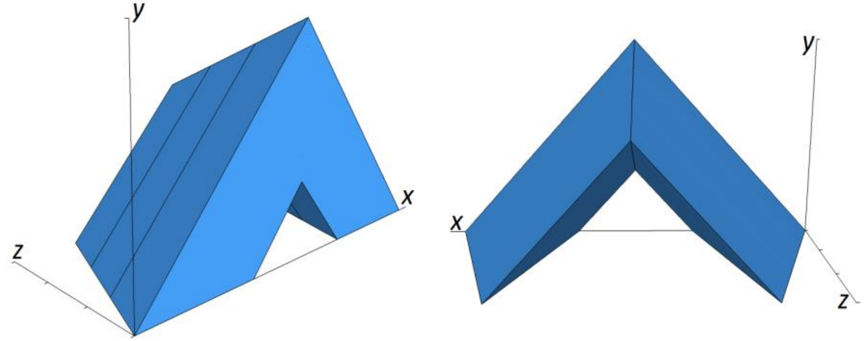




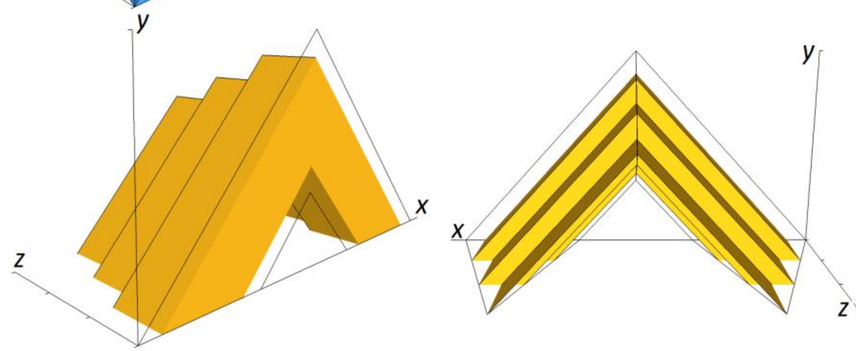
zSlice General Type-2 Fuzzy Sets

- Drastically reduced complexity & highly parallelizable
- Applications from robotic control to ambient intelligent agents, modelling of expert opinion, CWW, etc.

Standard General Type-2 Fuzzy Set



zSlices based General Type-2 Fuzzy Set

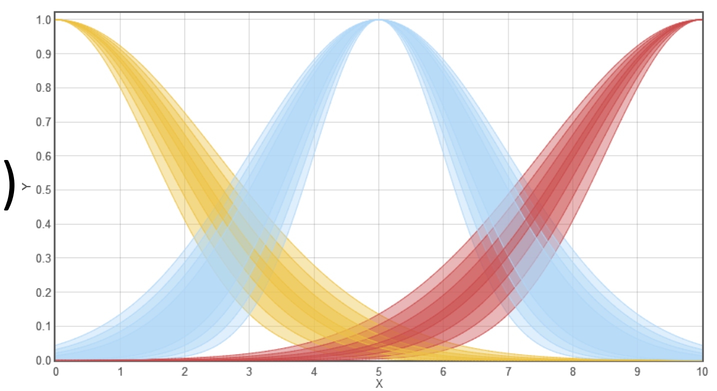


Fuzzy Logic Toolboxes

- There are a variety of toolkits available to develop Fuzzy Logic based applications, e.g.:
  – MATLAB® Fuzzy Logic Tool™ 2 Users'Guide, The MathWorks, Inc., Natick, USA, March 2010. **(type-1)**
  – J.R. Castro, O. Castillo, P. Melin, "An Interval Type-2 Fuzzy Logic Toolbox for Control Applications", *Proc. Int. Conference on Fuzzy Systems* , pp. 1-6, July 2007, London, UK. **(type-1, interval type-2)**
  – C. Wagner, S. Miller, J.M. Garibaldi, " A fuzzy toolbox for the R programming language," *Proc. Int. Conference on Fuzzy Systems*, pp.1185-1192, June 2011, Taipei, Taiwan. **(type-1, interval type-2)**
  – C. Wagner, "Juzzy – A Java based Toolkit for Type-2 Fuzzy Logic", Proc. IEEE Symp. on Advances in Type-2 Fuzzy Logic Systems, pp. 45-52, April 2013, Singapore. **(type-1, interval type-2, general type-2)**
  – And other sources, in particular source code snippets…
- All require the installation of specific software and/or programming experience

JuzzyOnline – Browser Fuzzy Toolkit

- “A graphical, browser-based toolkit for the development of
  - type-1, interval type-2 & general type-2 fuzzy systems.”
- Based on open-source Juzzy software
- Freely accessible online at: Link at end of presentation
  - Completely graphical (no coding required)
  - Compatible with most (all?) browsers & platforms, incl. tablets & smartphones
  - Technologies: Java, Apache Tomcat, PostgreSQL, Microsoft Azure
- Fully detailed outputs (e.g., type-reduced sets)
- Flexible figure generation (e.g., for papers!)
- Easy-sharing of FLSs via web-links (e.g., by email)

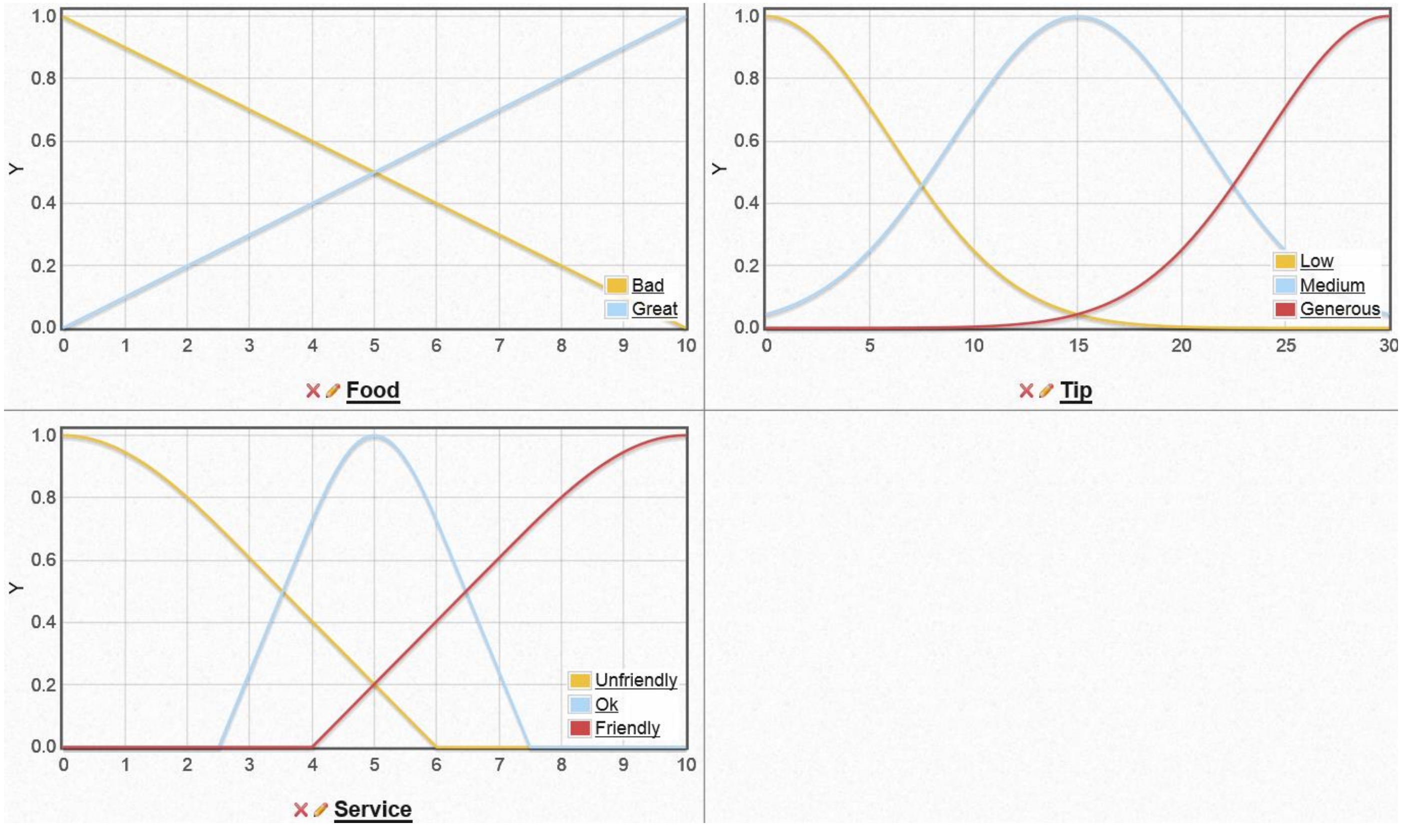


Example – The tipping problem

- We would like to determine the amount of tip (as a percentage) one should give to the waiting staff based on two variables: the quality of the food and the level /quality of service provided by the member(s) of waiting staff
  - Inputs
    - Food: [0, 10]
    - Service: [0, 10]
  - Output
    - Service(%): [0, 30]
- Three FLSs
  - type-1, interval type-2 and zSlices based general type-2

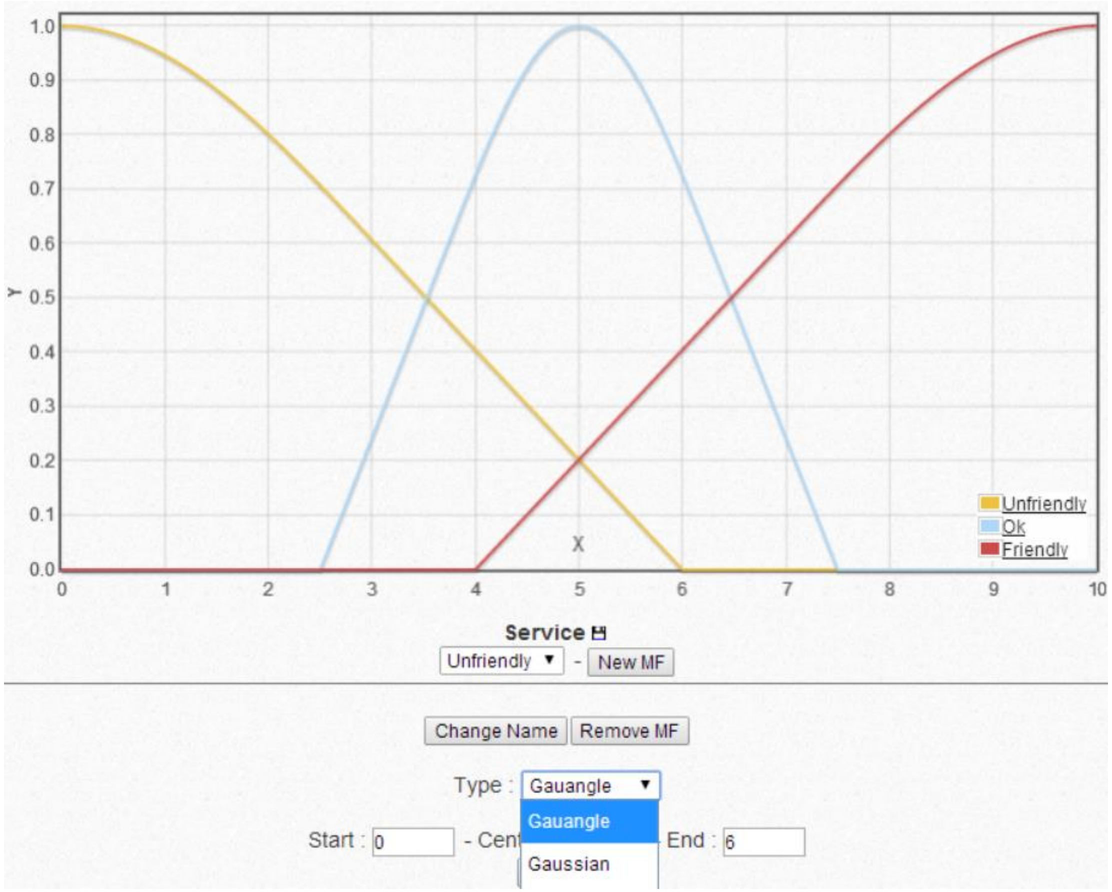
Type-1 Fuzzy Logic System

- A link to an existing system: [Type-1 FLS]
- The type-1 membership functions:



Type-1 Fuzzy Logic System

- Changing, adding and removing membership functions:



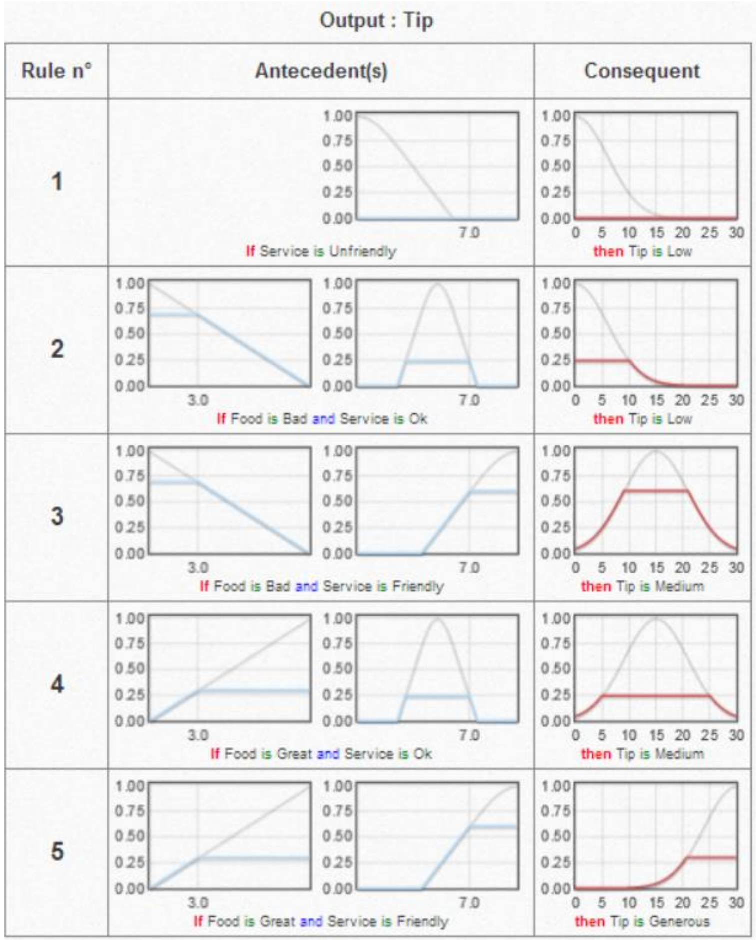
Type-1 Fuzzy Logic System

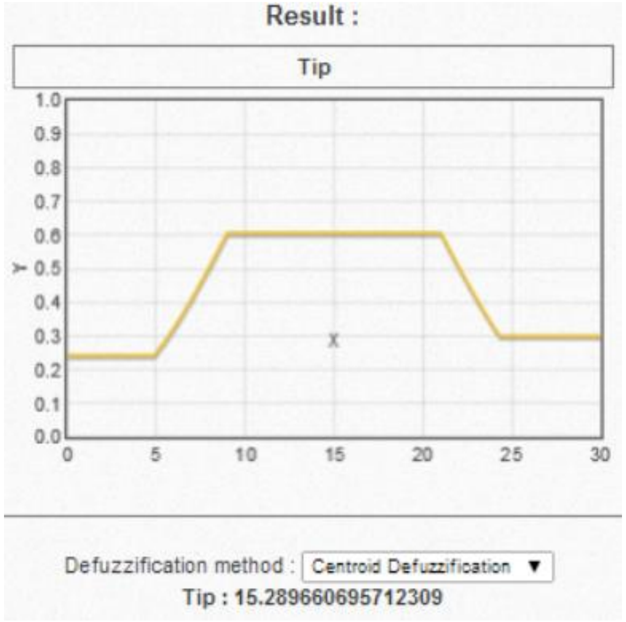
- Adding, removing and editing rules:

Rules
1. If Service is Unfriendly then Tip is Low
2. If Food is Bad and Service is Ok then Tip is Low
3. If Food is Bad and Service is Friendly then Tip is Medium
4. If Food is Great and Service is Ok then Tip is Medium
5. If Food is Great and Service is Friendly then Tip is Generous
New Rule

Type-1 Fuzzy Logic System

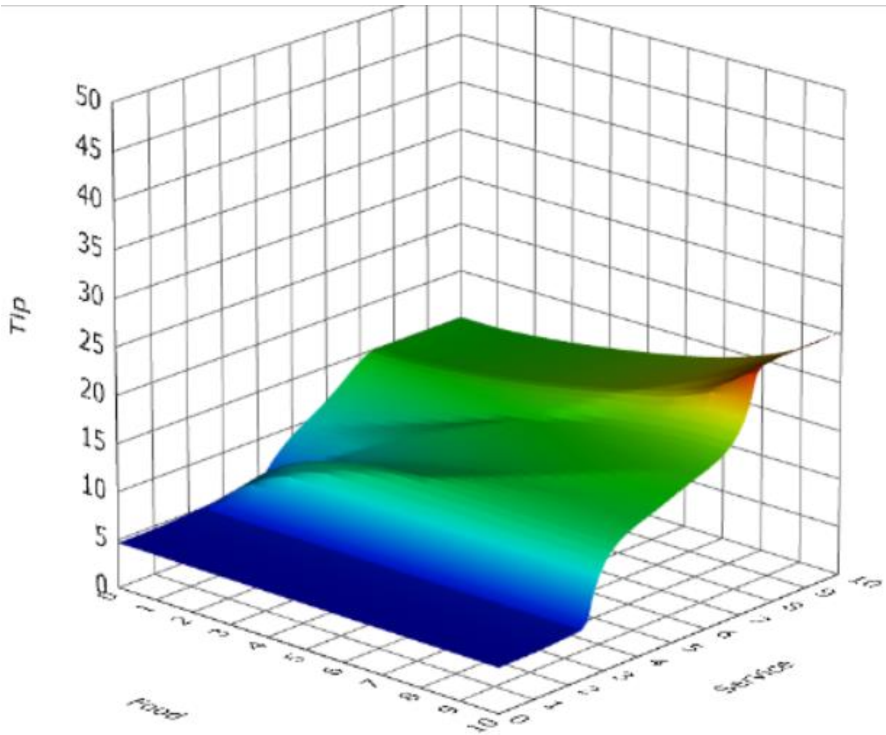
- Inference view
  - here, two inputs: food=3, service =7, centroid

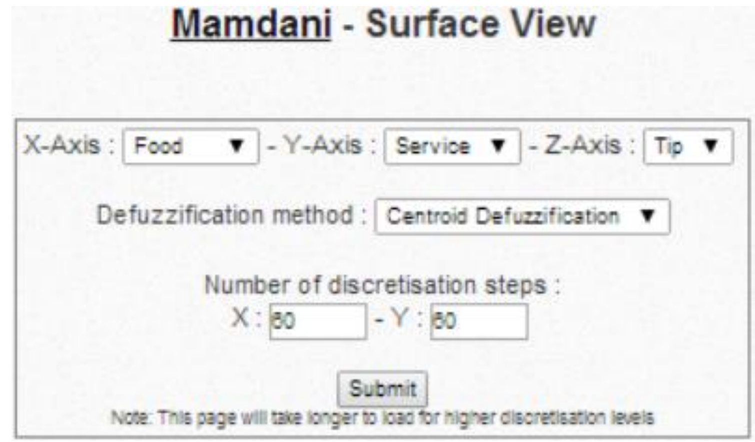




Type-1 Fuzzy Logic System

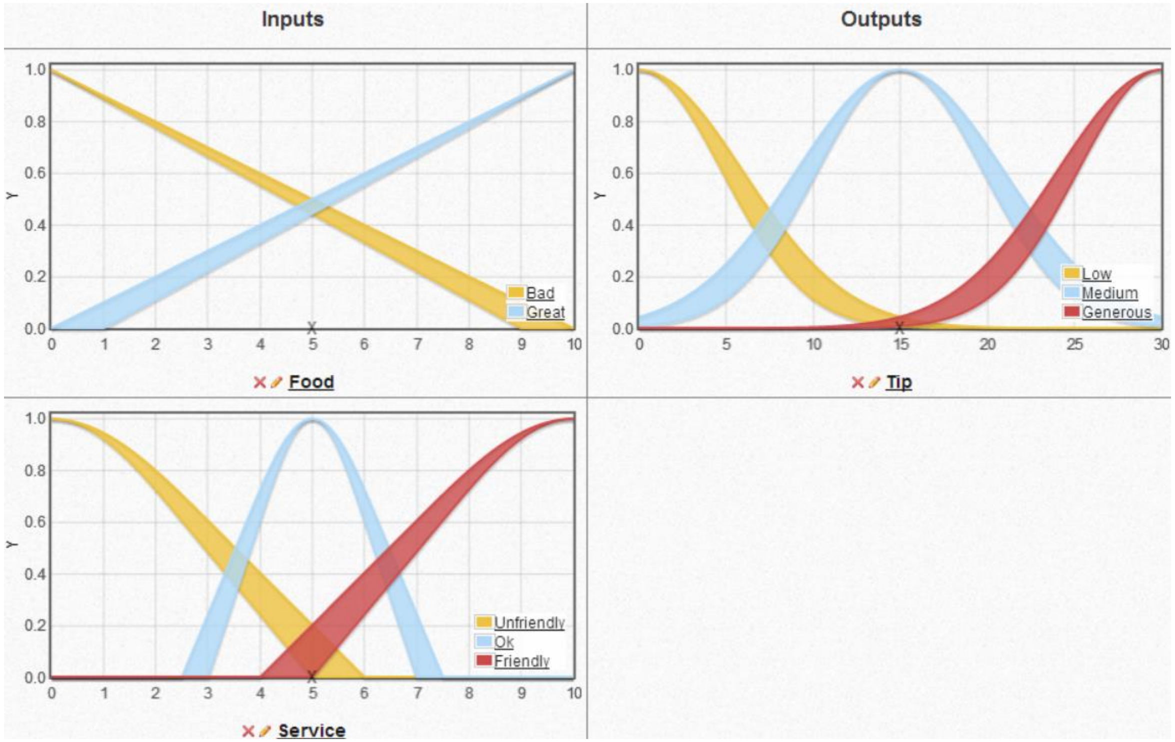
- Control Surface View
  - Two inputs, one output





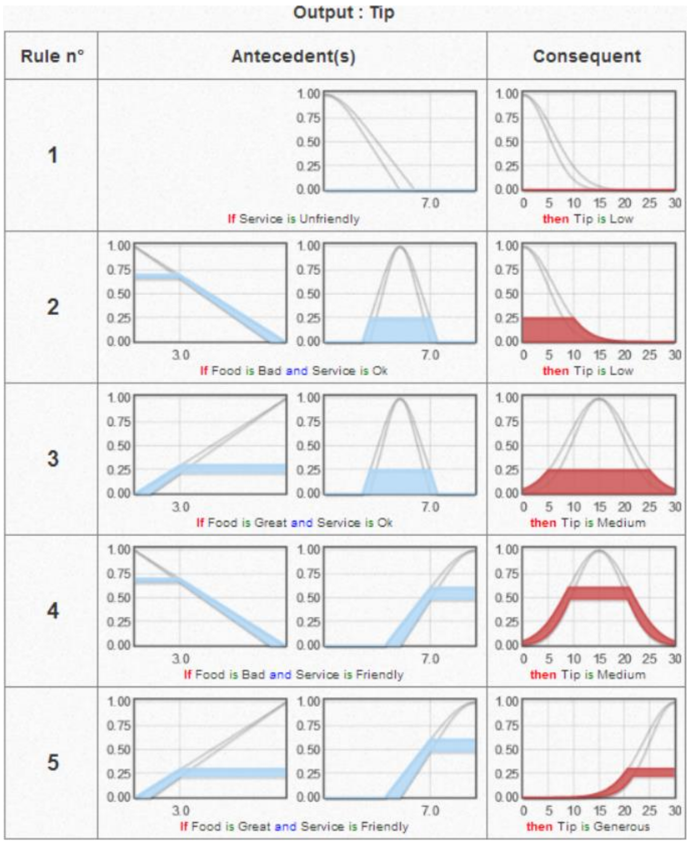
Interval Type-2 FLS Highlights

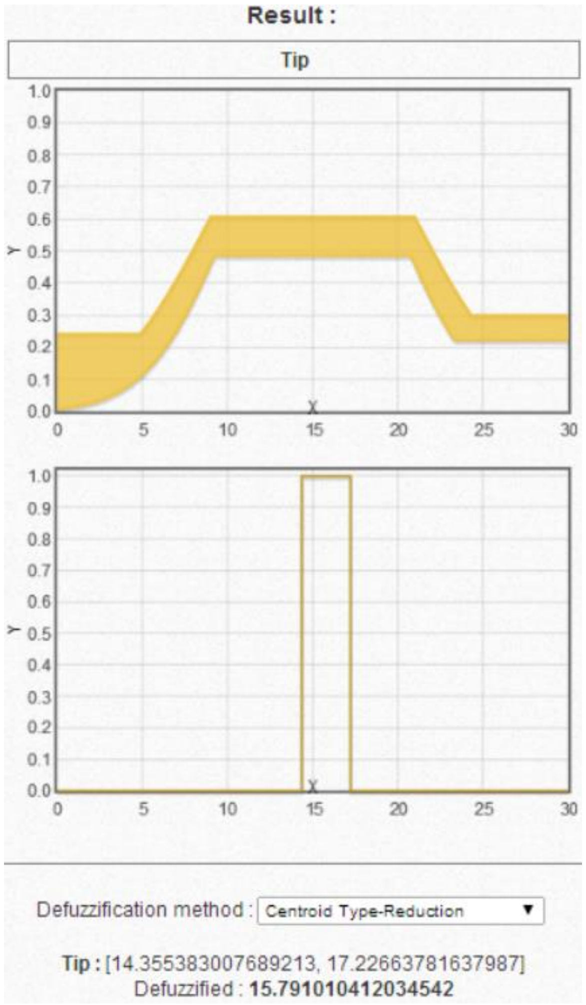
- A link to an existing system: [Interval Type-2 FLS]
  - The rules are identical to the type-1 FLS
- The interval type-2 membership functions:



Interval Type-2 Fuzzy Logic System

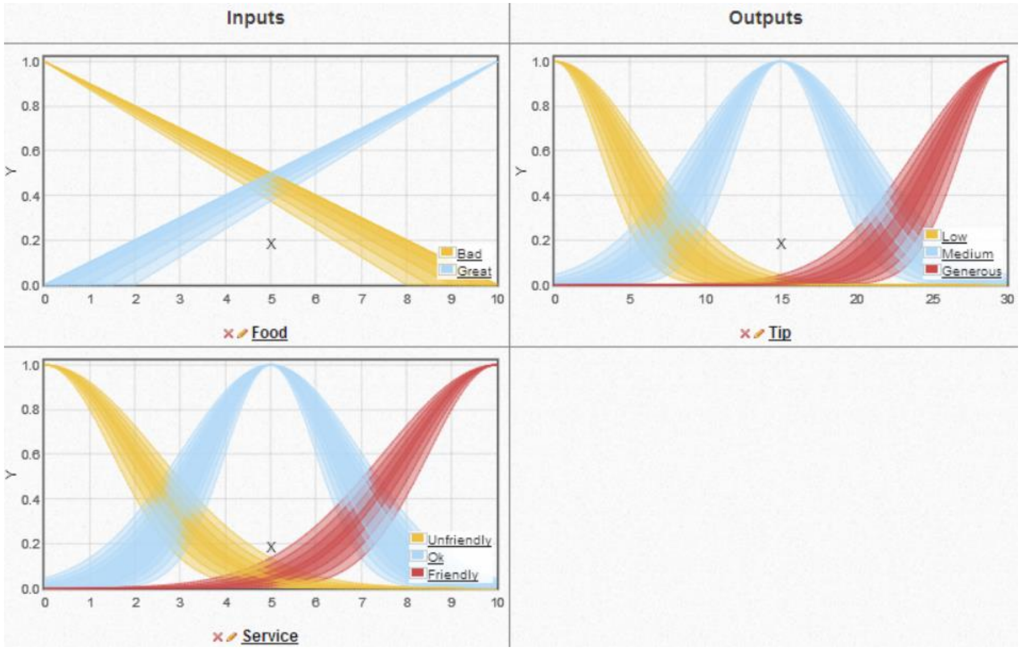
- Inference view
  - here, two inputs: food=3, service =7
  - Centroid type reduction





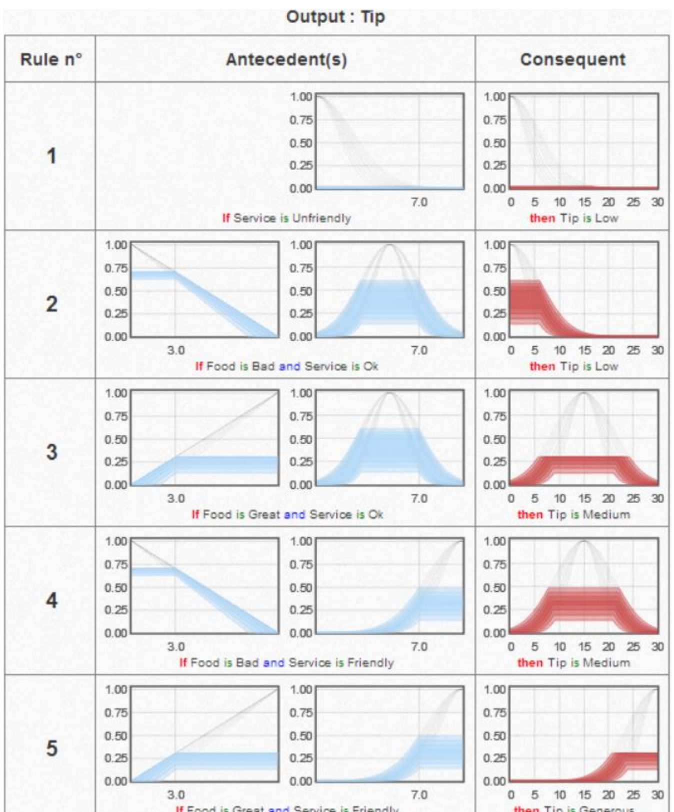
General Type-2 FLS Highlights

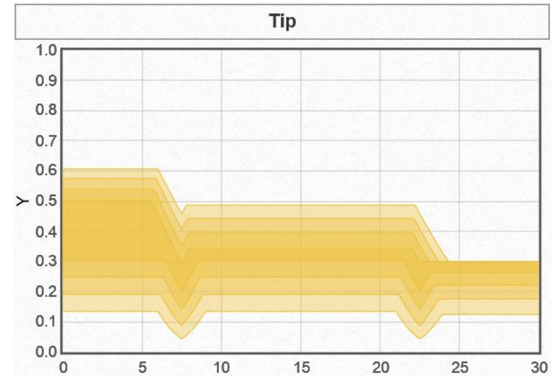
- A link to an existing system: [General Type-2 FLS]
  - The rules are identical to the type-1 FLS
- The zSlices based general type-2 membership functions:
  - (here with 4 zSlices)

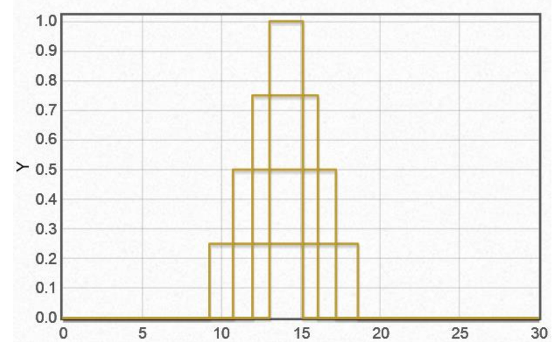


General Type-2 Fuzzy Logic System

- Inference view
  - here, two inputs: food=3, service =7
  - Centroid type reduction

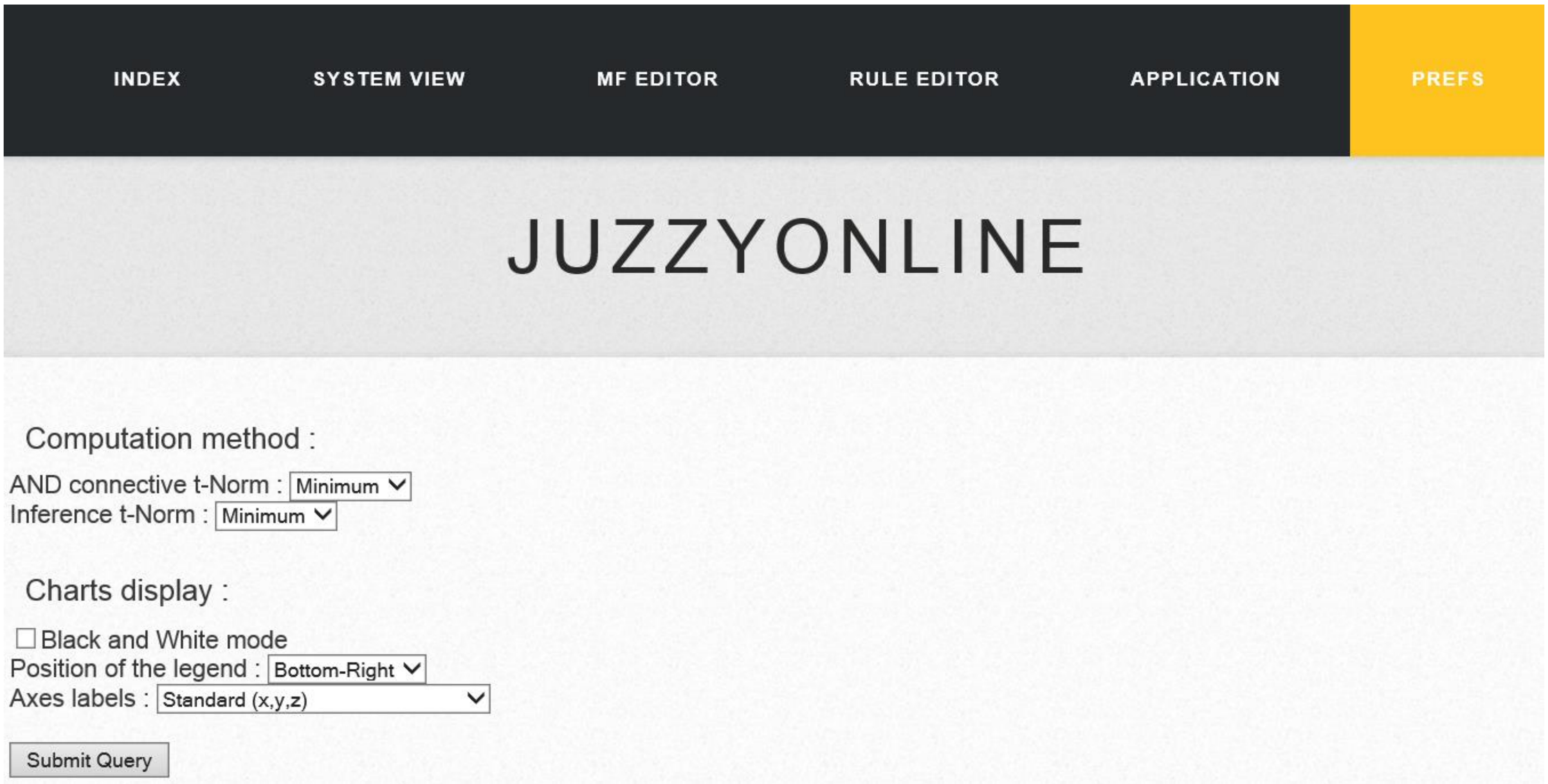






Tip				
zSlice No	zLevel		Output	
1	0.25	Centroid	9.246949872860087	18.598836453261516
		Defuzzified Value	13.922893163060802	
2	0.50	Centroid	10.729924110679805	17.214837768443775
		Defuzzified Value	13.972380939561791	
3	0.75	Centroid	11.952596643306966	16.07826427357507
		Defuzzified Value	14.015430458441017	
4	1.00	Centroid	13.031473015556638	15.131965290494085
		Defuzzified Value	14.081719153025361	
All		Overall Centroid	11.869048008636713	16.179116597285063
		Overall Defuzzified Value	**14.024082302960888**	

Navigation and Extra Features



Current and Future Development

- JuzzyOnline enables:
  - Design & execution of type-1, interval and general type-2 FLSs
  - No programming expertise or software required
  - Easy sharing of FLSs
  - Easy output and figure generation

- Current and future feature expansion:
  - Other membership functions
  - Detailed uncertainty visualisation
  - Non-singleton fuzzification
- Suggestions?

Links

- http://juzzyonline.wagnerweb.net :

Constructing General Type-2 Fuzzy Sets

C Wagner, S Miller, JM Garibaldi, DT Anderson, TC Havens,

"From Interval-valued Data to General Type-2 Fuzzy Sets"

IEEE Transactions on Fuzzy Systems, 23(2), 248-269, 2015

Aims

- In this work we aim to:
  - create an accurate representation of interval-valued survey data
  - capture the different types of uncertainty that are present
  - keeping all information contained in the original data, and making no assumptions about the data

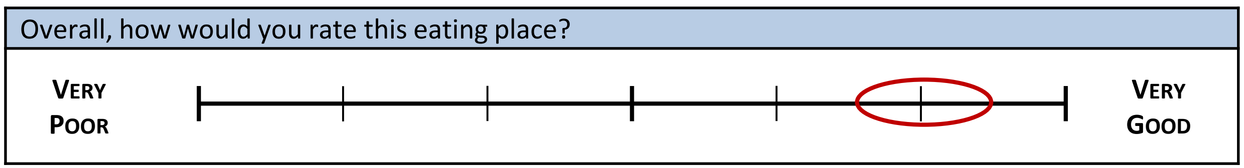
Survey Data

- The objective of our survey is to:
- Elicit opinions from a group of experts
- Study the dynamics (and variation) in decision making
  - Aggregate sections of data to derive overall decisions.
- Model survey data in an expert system for use on new data

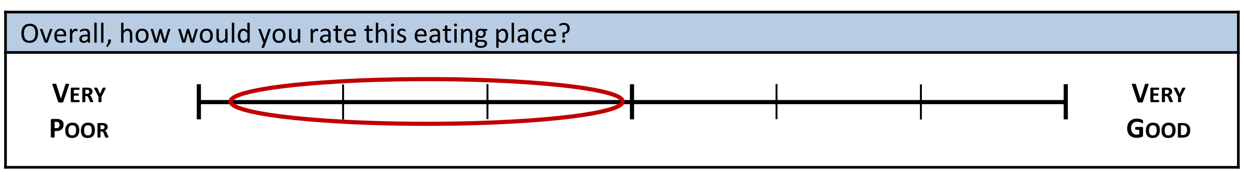
Survey Data

To capture uncertainty, intervals are used:

Certain



Uncertain



The width of the ellipse denotes certainty.

Fuzzy Sets

- We have proposed a novel fuzzy approach to interval modelling
- Fuzzyness in survey data could be:
  - An individual’s certainty in their answer
  - Variation in opinion between individuals (inter-expert)
  - Variation in opinion of one individual over repeated surveys (intra-expert)

- In particular, we have proposed the use of zSlices based General Type-2 fuzzy sets

From Intervals to Fuzzy Sets

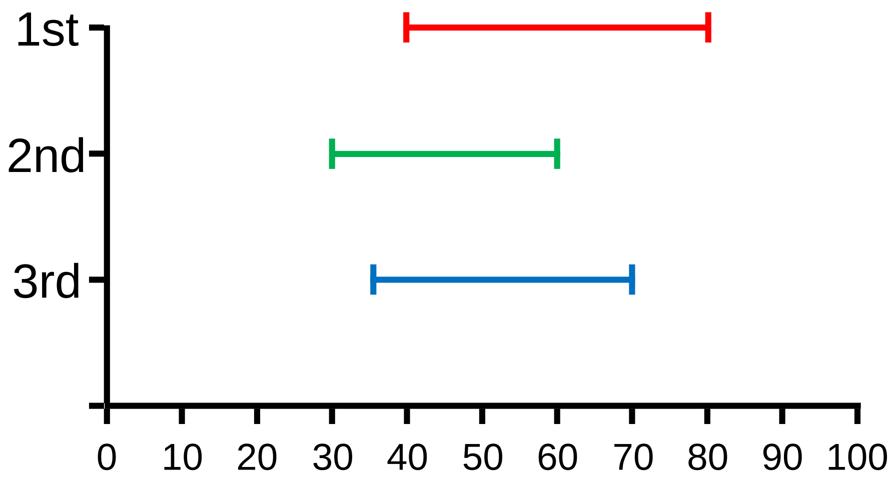
	Expert A		Expert B		Expert C		Expert D	
1st Answer	40	80	40	85	20	80	25	75
2nd Answer	30	60	50	80	30	85	35	75
3rd Answer	35	70	45	95	25	75	30	70

Here, 4 experts have been surveyed 3 times on the same subject.

From Intervals to Fuzzy Sets

Viewing the intervals, it can be seen that there is some agreement.

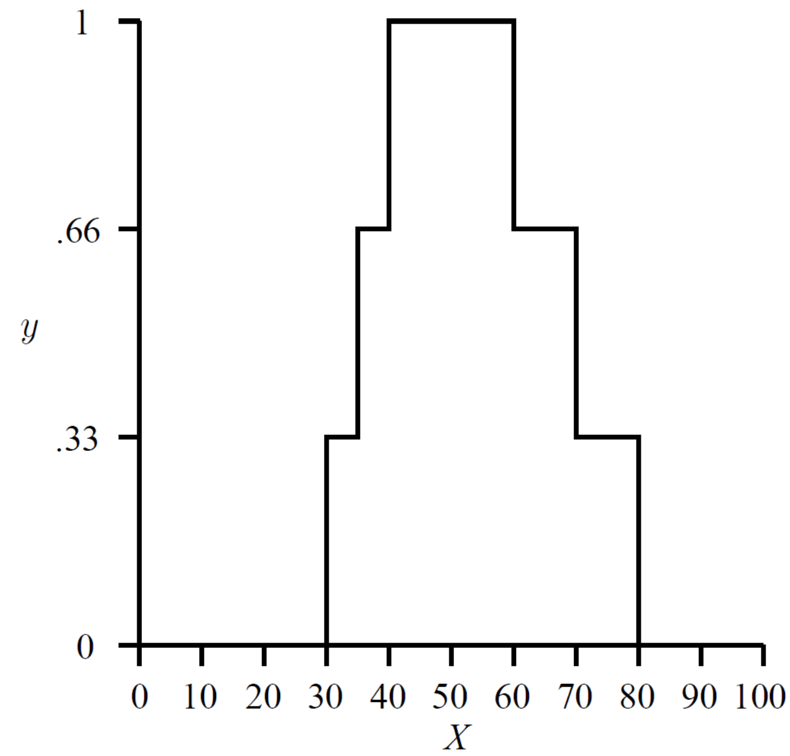
	Expert A	
1st Answer	40	80
2nd Answer	30	60
3rd Answer	35	70



Using the proposed method, this agreement can be used to produce a Type-1 fuzzy set.

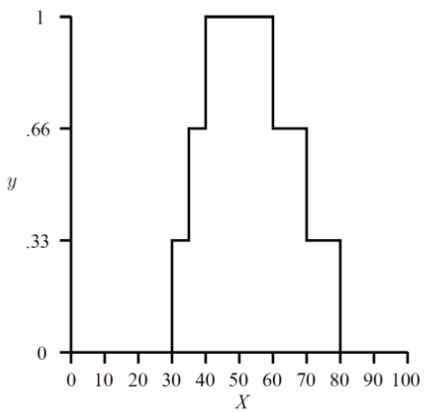
From Intervals to Fuzzy Sets

- To do this with the proposed method:
  - Divide y into 3 sections.
  - Compute 3 intervals in X for each level of agreement.
  - Create a Type-1 fuzzy set using the intervals.

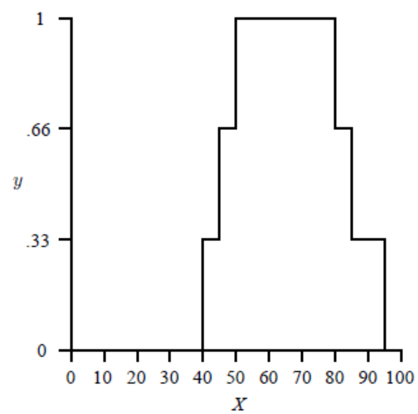


From Intervals to Fuzzy Sets

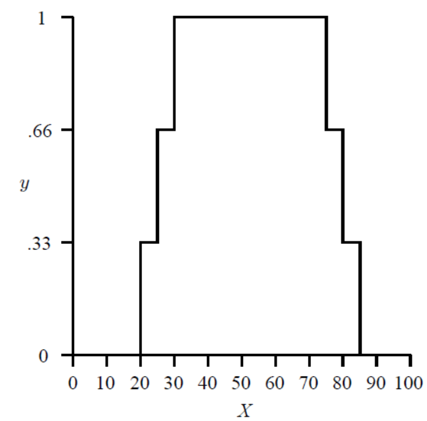
- If we repeat this process for all 4 experts:



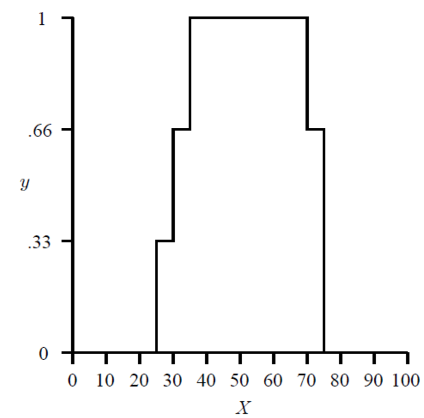
Expert A



Expert B



Expert C



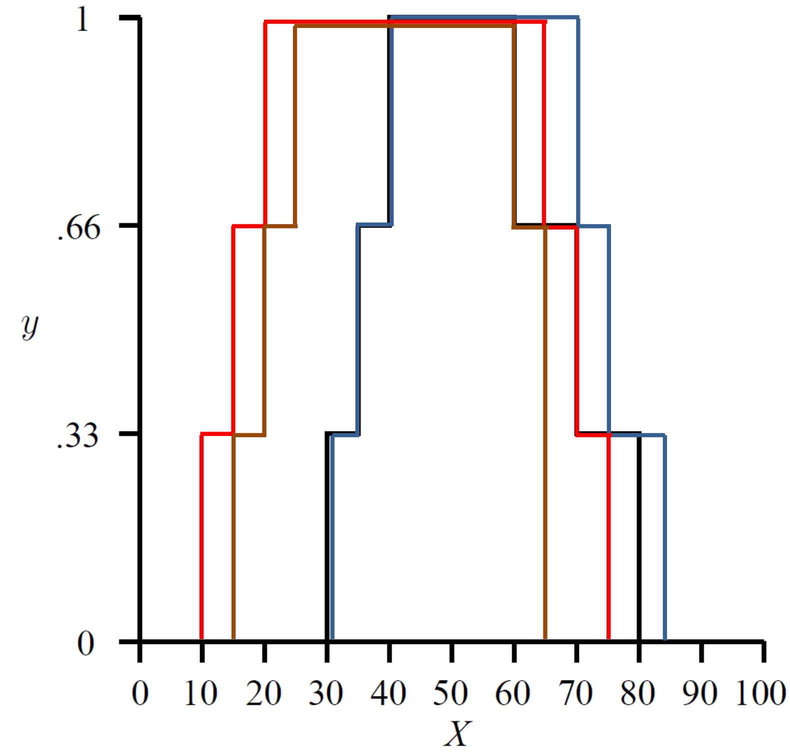
Expert D

- We have 4 fuzzy sets that model the intra-expert variation for each of the 4 experts.

From Intervals to Fuzzy Sets

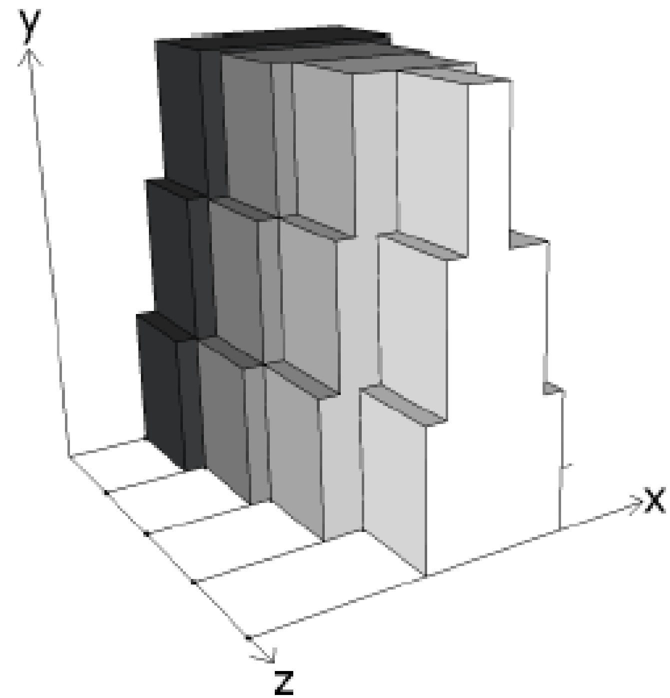
- Again, there is agreement between the 4 Type-1 fuzzy sets

- Using the proposed method, these sets can be used to create a General Type-2 Fuzzy set



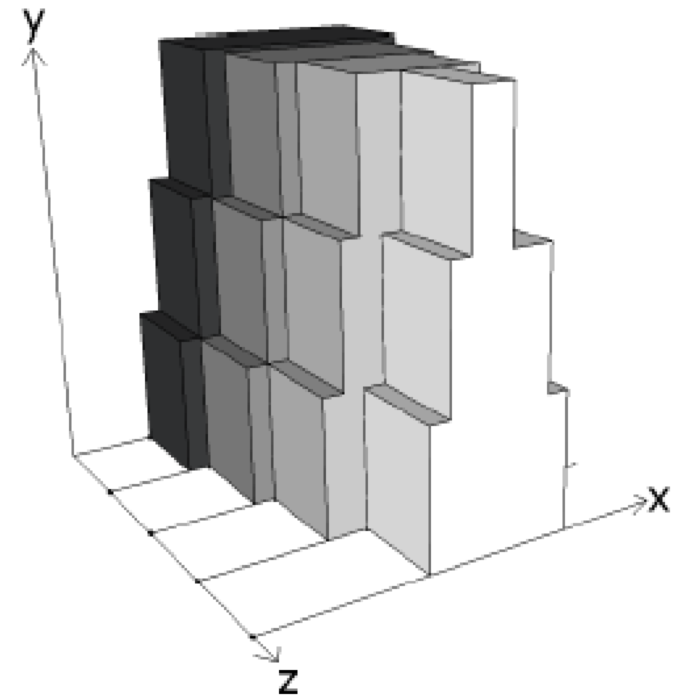
From Intervals to Fuzzy Sets

- To do this:
  - Divide Z into 4 sections
  - Compute 4 Type-1 Fuzzy sets for each level of agreement
  - Create a General Type-2 fuzzy set using the Type-1 fuzzy sets



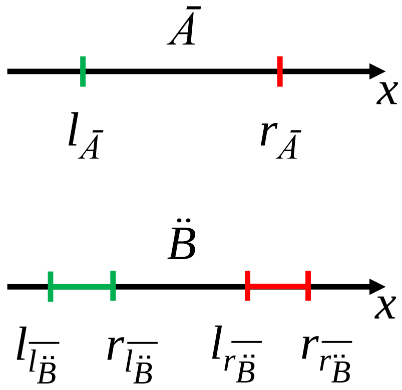
From Intervals to Fuzzy Sets

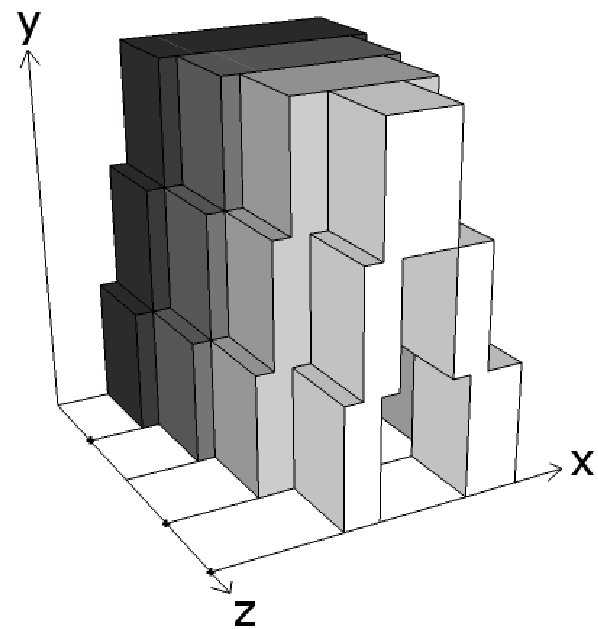
- Resulting in a General Type-2 Fuzzy set that:
  - Represents Inter- and Intra-expert variation in two distinct dimensions
  - Makes no assumptions about the data, discarding no information



Current Work

- Current work has focused on uncertain intervals:
- In uncertain intervals, the endpoints are uncertain
- This uncertainty can be modelled with the FOU
- Step 1 results in an IT2 fuzzy set
- Step 2 results in a GT2 fuzzy set with different LMF and UMF





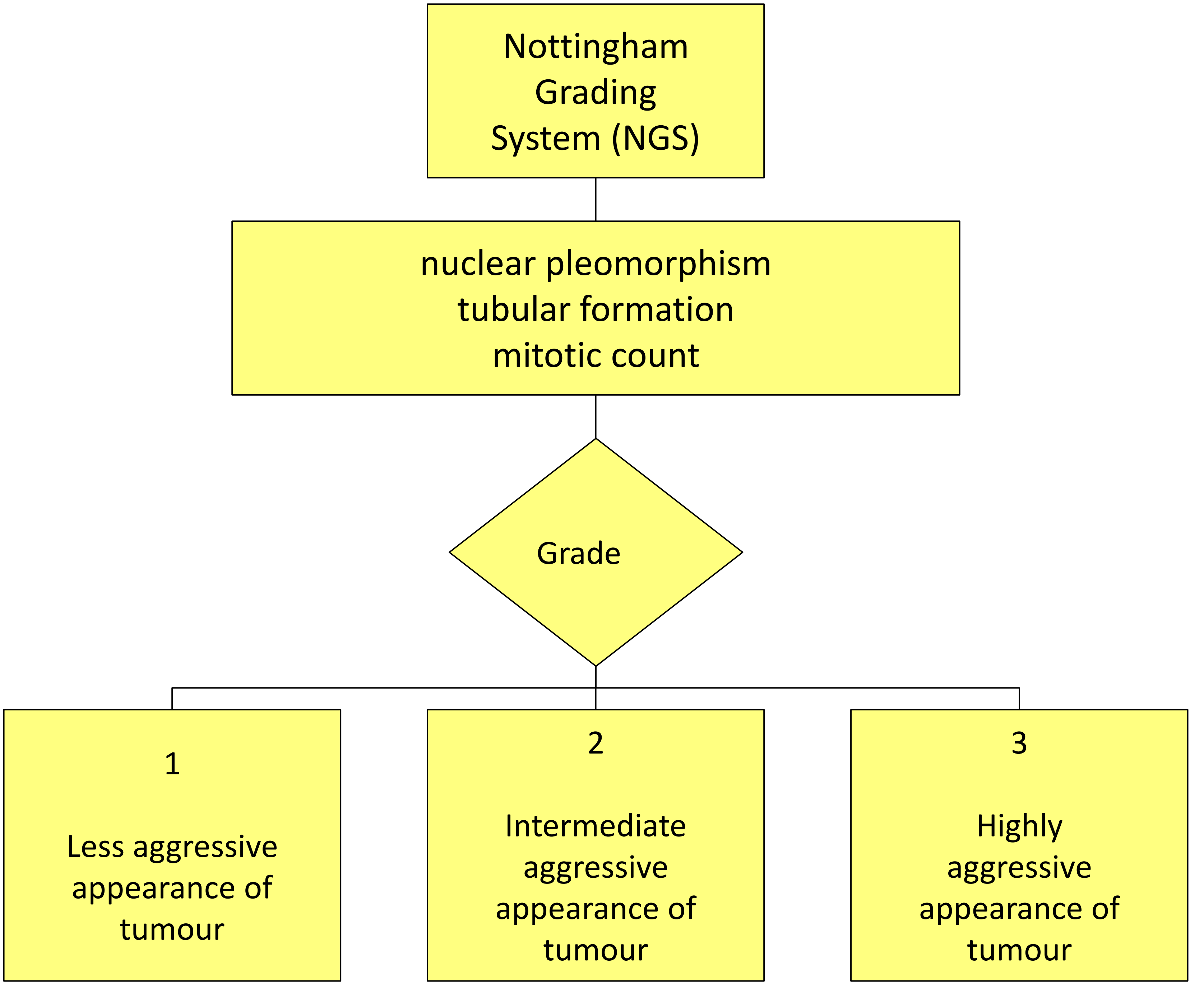
GT2 Application

S. Naqvi, S. Miller, J.M. Garibaldi
"A General Type-II Similarity Based Model for Breast Cancer Grading with FTIR Spectral Data"

FUZZ-IEEE 2014, Beijing, China

Background

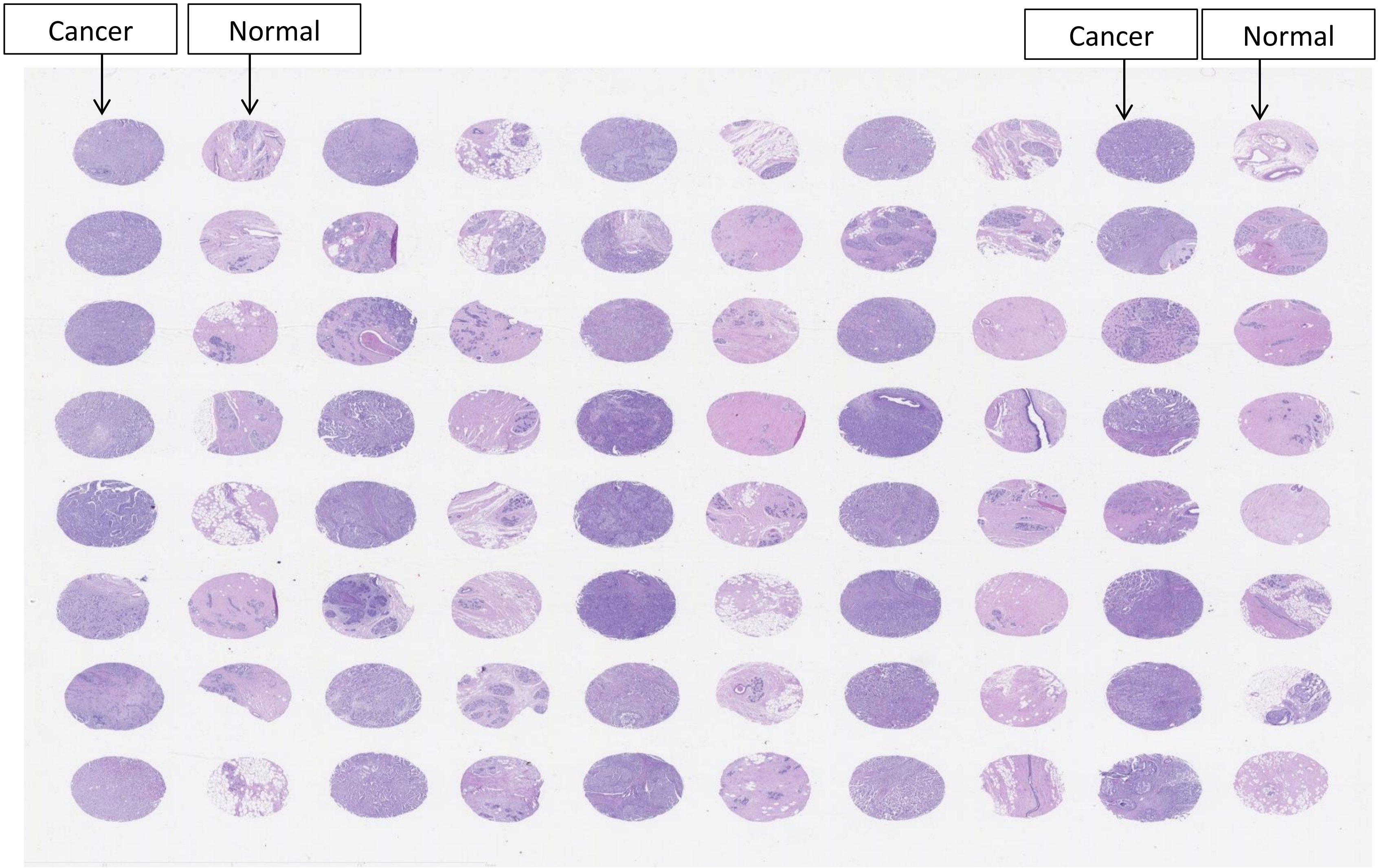
- Breast Cancer is one of the most frequent occurring cancers among women throughout the world
- Cancer diagnosis and prognosis
- Nottingham Prognostic Index (NPI)
- NPI= tumour diameter/5 + lymph node stage + **tumour grade**
  - NPI categorizes patients in three prognosis groups *Good*, *Intermediate* & *Poor*


Nottingham Grading System (NGS)
nuclear pleomorphism
tubular formation
mitotic count
Grade
1
Less aggressive appearance of tumour
2
Intermediate aggressive appearance of tumour
3
Highly aggressive appearance of tumour

Objective of Research

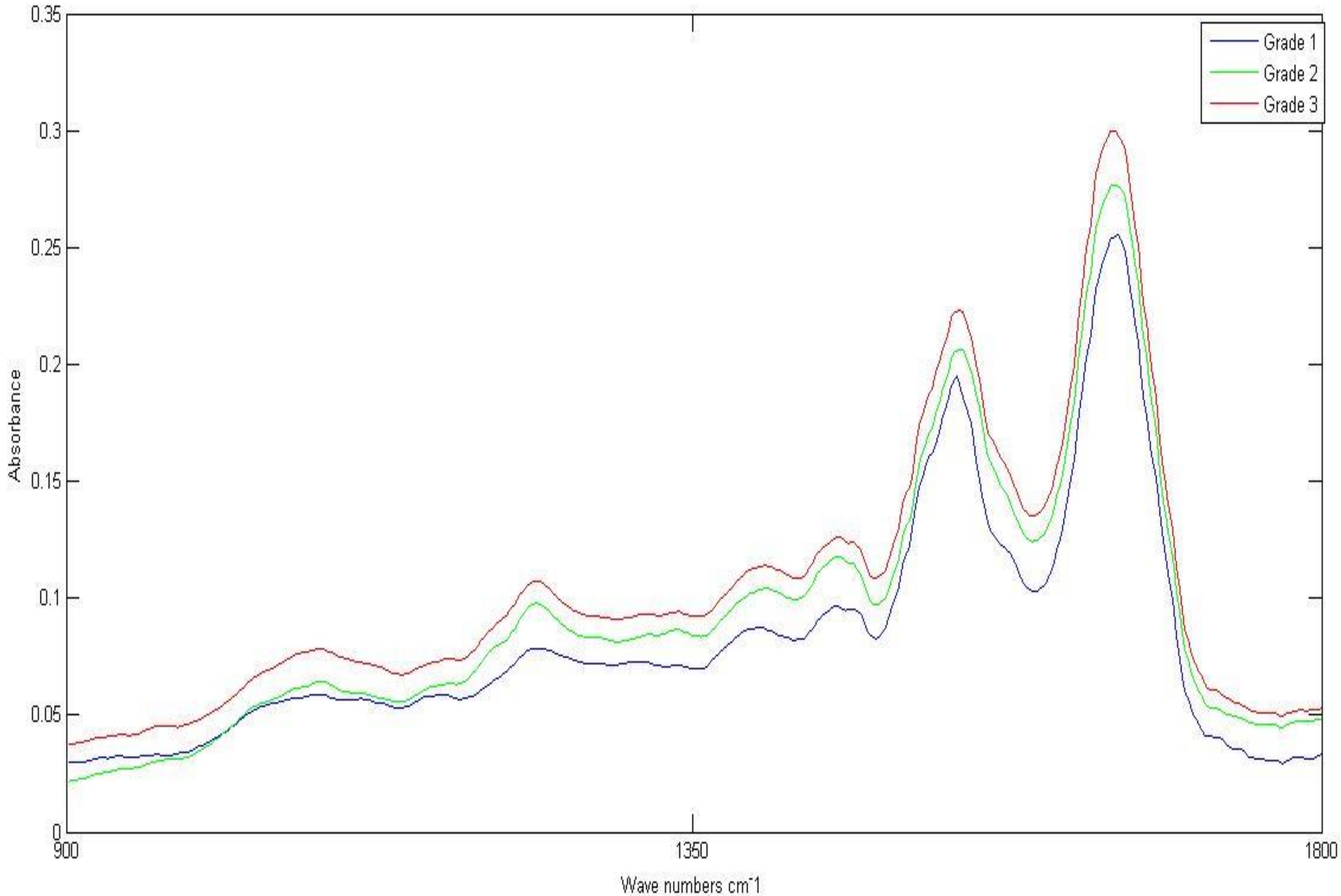
- FTIR (Fourier Transform Infra-red spectroscopy/microscopy) extracts a 'molecular fingerprint' of the sample after passing infra-red radiation though
  - FTIR has been increasingly applied to the study of biomedical problems including cancer
  - No two unique molecular structures produce the same infrared spectrum
- Can we characterise cancer grades using FTIR?
  - Using advanced fuzzy methods to represent the high levels of uncertainty observed in this context
  - Using the characterisation to automatically determine grade

Breast Tumour Tissue Microarray, Containing 40 Cases of Paired Breast Invasive Cancer



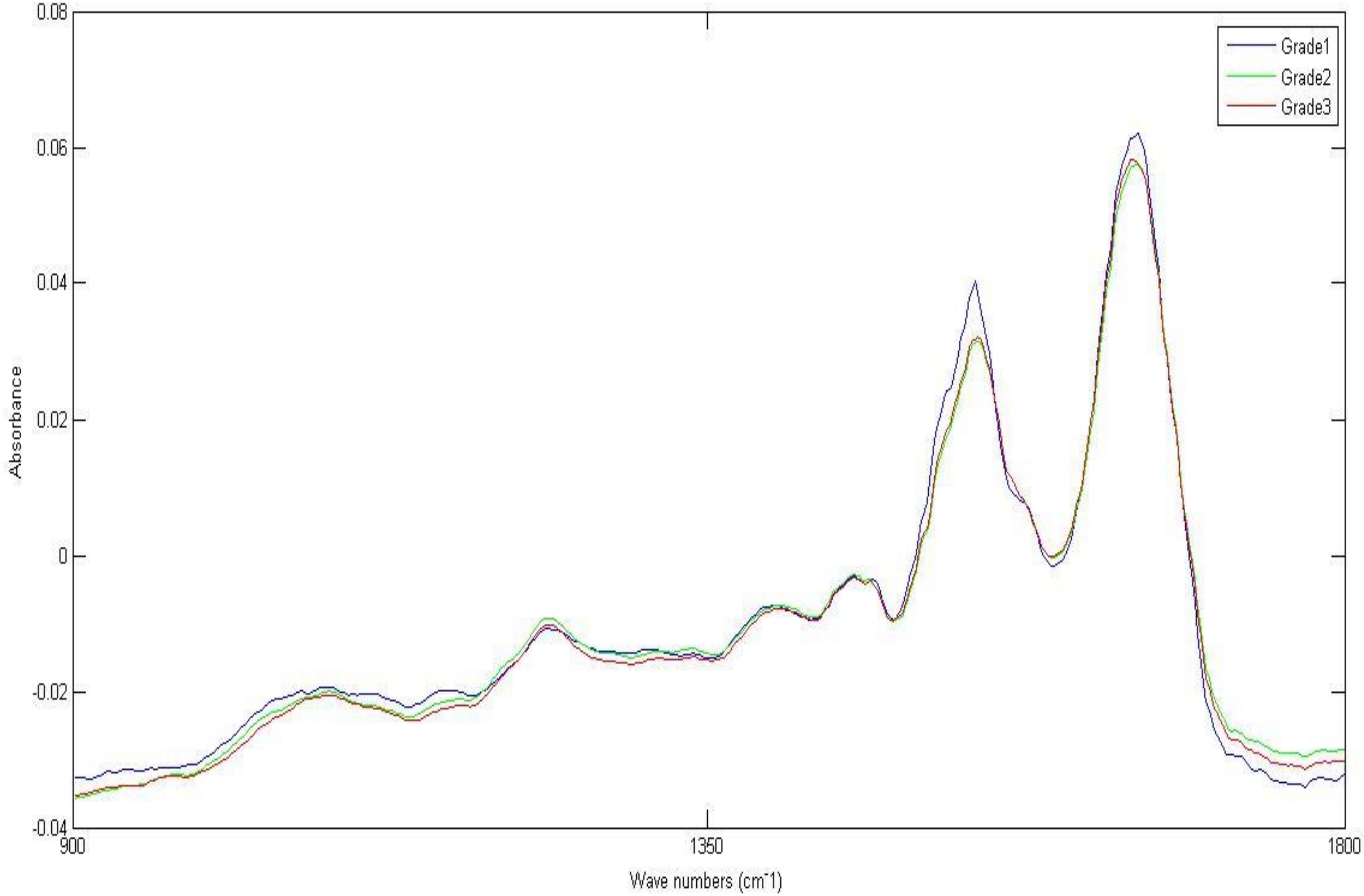
Example of a 10x10 Core Section

Raw Spectra



Grade 1
Grade 2
Grade 3
Absorbance
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
900
1350
1800
Wave numbers cm-1

Processed Spectra



Select five features for each grade

Features are based on absorbance values at different wave lengths in the spectral region

↓

Create 6 T-I fuzzy sets for each feature for each grade (G-I: 2 cases with 3 sets per case,

G-II: 6 sets from 6 cases, G-III: 6 sets from 6 cases)

↓

Combine 6 T-I fuzzy sets into a zGT-II fuzzy set with cases represented as z-axes (zSlices)

We have one zGT-II fuzzy set per feature per grade with 6 zSlices

Each zGT-II fuzzy set of a grade is considered as a prototype for that grade for a particular feature

↓

Select spectra from unseen data for any grade for each of the five features

Create a T-I fuzzy set with 30 spectra per feature

Use weighted zsliced based similarity measure to find similarity between created T-I fuzzy set and zSlices of prototype zGT-II fuzzy sets for each feature for each grade.

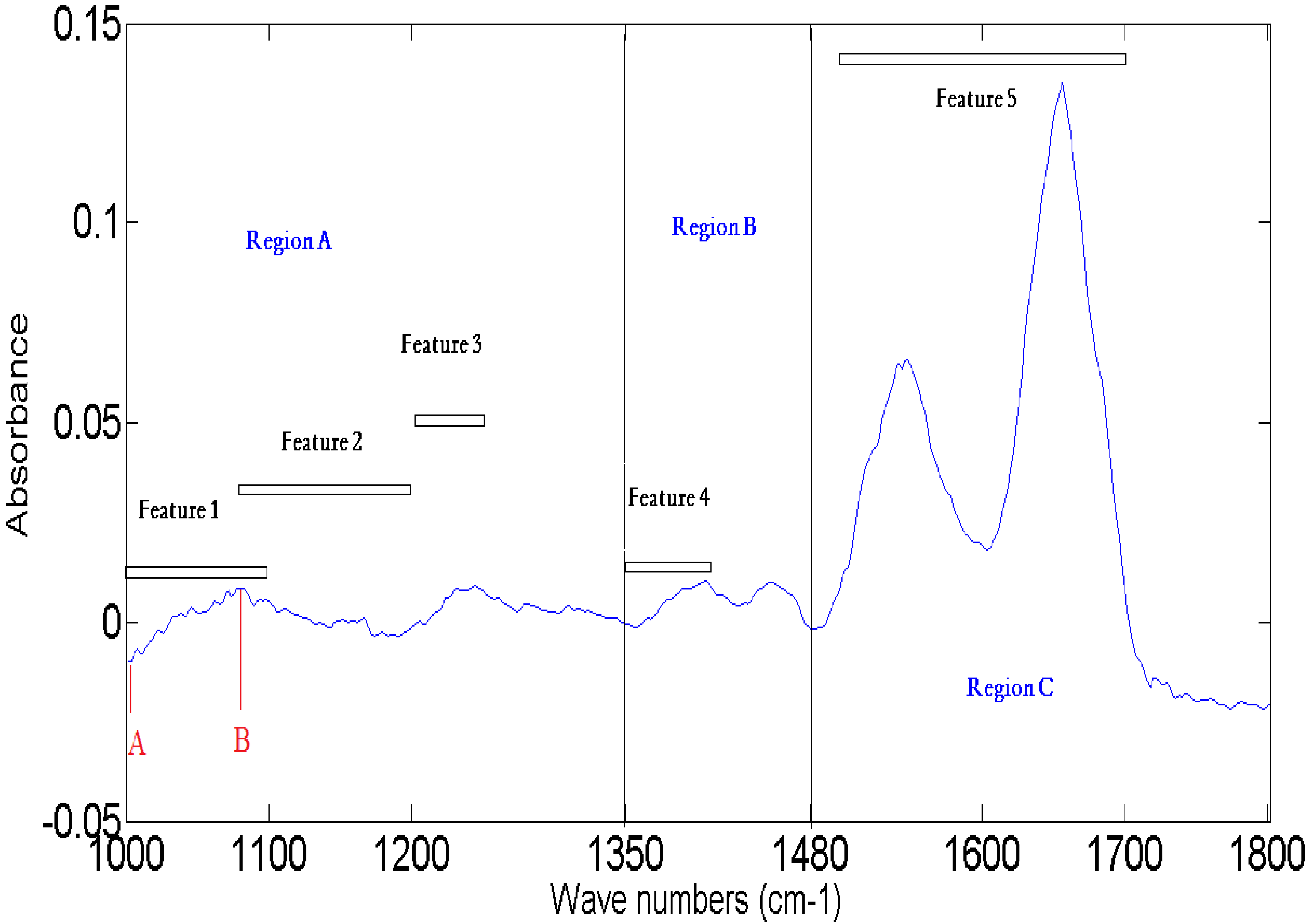
A similarity score for zSlices for each feature for each grade is obtained by this method

Based on the results of similarity measure, predict the grade of unseen data by using the following methods

- **Maximum sum of summation of similarity scores**
- **Majority vote**

Discuss the results of similarity scores for each feature for each grade and classification method

Describe a Grade profile for each grade on the basis of features and similarity scores

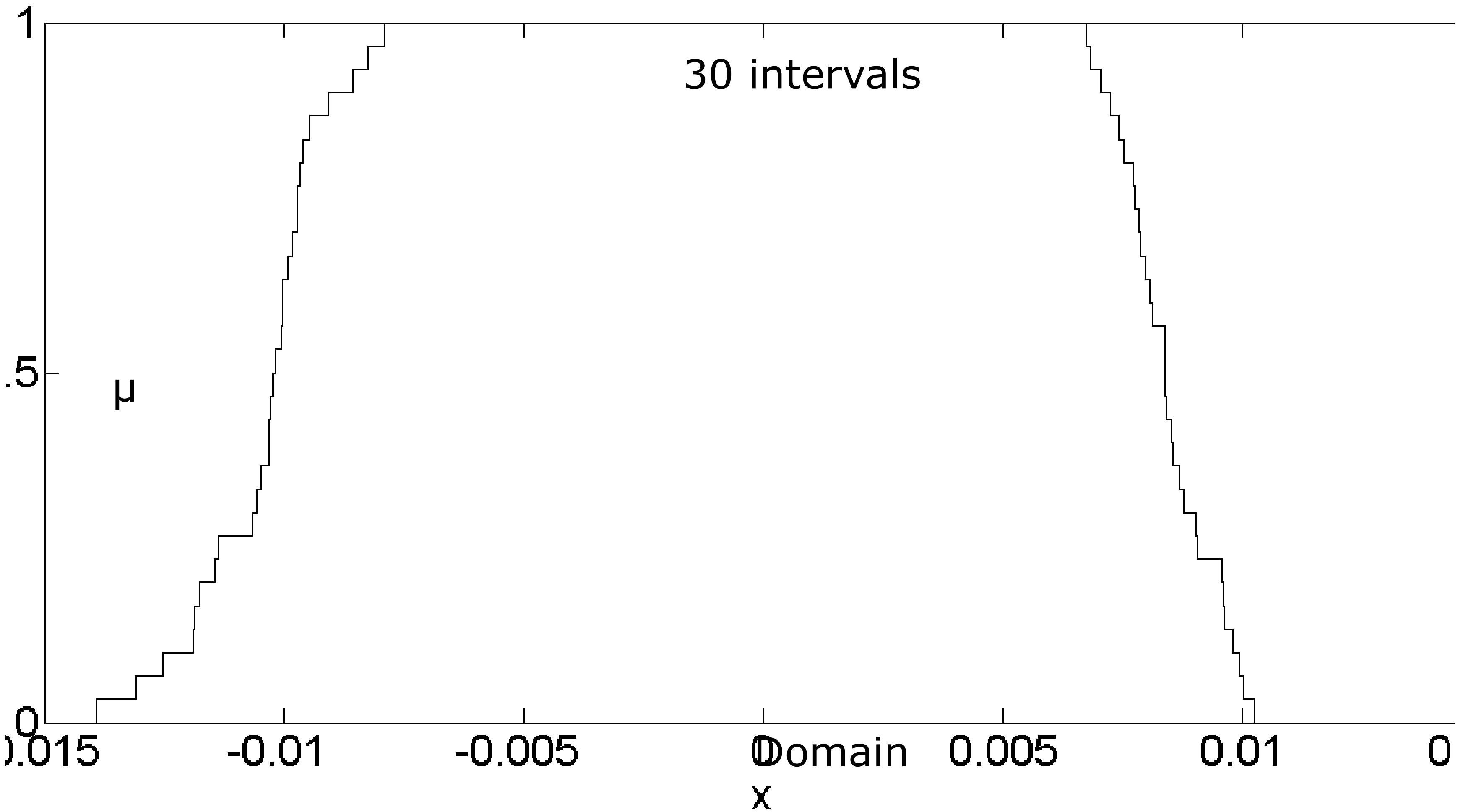


0.15
0.1
0.05
0
-0.05
Absorbance
Region A
Region B
Region C
Feature 1
Feature 2
Feature 3
Feature 4
Feature 5
A
B
1000
1100
1200
1350
1480
1600
1700
1800
Wave numbers (cm-1)

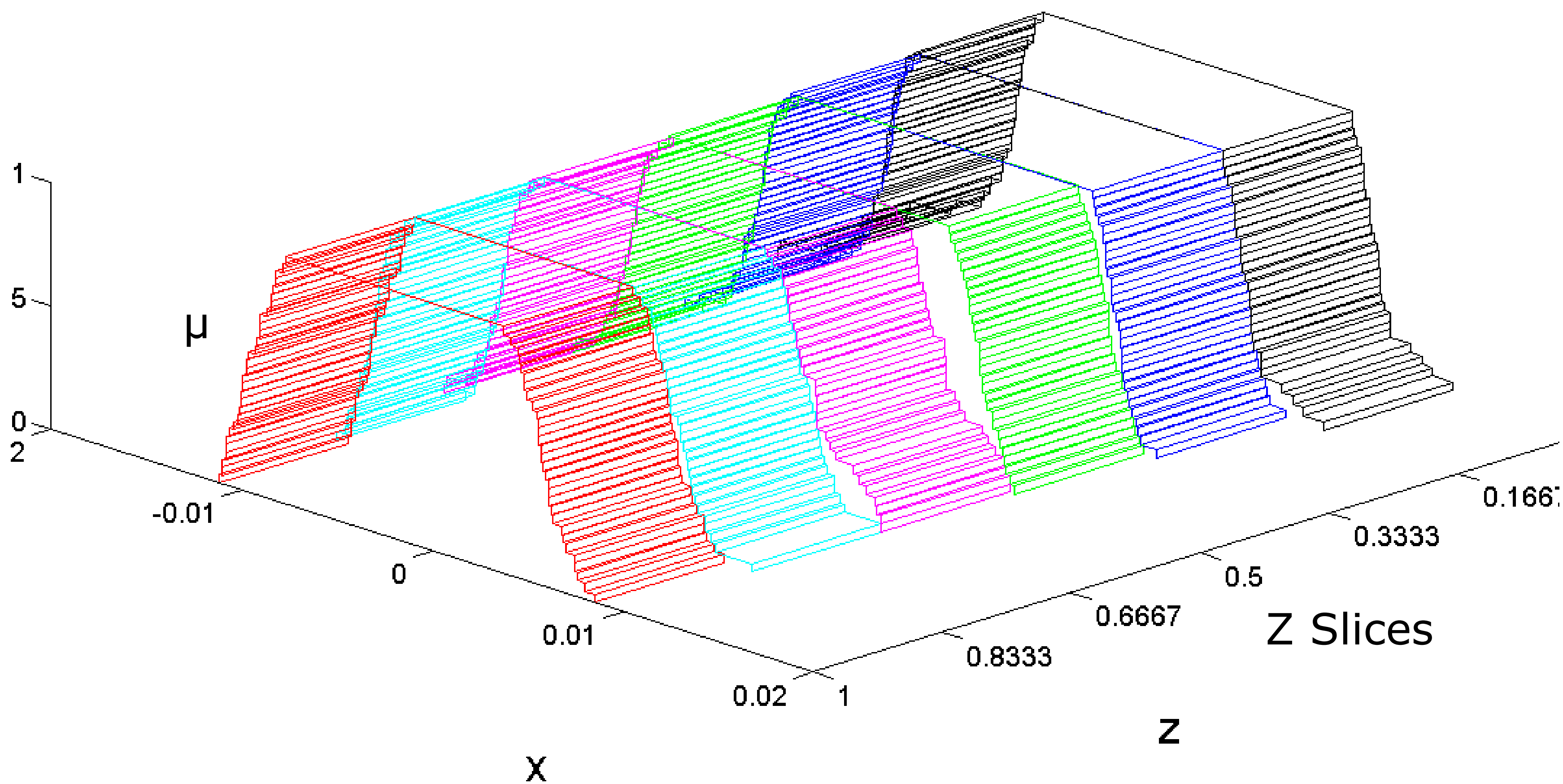
Similarity Measures for zGT2 Sets

- Representation of degree to which fuzzy sets are similar
- Method used is developed by McCulloch et al. (2013)
  - J McCulloch, C Wagner, U Aickelin, "Extending similarity measures of interval type-2 fuzzy sets to general type-2 fuzzy sets", FUZZ-IEEE 2013, 1-8
- The method results in a similarity score between zero and one
  - zero indicates completely disjoint sets and one indicates completely similar sets

Example of a Type-I Fuzzy Set



Example zGT2 Set for a Feature



Result Test Cases Grade 1

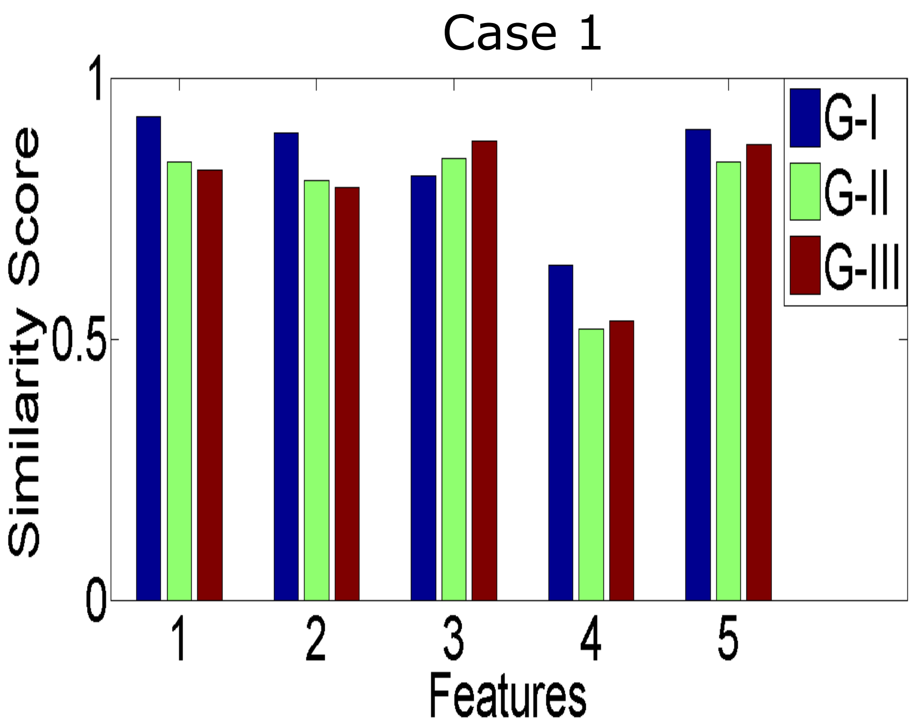
Case 1

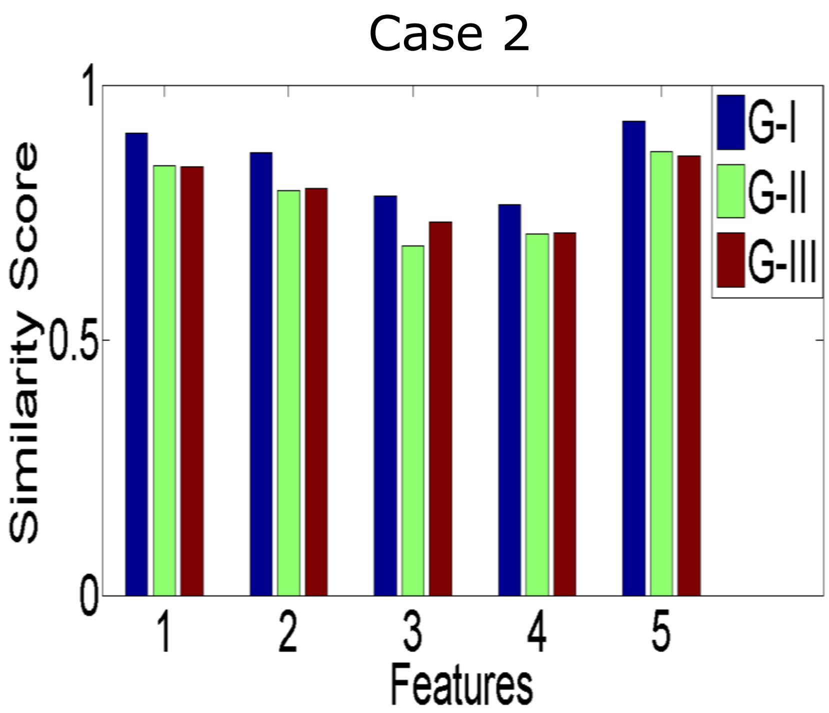
Feature	G-I	G-II	G-III
1	**0.9264**	0.8397	0.8235
2	**0.8938**	0.8035	0.7905
3	0.8122	0.8452	**0.8790**
4	**0.6407**	0.5194	0.5347
5	**0.9001**	0.8391	0.8719
Sum	**4.1732**	3.8469	3.8996
Majority vote	**W**	L	L

Case 2

Feature	G-I	G-II	G-III
1	**0.9047**	0.8424	0.8235
2	**0.8681**	0.7935	0.7905
3	**0.7816**	0.6838	0.7319
4	**0.7653**	0.7089	0.7102
5	**0.9283**	0.8684	0.8617
Sum	**4.248**	3.8970	3.9421
Majority vote	**W**	L	L

Grade Profiles (G-I)





Summary of Classification Methods

Summation

Type	Cases	Correct	Incorrect
G-I	2	2	0
G-II	6	1	5
G-III	6	6	0

Majority Vote

Type	Cases	Correct	Incorrect
G-I	2	2	0
G-II	6	2	3 (1 Tie)
G-III	6	3	1 (2 Tie)

Conclusions

- Extracting features in terms of intervals from FTIR spectra
- A new method based on Type-II Fuzzy sets (zSlices approach) for breast cancer grade classification
- Results were appreciable for G-I and G-III
- G-II was found to be most difficult to categorise (More uncertainty between cases)

Global Top 100 University

The University of Nottingham

UNITED KINGDOM · CHINA · MALAYSIA

Summary and Questions

Jon Garibaldi, Robert John and Christian Wagner

L U C I D

Lab for
Uncertainty in Data
and Decision Making

http://lucidresearch.org

Global
Top 100
University

The University of Nottingham
UNITED KINGDOM · CHINA · MALAYSIA

ASAP research group

horizon
DIGITAL ECONOMY RESEARCH

ima
Intelligent
Modelling & Analysis

Jon Garibaldi
https://scholar.google.co.uk/citations?user=L_9C4v0AAAAJ

Robert John
https://scholar.google.co.uk/citations?user=33ftCdEAAAAJ

Christian Wagner
https://scholar.google.co.uk/citations?user=tAQTJRIAAAAJ

L U C I D

Lab for
Uncertainty in Data
and Decision Making

http://lucidresearch.org