Combining the α-Plane Representation with an Interval Defuzzification Method

Sarah Greenfield[1] **Francisco Chiclana**[2]

[1,2]Centre for Computational Intelligence, De Montfort University, Leicester, LE1 9BH, UK

Abstract

This paper is concerned with the defuzzification of the discretised generalised type-2 fuzzy set.

In 2008 Liu proposed the α-Plane Representation — a decomposition of the generalised type-2 set into horizontal slices termed 'α-planes'. An α-plane is akin to an interval type-2 fuzzy set. The α-Plane Representation must be used in conjunction with an interval defuzzification method: The three main options are 1. the Karnik-Mendel Iterative Procedure, 2. the Greenfield-Chiclana Collapsing Defuzzifier, or 3. the Nie-Tan Method. The experiments recorded in this paper address the question, "Which is the best interval defuzzification method for the α-Plane Representation to be combined with?"

Keywords: type-reduction, α-Plane Representation, Karnik-Mendel Iterative Procedure, Greenfield-Chiclana Collapsing Defuzzifier, Nie-Tan Method

1. Introduction

The main strength of type-2 fuzzy logic[1] is its ability to deal with second-order uncertainties. Most researchers concentrate exclusively on the comparatively simple interval secondary membership functions [2], for which an increasing number of applications are being developed [3], [4], [5], [6], [7], [8], [9]. The capability of the generalised type-2 paradigm to handle uncertainty is explored in [10]. Since they lack the variability of the third dimension [2], interval type-2 fuzzy sets are not able to model uncertainty as subtly as their generalised counterparts. We therefore see developing *generalised* type-2 systems as an important goal. A triangular type-2 system with a defuzzification algorithm based on the Karnik-Mendel Iterative Procedure [11] has been developed by Starczewski [12]; this goes some way towards achieving our aim. Coupland and John [13], [14], have exploited geometry to improve the speed of inferencing in generalised type-2 fuzzy sets.

There are five stages to any FIS: fuzzification, antecedent computation, implication, aggregation and defuzzification (figure 1). For a type-2 FIS defuzzification consists of two parts — *type-reduction* and defuzzification proper. Type-reduction is the procedure by which a type-2 fuzzy set is converted to a type-1 fuzzy set. This set is then defuzzified to give a crisp number.

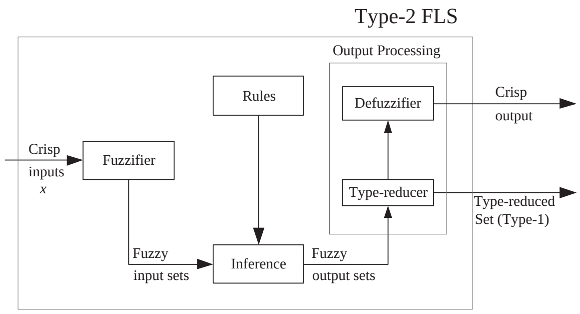


Figure 1: Type-2 FIS (from Mendel [15]).

The research reported in this paper concerns type-reduction, which in generalised type-2 fuzzy inferencing has been regarded as a computational bottleneck because of the high cardinality of the totality of embedded sets. A faithful implementation of the generalised type-reduction algorithm, known as *exhaustive defuzzification* (subsection 3), requires that every embedded set be processed. The finer the discretisation, the better the representation of a given fuzzy set, but the greater the number of embedded sets generated. Literally trillions upon trillions of embedded sets can be produced from unremarkable starting conditions. With such vast numbers of embedded sets, a defuzzification algorithm that works through each one is impractical.

Overcoming this bottleneck requires the development of efficient generalised type-2 defuzzification techniques with high accuracy. In the last five years two such strategies have been devised: 1. In 2005 Greenfield devised the Sampling Method [16] which is a direct, stochastic approach, in which a random sample of the many embedded sets is selected and processed. 2. In 2008 Liu published his strategy based on the α-Plane Representation (APR) [17], which is the topic of this paper. The APR operates in conjunction with an interval defuzzification method, and thus extends interval defuzzification to generalised type-2 sets. Liu envisaged that the APR be employed in association with the Karnik-Mendel Iterative Procedure [17, page 2225]. However two other candidates for the interval defuzzifier deserve consideration as potential partners for the APR. They are the Greenfield-Chiclana Collapsing Defuzzifier (GCCD) and the Nie-Tan Method (NTM). The purpose of this piece of research is to

[1]For further details on the history and background of type-2 fuzzy logic, see [1].

compare experimentally the accuracy and speed of the following three strategies:

- APR with KMIP,
- APR with GCCD, and
- APR with NTM.

Overview of the Paper The next section of this paper deals with some terminology, assumptions and definitions. Section 3 describes the accurate but slow exhaustive defuzzification strategy. Following that, section 4 presents the APR, after which section 5 introduces the trio of defuzzification techniques applicable to the interval type-2 set. Section 6 describes the experimental comparison between the interval methods, the results of which are presented in section 7.1 together with the conclusions drawn. The figures depicting the test sets and the tables of results appear at the end of the paper.

2. Preliminaries

2.1. Terminology

Each point on the surface of the type-2 set is referenced by co-ordinates deriving from the x, u, and z-axes. The x-axis is often referred to as the *primary domain*, and the u-axis as the *secondary domain*.

2.2. Assumptions

Scaling With no loss of generality we assume that the type-2 fuzzy set is contained within a unit cube.

Discretisation The work presented here is concerned only with defuzzification of *discretised* type-2 fuzzy sets.

Centroid Method of Defuzzification It is assumed that the centroid method of defuzzification [18, page 336] is used.

2.3. Definitions

Definition 1 (Slice) *A* slice *is a plane which intersects the primary or secondary domain (x or u-axis) perpendicularly.*

Definition 2 (Vertical Slice) *A* vertical slice *is a plane which intersects the primary domain (x-axis) perpendicularly.*

Definition 3 (Degree of Discretisation) *The* degree of discretisation *of a discretised fuzzy set is the separation of the slices.*

The primary and secondary domains may have different degrees of discretisation. Furthermore the secondary domain's degree of discretisation is not necessarily constant.

Scalar Cardinality For type-1 fuzzy sets, Klir and Folger ([19], p17) define scalar cardinality as follows:

Definition 4 (Scalar Cardinality) *The* scalar cardinality *of a fuzzy set A defined on a finite universal set X is the summation of the membership grades of all the elements of X in A. Thus,*

$$| A |= \sum_{x \in X} \mu_A(x). \qquad [19, page 17]$$

To distinguish scalar cardinality from cardinality in the classical sense, we adopt the '$\| \ \|$' symbol for scalar cardinality.

3. Exhaustive Defuzzification

Algorithm 1, known as *exhaustive defuzzification* is adapted from Mendel [15] and is the standard by which the accuracy of other defuzzification techniques may be assessed. It is termed the exhaustive method because *all* the embedded sets are required to be processed [20]. It is applicable to both generalised and interval type-2 fuzzy sets.

```
Input: a discretised generalised type-2
       fuzzy set
Output: a discrete type-1 fuzzy set
forall the embedded sets do
    find the minimum secondary
    membership grade {T-norms other than
    minimum may be employed} ;
    calculate the primary domain (x) value
    of the type-1 centroid of the type-2
    embedded set ;
    pair the secondary grade with the
    x-value to give set of ordered pairs (x, z)
    {some values of x may correspond to
    more than 1 value of z} ;
end
forall the primary domain (x) values do
    select the maximum secondary grade
    {make each x correspond to a unique
    secondary domain value} ;
end
```

Algorithm 1: Type-reduction of a discretised type-2 fuzzy set to a type-1 fuzzy set.

Though individually easily processed, embedded sets in their totality give rise to a processing bottleneck simply by virtue of their high cardinality. Consequently, exhaustive defuzzification is to be seen as a theoretical approach rather than a practical technique. At coarse discretisations this strategy may be implemented but is extremely slow (see subsection 7.1); at finer discretisations the issues of memory space and representation of very large numbers make implementation impossible.

We regard the exhaustive defuzzification algorithm as the standard by which other type-2 defuzzification algorithms must be evaluated.

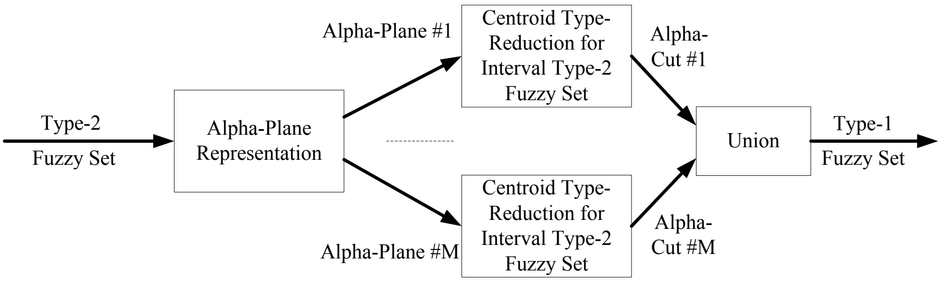


Figure 2: Defuzzification using the α-Planes Representation (from Liu [17]).

4. The α-Plane Representation

By repeated application of an interval defuzzification method, Liu [17] has shown that a generalised type-2 fuzzy set may be type-reduced. The generalised set is sliced horizontally into a number of α-planes. Each α-plane is defuzzified individually by means of an interval method. The set of ordered pairs $\{$(defuzzified value, α-plane height)$\}$ forms a type-1 fuzzy set. This is a method of type-reduction (figure 2). By defuzzifying this type-1 set, the defuzzified value for the generalised type-2 set is obtained.

5. Interval Type-Reduction Strategies

In this section we describe three approaches to interval type-reduction. These may be used to defuzzify interval type-2 sets, or, in conjunction with the APR, may be used to defuzzify generalised type-2 fuzzy sets. Two of the options, the KMIP and the GCCD, are iterative. The other alternative, the NTM, is non-iterative.

5.1. The Karnik-Mendel Iterative Procedure

The Karnik-Mendel Iterative Procedure (KMIP) [11] is the most widely adopted method for type-reducing an interval type-2 fuzzy set. The result of type-reduction of an interval type-2 fuzzy set is an interval type-1 set where the centroid lies between the two endpoints. The iterative procedure is an efficient search algorithm for finding these endpoints. The centroid of the type-1 set (i.e. the defuzzified value of the type-2 set) is taken to be the centre of this interval.

As the procedure works by finding the mid-point[2] of the TRS interval without taking account of the distribution of the values along the interval, there is inevitably an element of approximation inherent in the method.

[2] There is no need to calculate the midpoint when the KMIP is combined with the APR, as the endpoints themselves may be inserted into the type-1 set resulting from the decomposition into α-planes. This saves a little processing time.

5.2. The Greenfield-Chiclana Collapsing Defuzzifier

Another computationally simple alternative to the exhaustive method is the *Greenfield-Chiclana Collapsing Defuzzifier* [21]. This technique converts a discretised interval type-2 fuzzy set into a type-1 fuzzy set known as the *representative embedded set approximation (RESA)*, depicted in figure 3[3]. As a type-1 set, the RESA may then be defuzzified straightforwardly. The principle of the collapsing

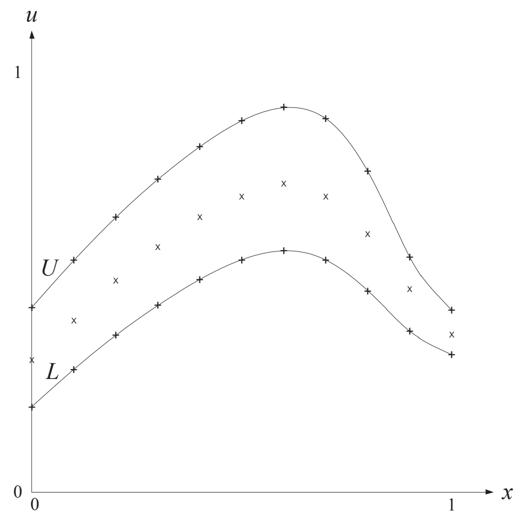


Figure 3: The Representative Embedded Set Approximation. Domain degree of discretisation = 0.1. The RESA is indicated by the line of crosses.

algorithm is that one by one each vertical slice is collapsed onto a point (located between the lower and upper membership functions). This produces a type-1 set.

We formally state the Simple[4] Representative Embedded Set Approximation:

[3] The RESA approximates to the the *representative embedded set (RES)*, whose defuzzified value is equal to that of the original type-2 set.

[4] In [21], we used the term 'simple' to describe an interval set in which each vertical slice consists of only two points, corresponding to L and U. The term is redundant in the context of this paper.

Theorem 1 (Simple Rep. Embedded Set Approx.) *The membership function of the* embedded set R *derived by dynamically collapsing slices of a discretised type-2 interval fuzzy set* $\tilde{F}$*, having lower membership function* L*, and upper membership function* U*, is:*

$$\mu_R(x_i) = \mu_L(x_i) + r_i \tag{1}$$

with

$$r_i = \frac{\left(\|L\| + \sum_{j=1}^{i-1} r_j\right) b_i}{2\left(\|L\| + \sum_{j=1}^{i-1} r_j\right) + b_i},$$

and $b_i = \mu_U(x_i) - \mu_L(x_i),\ r_0 = 0.$

The proof of the collapsing algorithm may be found at [21].

5.2.1. Collapsing Variants

Equation (1), the formula for collapsing, is in fact only *a version* of collapsing — the most intuitive variant, whereby the slices are collapsed in the order of increasing domain value ($x = 0$ to $x = 1$). We term this *collapsing forward.* However slice collapse may be performed in any slice order giving slightly different RESAs. These in turn give slightly different defuzzified values.

In [22] we concluded that *collapsing outward* gave the most accurate results. This comes in two forms, *outward right* and *outward left.* In outward right the middle slice is collapsed first, then the one to its right, then the one to the left of the middle, and so on until all the slices have been collapsed. Outward left is the opposite of outward right, i.e. the second slice collapsed is to the left of the middle. There is nothing to choose between these two forms of collapsing outward. We therefore employ a two pass strategy, *collapsing outward right-left*, in which collapsing outward right and collapsing outward left are performed sequentially, and their results averaged.

5.3. The Nie-Tan Method

Nie and Tan [23] describe an efficient type-reduction method for interval sets, which involves taking the mean of the lower and upper membership functions of the interval set, so creating a type-1 fuzzy set. Symbolically, $\mu_N(x_i) = \frac{1}{2}(\mu_L(x_i) + \mu_U(x_i))$, where N is the resultant type-1 set.

6. Experimental Comparison of the Methods

6.1. Experimental Methodology

The above three interval type-reduction strategies in conjunction with the APR were evaluated for both accuracy and speed. Three test sets were created:

Heater Setting Test Set This is a Mamdani FIS generated test set, produced by the aggregation stage of a prototype FIS whose purpose is to determine the setting for a heater. It has 5 rules and 2 inputs which are summarized in Table 1. The primary and secondary degrees of discretisation are 0.125. This is not a normal[5] type-2 fuzzy set; its highest secondary grade is 0.6806.

Washing Powder FIS Test Set This is a Mamdani FIS generated test set, produced by the aggregation stage of a prototype FIS whose purpose is to determine the amount of washing powder required by a washing machine for a given wash load. It has 4 rules and 3 inputs which are summarized in Table 2. The primary and secondary degrees of discretisation are 0.1. This test set is not normal; the highest secondary grade is 0.9167.

Shopping FIS Test Set This is a Mamdani FIS generated test set, produced by the aggregation stage of a prototype FIS designed to answer the dilemma of whether to go shopping by car, or walk, depending on weather conditions, amount of shopping, etc.. The defuzzified value is therefore rounded to one of two possible answers representing either 'by car' or 'on foot'. The FIS has 4 rules and 3 inputs as tabulated in Table 3. The primary and secondary degrees of discretisation are 0.1. This type-2 set is normal as it contains secondary membership grades of 1.

Tables 1 to 3 show the rules sets used by the Matlab^{TM} FIS[6] in producing the test sets. In figures 4 to 6, the 3-D representation and FOU of each test set is depicted. As the degrees of discretisation are coarse, the images look rather rough, but still give an indication of the shape of the test sets.

INPUTS		OUTPUTS
Temperature	Date	Heating
cold	—	high
—	winter	high
hot	not winter	low
—	spring	medium
—	autumn	medium

Table 1: Heater FIS rules

Each test set was defuzzified using Matlab^{TM} implementations of five type-2 defuzzification strategies:

1. Generalised exhaustive,

[5] A normal type-2 fuzzy set is one in which at least one secondary membership grade is 1.

[6] The FIS was run up to the end of the inference stage, producing aggregated sets to be defuzzified.

INPUTS			OUTPUTS
Washing	Water	Pre-Soak	Powder
very dirty	—	—	a lot
—	hard	—	a lot
slightly dirty	soft	—	a bit
—	—	lengthy	a bit

Table 2: Washing Powder FIS rules

INPUTS			OUTPUTS
Distance	Shopping	Weather	Travel Method
short	light	—	walk (< 0.5)
long	—	—	use car (> 0.5)
—	heavy	—	use car (> 0.5)
—	—	raining	use car (> 0.5)

Table 3: Shopping FIS rules

2. APR with KMIP,
3. APR with GCCD,
4. APR with NTM, and
5. APR with interval exhaustive[7].

To begin with each test set was exhaustively defuzzified and the defuzzified values and defuzzification times recorded. Then each APR/interval method combination was invoked with differing numbers of α-planes ranging from 3 to 100001 and the defuzzified values obtained tabulated (tables 5 to 7). The total run time for the defuzzification of each test set using each method combination was recorded in table 8. The code for the tests may be accessed on [24].

The tests were run on a PC with an Intel(R) Core(TM)2 Duo CPU, a clock speed of 2.93 GHz, a 2.96 GB RAM, running the MS Windows XP Professional 2002 operating system. Matlab^{TM} was run as a process with a priority that was higher than that of the operating system, so as to eliminate any timing errors caused by other operating system processes.

7. Results and Conclusions

7.1. Results

Table 4 shows the defuzzified values and timings from the exhaustive defuzzification of the test sets. These defuzzified values are used as a benchmark for accuracy. Tables 5 to 7 show the defuzzified values of the three test sets for the four different APR/interval method combinations. In the case of a non-normal type-2 set, the number of α-planes actively used in the processing is not always as high

[7]The APR with interval exhaustive defuzzification was included in the experiments for its value in shedding light on the convergence of the results.

Test Set	Defuzzified Value	Defuzzification Time
heater	0.6254971587	0.677822 seconds
powder	0.2896069752	1.391705 seconds
shopping	0.5931875193	11.96 minutes

Table 4: Generalised exhaustive defuzzification results for the three test sets.

as the the number of α-planes in the APR. This is the reason for the inclusion of the second column in the tables, showing the number of α-planes actually used. Table 8 shows the defuzzification times for each test set under each APR/interval combination. The timings were taken for the *total* sequence of test runs for each test set under each APR/interval method combination.

7.2. Conclusions

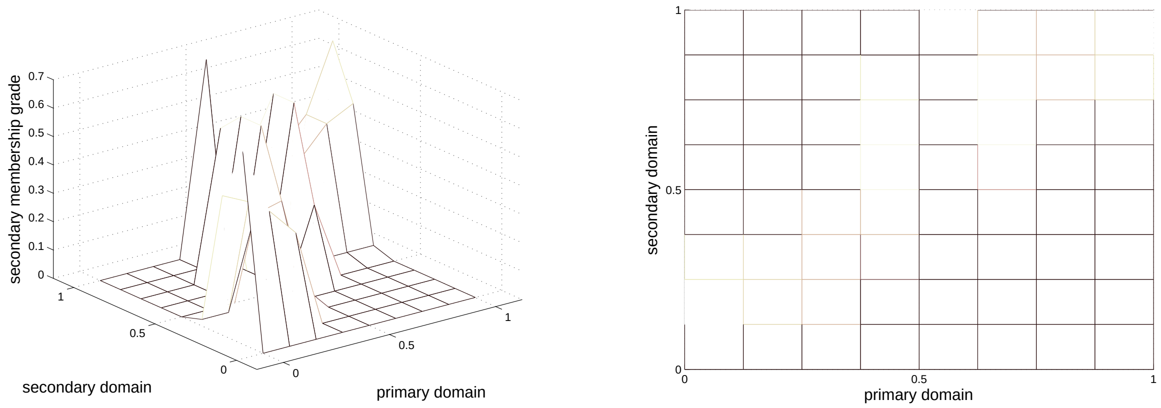
From the results of our experiments we draw the following conclusions:

Convergence of APR Combinations For all four APR/interval method combinations, as the number of α-planes increases, the defuzzified values tend to oscillate before converging. In each case the value they appear to converge to is different from the (generalised) exhaustive defuzzified value. This is to be expected in the cases of the APR + KMIP, the APR + GCCD and the APR + NTM, as these three interval methods are themselves approximate. It is somewhat surprising that this happens when the APR is partnered with interval exhaustive defuzzification.

Accuracy When the number of α-planes is low, the errors are erratic and no conclusion may be drawn as to which combination is the most accurate. However when the number of α-planes is 11 or higher we observe that the APR + GCCD performed better than the APR + KMIP and the APR + NTM. Indeed the APR + GCCD performed marginally better than the APR + interval exhaustive.

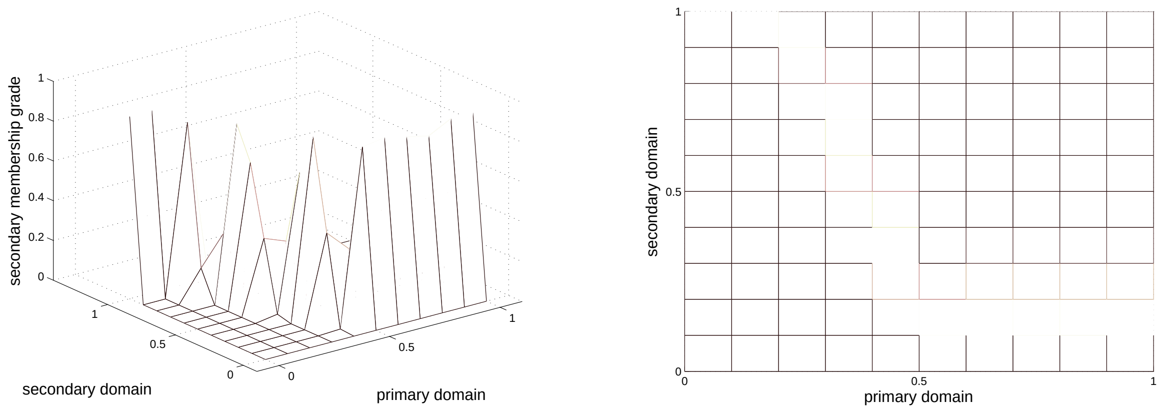
Defuzzification Times The primary focus of this investigation was in relation to accuracy; however timings of the test runs were recorded. The APR + NTM combination is the fastest, the APR + GCCD combination takes about 20% longer, and the APR + KMIP[8] combination takes about 5 times as long.

[8]The KMIP has been enhanced [25] to give the same results as the original KMIP used here, but in a computation time 39% less. However, had the enhanced algorithms been used in these experiments, the KMIP would still have been the slowest of the three practical methods.



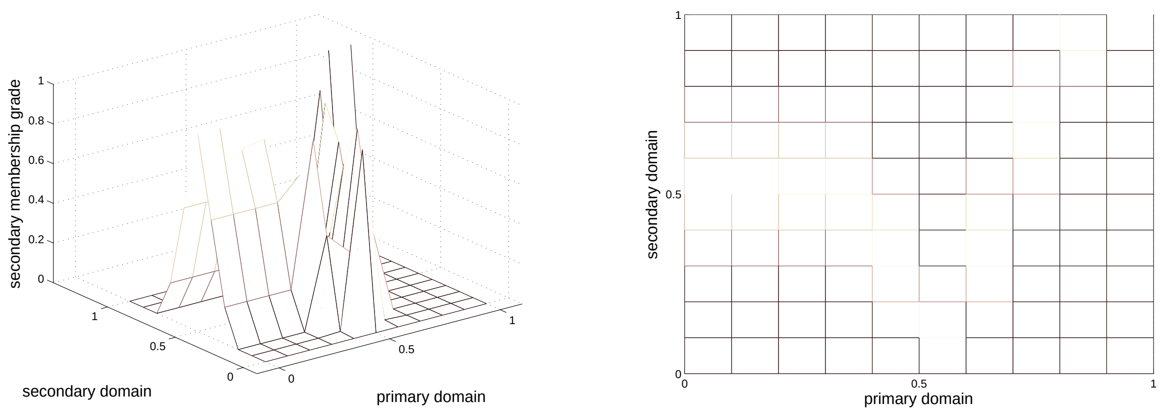
(a) 3-D representation (b) FOU

Figure 4: Heater test set: aggregated type-2 fuzzy set; degree of discretisation of the primary and secondary domains is 0.125.



(a) 3-D representation (b) FOU

Figure 5: Washing powder test set: aggregated type-2 fuzzy set; degree of discretisation of the primary and secondary domains is 0.1.



(a) 3-D representation (b) FOU

Figure 6: Shopping test set: aggregated type-2 fuzzy set; degree of discretisation of the primary and secondary domains is 0.1.

No. of α-Planes	No. of α-Planes Used	α-Planes & KMIP	Error α-Planes & KMIP	α-Planes & GCCD Outward Right-Left	Error α-Planes & GCCD	α-Planes and Nie-Tan Method	Error α-Planes & Nie-Tan Method	α-Planes and Exhaustive Method	Error α-Planes & Exhaust. Method
3	3	0.5935330	-0.0319642	0.5974412	-0.0280560	0.5972222	-0.0282749	0.5974396	-0.0280576
5	4	0.5933797	-0.0321174	0.6014929	-0.0240043	0.6012226	-0.0242746	0.6014844	-0.0240127
9	7	0.6149383	-0.0105588	0.6220020	-0.0034951	0.6218307	-0.0036664	0.6219955	-0.0035017
11	8	0.6119623	-0.0135348	0.6202109	-0.0052863	0.6200068	-0.0054904	0.6202020	-0.0052952
21	15	0.6096300	-0.0158671	0.6176530	-0.0078442	0.6174595	-0.0080377	0.6176442	-0.0078530
51	36	0.6067308	-0.0187663	0.6149639	-0.0105333	0.6147760	-0.0107212	0.6149553	-0.0105419
101	70	0.6063363	-0.0191609	0.6146819	-0.0108153	0.6144948	-0.0110024	0.6146732	-0.0108239
1001	682	0.6065354	-0.0189618	0.6149166	-0.0105805	0.6147268	-0.0107703	0.6149079	-0.0105893
10001	6808	0.6066098	-0.0188873	0.6149818	-0.0105153	0.6147921	-0.0107051	0.6149731	-0.0105240
100001	68061	0.6066101	-0.0188871	0.6149819	-0.0105153	0.6147921	-0.0107051	0.6149732	-0.0105240

Table 5: Defuzzified values obtained for the Heater Test Set. (Exhaustive defuzzified value = 0.6254971587.)

No. of α-Planes	No. of α-Planes Used	α-Planes & KMIP	Error α-Planes & KMIP	α-Planes & GCCD Outward Right-Left	Error α-Planes & GCCD	α-Planes and Nie-Tan Method	Error α-Planes & Nie-Tan Method	α-Planes and Exhaustive Method	Error α-Planes & Exhaust. Method
3	3	0.2987646	0.0091576	0.3100683	0.0204614	0.3102564	0.0206494	0.3100715	0.0204645
5	5	0.2930690	0.0034621	0.2990423	0.0094353	0.2991462	0.0095392	0.2990447	0.0094377
9	9	0.2905533	0.0009464	0.2949802	0.0053732	0.2950580	0.0054510	0.2949820	0.0053750
11	11	0.2818122	-0.0077947	0.2860659	-0.0035410	0.2861420	-0.0034649	0.2860678	-0.0035392
21	20	0.2861117	-0.0034953	0.2903153	0.0007084	0.2903923	0.0007853	0.2903173	0.0007103
51	47	0.2887201	-0.0008869	0.2928824	0.0032755	0.2929596	0.0033527	0.2928845	0.0032775
101	93	0.2868682	-0.0027388	0.2909067	0.0012997	0.2909814	0.0013744	0.2909086	0.0013017
1001	918	0.2868148	-0.0027922	0.2907821	0.0011752	0.2908556	0.0012486	0.2907841	0.0011771
10001	9168	0.2867620	-0.0028450	0.2907216	0.0011146	0.2907949	0.0011879	0.2907235	0.0011165
100001	91671	0.2867607	-0.0028463	0.2907192	0.0011122	0.2907925	0.0011856	0.2907212	0.0011142

Table 6: Defuzzified values obtained for the Washing Powder Test Set. (Exhaustive defuzzified value = 0.2896069752.)

No. of α-Planes	No. of α-Planes Used	α-Planes & KMIP	Error α-Planes & KMIP	α-Planes & GCCD Outward Right-Left	Error α-Planes & GCCD	α-Planes and Nie-Tan Method	Error α-Planes & Nie-Tan Method	α-Planes and Exhaustive Method	Error α-Planes & Exhaust. Method
3	3	0.6110598	0.0178723	0.6151870	0.0219995	0.6150809	0.0218934	0.6151852	0.0219977
5	5	0.5979297	0.0047422	0.6018755	0.0086880	0.6017783	0.0085908	0.6018736	0.0086861
9	9	0.5890816	-0.0041059	0.5932603	0.0000727	0.5931403	-0.0000472	0.5932572	0.0000697
11	11	0.5905091	-0.0026784	0.5946488	0.0014612	0.5945378	0.0013502	0.5946460	0.0014585
21	21	0.5882242	-0.0049633	0.5929872	-0.0002003	0.5928530	-0.0003345	0.5929838	-0.0002037
51	51	0.5868927	-0.0062948	0.5920148	-0.0011727	0.5918663	-0.0013212	0.5920111	-0.0011765
101	101	0.5867273	-0.0064602	0.5919492	-0.0012383	0.5917992	-0.0013883	0.5919455	-0.0012420
1001	1001	0.5862202	-0.0069674	0.5914404	-0.0017472	0.5912909	-0.0018966	0.5914366	-0.0017509
10001	10001	0.5861870	-0.0070006	0.5914135	-0.0017741	0.5912639	-0.0019236	0.5914097	-0.0017778
100001	100001	0.5861793	-0.0070082	0.5914059	-0.0017816	0.5912563	-0.0019312	0.5914021	-0.0017854

Table 7: Defuzzified values obtained for the Shopping Test Set. (Exhaustive defuzzified value = 0.5931875193.)

Test Set	Degree of Discretisation	Number of Embedded Sets	α-Planes & KMIP	α-Planes & GCCD	α-Planes & NT Method	α-Planes and Exhaustive Method
heater	0.125	14580	129.0 secs.	32.3 secs.	25.5 secs.	24.4 minutes
powder	0.1	24300	242.6 secs.	62.6 secs.	52.1 secs.	133.6 minutes
shopping	0.1	312500	290.3 secs.	74.4 secs.	63.7 secs.	145.7 minutes

Table 8: Defuzzification times for the three test sets.

References

[1] R. I. John and S. Coupland. Extensions to Type-1 Fuzzy Logic: Type-2 Fuzzy Logic and Uncertainty. In Gary Y. Yen and David B. Fogel, editor, *Computational Intelligence: Principles and Practice*, pages 89–102. IEEE Computational Intelligence Society, 2006.

[2] J. M. Mendel and R. I. John. Type-2 fuzzy sets made simple. *IEEE Transactions on Fuzzy Systems*, 10(2):117–127, 2002.

[3] H. Hagras and C. Wagner. Introduction to interval type-2 fuzzy logic controllers — Towards better uncertainty handling in real world applications. *IEEE Systems, Man and Cybernetics eNewsletter*, 2009. Issue 27.

[4] E. A. Jammeh, M. Fleury, C. Wagner, H. Hagras, and M. Ghanbari. Interval type-2 fuzzy logic congestion control for video streaming across IP networks. *IEEE Transactions on Fuzzy Systems*, 17:1123–1142, 2009. issue 5.

[5] S. M. Miller, V. Popova, R. I. John, and M. Gongora. An interval type-2 fuzzy distribution network. In *Proc. 2009 IFSA World Congress/EUSFLAT Conference*, pages 697–702, Lisbon, July 2009.

[6] J. M. Garibaldi and P. A. Birkin. A novel dual-surface type-2 controller for micro robots. In *Proceedings of FUZZ-IEEE 2010*, pages 359–366, Barcelona, Spain, 2010.

[7] M. Zaher, H. Hagras, A. Khairy, and M. Ibrahim. A type-2 fuzzy logic based model for renewable wind energy generation. In *Proceedings of FUZZ-IEEE 2010*, pages 511–518, Barcelona, Spain, 2010.

[8] E. Kayacan, O. Kaynak, R. Abiyev, J. Tørresen, M. Høvin, and K. Glette. Design of an adaptive interval type-2 fuzzy logic controller for the position control of a servo system with an intelligent sensor. In *Proceedings of FUZZ-IEEE 2010*, pages 1125–1132, Barcelona, Spain, 2010.

[9] L. Leottau and M. Melgarejo. Implementing an interval type-2 fuzzy processor onto a DSC 56F8013. In *Proceedings of FUZZ-IEEE 2010*, pages 1939–1942, Barcelona, Spain, 2010.

[10] S. Greenfield and R. I. John. The Uncertainty Associated with a Type-2 Fuzzy Set. In *Rudolf Seising (editor) Views on Fuzzy Sets and Systems from Different Perspectives Philosophy and Logic, Criticisms and Applications, Studies in Fuzziness and Soft Computing vol. 243, Springer-Verlag*, pages 471–483, 2009.

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

[12] J. T. Starczewski. Efficient triangular type-2 fuzzy logic systems. *International Journal of Approximate Reasoning*, 50(5):799–811, May 2009.

[13] S. Coupland and R. I. John. Fuzzy logic and computational geometry. In *Proceedings of RASC 2004*, pages 3 – 8, Nottingham, England, December 2004.

[14] S. Coupland and R. I. John. Geometric type-1 and type-2 fuzzy logic systems. *IEEE Transactions on Fuzzy Systems*, 15(1):3–15, February 2007.

[15] J. M. Mendel. *Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions*. Prentice-Hall PTR, 2001.

[16] S. Greenfield, R. I. John, and S. Coupland. A novel sampling method for type-2 defuzzification. In *Proc. UKCI 2005*, pages 120–127, London, September 2005.

[17] F. Liu. An efficient centroid type-reduction strategy for general type-2 fuzzy logic system. *Information Sciences*, 178:2224–2236, 2008.

[18] G. J. Klir and B. Yuan. *Fuzzy Sets and Fuzzy Logic*. Prentice-Hall P T R, 1995.

[19] George J. Klir and T. A. Folger. *Fuzzy sets, Uncertainty, and Information*. Prentice-Hall International, 1992.

[20] S. Greenfield, F. Chiclana, and R. I. John. Type-reduction of the discretised interval type-2 fuzzy set. In *Proceedings of FUZZ-IEEE 2009*, pages 738–743, Jeju Island, Korea, August 2009.

[21] S. Greenfield, F. Chiclana, S. Coupland, and R. I. John. The collapsing method of defuzzification for discretised interval type-2 fuzzy sets. *Information Sciences: Special Section on Higher Order Fuzzy Sets*, 179(13):2055–2069, June 2008.

[22] S. Greenfield, F. Chiclana, and R. I. John. The collapsing method: Does the direction of collapse affect accuracy? In *Proceedings of IFSA-EUSFLAT 2009*, pages 980–985, Lisbon, Portugal, July 2009.

[23] M. Nie and W. W. Tan. Towards an efficient type-reduction method for interval type-2 fuzzy logic systems. In *Proceedings of FUZZ-IEEE 2008*, pages 1425–1432, Hong Kong, June 2008.

[24] Sarah Greenfield. Matlab code for general type-2 defuzzification, April 2011.

[25] D. Wu and J. M. Mendel. Enhanced Karnik-Mendel algorithms. *IEEE Transactions on Fuzzy Systems*, 17(4):923–934, 2009.