

# Novel Methods for the Design of General Type-2 fuzzy Sets based on Device Characteristics and Linguistic Labels Surveys

Christian Wagner    Hani Hagrass

The Computational Intelligence Centre, School of Computer Science and Electronic Engineering, University of Essex  
Colchester, United Kingdom

Email: chwagn@essex.ac.uk, hani@essex.ac.uk

**Abstract**—Fuzzy Logic Systems are widely recognized to be successful at modelling uncertainty in a large variety of applications. While recently interval type-2 fuzzy logic has been credited for the ability to better deal with large amounts of uncertainty, general type-2 fuzzy logic has been a steadily growing research area. All fuzzy logic systems require the accurate specification of the membership functions' (MFs) parameters. While some work for automatic or manual design of these parameters has been proposed for type-1 and interval type-2 fuzzy logic, the problem has not yet been widely addressed for general type-2 fuzzy logic. In this paper we propose two methods which allow the automatic design of general type-2 MFs using either data gathered through a survey on the linguistic variables required or, in the case of physical devices (e.g. sensors, actuators), using data directly gathered from the specific devices. As such, the proposed methods allow for the creation of general type-2 MFs which directly model the uncertainty incorporated in the respective applications. Additionally, we demonstrate how interval type-2, type-1 and the recently introduced zSlices based general type-2 MFs can be extracted from the automatically designed general type-2 MFs. We also present a recursive algorithm that computes the convex approximation of generated fuzzy sets.

**Keywords**— Type-2 fuzzy systems, general type-2 fuzzy sets, specification of fuzzy sets.

## 1 Introduction

Fuzzy Logic is widely applied because of its ability to model the uncertainties which are inherent to most real world applications and environments. The high levels of uncertainty encountered by such fuzzy logic systems can generally be attributed to one or more of the categories below:

- Uncertainties in inputs to FLCs which translate to uncertainties in the antecedent Membership Functions (MFs) as for example sensor measurements are typically noisy and affected by the conditions of observation (i.e. their characteristics are changed by the environmental conditions such as wind, sunshine, humidity, rain, etc.).
- Uncertainties in control outputs which translate to uncertainties in the consequent MFs of FLCs. Such uncertainties can result from the change of the actuators characteristics which can be due to wear, tear, environmental changes, etc.
- Linguistic uncertainties associated with the meaning of words (that are used as antecedents or consequents in FLCs or used as part of Computing With Words (CWW) applications) which can be uncertain as “words mean different things to different people” [1]. For example the perception of the linguistic label “warm” varies for

different people while also being heavily subject to context, e.g. “warm” in the Caribbean will usually be associated with a different temperature than “warm” at the North Pole etc.

While the capability of modelling uncertainty provided by type-1 fuzzy sets has been intensively researched and shown to provide good results, in recent years interval type-2 fuzzy sets have been found to provide an even higher potential at modelling large amounts of uncertainty [1],[2],[3],[4]. Additionally, a variety of significant advances has been made in the area of general type-2 fuzzy logic which is hoped to provide the framework for an accurate model of uncertainty.

One of the main challenges during the design of any fuzzy based system (general type-2, interval type-2 or type-1) is the specification of the fuzzy Membership Functions (MFs). The choice of type of membership function (such as gaussian, triangular, etc.) as well as the choice of their specific parameters strongly affects the performance of the fuzzy system. A variety of methods to alleviate this problem have been researched for mainly type-1 and interval type-2 FLCs. Such methods are generally based on the use of expert knowledge, evolutionary techniques (such as genetic algorithms) and neural networks [5], [6]. However, there is still much work needed to standardize and simplify the selection of appropriate MFs.

Recent developments in general type-2 fuzzy theory [7], [8], [9], [10], [11] have made general type-2 applications a realistic option for the future. Hence, we have developed novel techniques to determine the appropriate parameters for general type-2 MFs which due to their complex nature are highly difficult to specify. The proposed techniques are based on directly using existing knowledge about the uncertainty contained within the variables represented by the MFs. Additionally, the proposed techniques have the advantage that they can be used for the generation of all currently common types of fuzzy sets: type-1, interval type-2 and general type-2.

Section 2 describes the design of fuzzy MFs modelling linguistic concepts using surveys while Section 3 details the design of fuzzy MFs based on physical device or variable characteristics. Both sections present real world examples to demonstrate the techniques. Section 4 describes how to extract zSlice based general type-2, interval type-2 and type-1 MFs from the general type-2 MFs generated in Section 2 and 3. In Section 5 we present the recursive convexity algorithm employed throughout the paper and give a detailed example of its application. Finally, in Section 6 we present

the conclusions and address the work planned for the near future.

## 2 Design of general type-2 fuzzy sets based on the perception of linguistic labels determined through surveys.

### 2.1 The Human Perception of Linguistic Labels.

Fuzzy sets are frequently used to specify linguistic variables such as *temperature* or *height* used by people on an everyday basis. As such, the variable *temperature* can for example be modelled using three linguistic labels, *cold*, *moderate* and *warm* which in turn are represented using fuzzy MFs. While type-1 sets can be used to model such labels, it is clear that as type-1 MFs associate each input with a crisp membership value, hence a given type-1 MF will need to be constantly re-tuned for different perceptions of the same linguistic label. For example, the perception of the linguistic label *warm* can easily be seen to depend on factors such as the country of origin of a person (someone from Alaska will consider warm a different temperature than someone from the Caribbean), the age, sex, current season, etc.

Interval type-2 fuzzy sets can alleviate the above need for the continuous re-tuning of individual MFs as the incorporated Footprint of Uncertainty (FOU) allows for the model to include the full range of values associated with a specific linguistic label by different people in different contexts. Interval type-2 fuzzy sets nevertheless have the significant shortcoming that the uncertainty is spread evenly across the FOU which does not allow for variations of the uncertainty within this FOU to be modelled. For example, considering the example of *warm* temperature, if two people associate warm with 25°C and one person associates it with 28°C, a FOU incorporating 25°C and 28°C will not reflect that the majority of the people (here two) associate 25°C with the linguistic label *warm* - we have effectively lost this information.

This additional information can be modelled using general type-2 fuzzy sets by employing the third dimension associated with the FOU to encode the amount of uncertainty associated with each point within the FOU. In terms of our example, the FOU would be associated in its third dimension with more uncertainty towards 28°C and less uncertainty towards 25°C.

In the next few subsections we will describe how a survey on linguistic labels can be used to gather information on a specific linguistic variable using the example of *Indoor Temperature*. While this example is chosen to illustrate and simplify the process, the mechanism can be applied for any linguistic variable which is modelled by a series of consecutive linguistic labels. We will show how the information can be used to generate fuzzy MFs which model the linguistic labels and their associated uncertainty accurately using general type-2 fuzzy sets. In addition, we will present the mechanisms used to extract appropriate interval type-2 and type-1 MFs as well as zSlices based general type-2 MFs which were introduced in [11].

### 2.2 The survey on the linguistic variable “Indoor Temperature”.

As part of our work on the generation of MFs from surveys we have conducted a survey investigating the perception of

the linguistic variable *Indoor Temperature*. We have presented the participants with a questionnaire asking them to identify the position of the three linguistic labels *Cold*, *Moderate* and *Warm* in the context of an indoor environment on a temperature scale ranging from 0°C to 40°C. Further details on the survey and its results are given in Table 1. It should be noted that participants could supply additional information during the survey, specifically their age, sex and country of origin. While maintaining anonymity, we consider this information highly useful as it can be expected that factors such as the country of origin significantly affect people’s perception for example of temperature.

Table 1: Details on the linguistic survey on the perception of *Indoor Temperature*. (\* based on information supplied)

Number of participants:	62
Age range of participants:	18-58 years*
Mean age of participants:	30.145 years*
Number of different countries of origin:	22 countries*
Location of survey:	Colchester, UK
Date of survey:	January 2009
Minimum values with <i>cold</i> , <i>moderate</i> , <i>warm</i> :	0°C, 10°C, 16°C
Maximum values with <i>cold</i> , <i>moderate</i> , <i>warm</i> :	24°C, 27°C, 35°C
Mean values for <i>cold</i> , <i>medium</i> , <i>warm</i> :	12.062°C, 19.664°C, 25.664°C
Standard deviations for <i>cold</i> , <i>medium</i> , <i>warm</i> :	4.637, 3.552, 3.926

### 2.3 From a Survey to a General Type-2 Fuzzy Set model of the Linguistic Variable

In order to utilise the data provided through the survey to generate a general type-2 fuzzy set model of the linguistic variable which includes all the uncertainty information gathered in the data, we follow the following steps:

- (1) Map all the values associated with a specific linguistic label to the frequency with which the values occur in the data. This step can be likened to creating a continuous histogram for the data where the xAxis represents the values (in our case, the temperature) and the yAxis represents the number of occurrences. Fig. 1 shows this mapping for our data using interpolation between the data points.

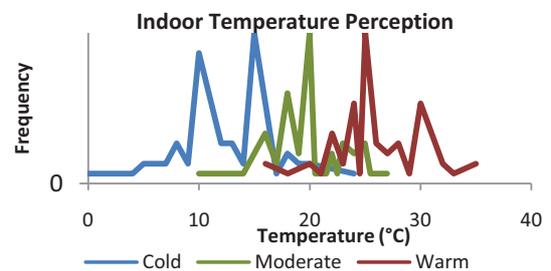


Figure 1: Mapping of survey data for the three linguistic labels *cold*, *moderate* and *warm*, showing the large amounts of uncertainty contained within the perceptions of the linguistic labels (e.g. large overlap).

- (2) The data mapped in Step (1) and illustrated in Fig. 2 illustrates the significant variation in the perception of the individual linguistic labels as well as their strongly differing interrelation for different people. In order to

utilize the information to define the third dimension of general type-2 fuzzy sets we normalise the frequency so it correlates with the uncertainty modelled in the third dimension of a fuzzy set, i.e. the more frequent a specific sample was identified in the survey, the less uncertainty is associated with it.

- (3) As convexity is a general requirement in several general type-2 fuzzy logic operations such as the join and meet computed on the membership grade presented in [12], we convert the normalised plots created in Step 2 to their convex approximation using a recursive algorithm which we describe in Section 5 in more detail. The original, normalised data as well as its convex approximations using our recursive algorithm and basic triangular approximation are shown for all three linguistic labels in Figs. 2a, 2b and 2c.

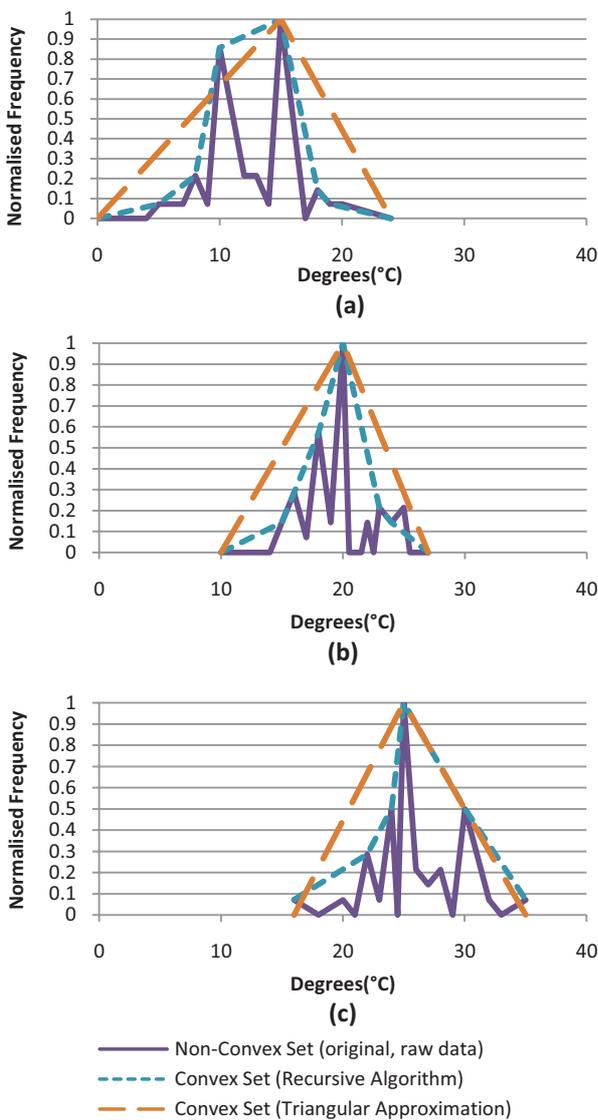


Figure 2: Representations of the linguistic labels, (a) H(cold), (b) H(moderate), (c) H(warm).

Fig. 2a, Fig. 2b and Fig. 2c clearly show that a more accurate convex approximation of the original sets is achieved when employing the proposed recursive algorithm (explained in Section 5) when compared to a simple triangular approximation. We refer to the

“histogram based” representations of the linguistic labels cold, moderate and warm as  $H(\text{cold})$ ,  $H(\text{moderate})$  and  $H(\text{warm})$ .

- (4) In order to create a fuzzy model of the linguistic variable *Indoor Temperature* we proceed one linguistic label at a time:

- a. For *cold*, we associate  $H(\text{cold})$  with the level 1 on the yAxis and  $H(\text{moderate})$  with level 0 on the yAxis. We employ interpolation to recreate the general type-2 set for *cold* as shown in Fig. 3a, Fig. 3b and Fig. 3c. It should be noted that the positioning of  $H(\text{cold})$  and  $H(\text{moderate})$  at this step is not arbitrary but follows logically from the linguistic labels, i.e.  $H(\text{cold})$  represents the temperature values associated with *cold*, hence they should be related to a yLevel of 1.  $H(\text{moderate})$  on the other hand represents the temperature values which are associated with *moderate* and as such not anymore related with *cold*, thus, in terms of the fuzzy set **cold**, it should be associated with yLevel 0 at  $H(\text{moderate})$ .

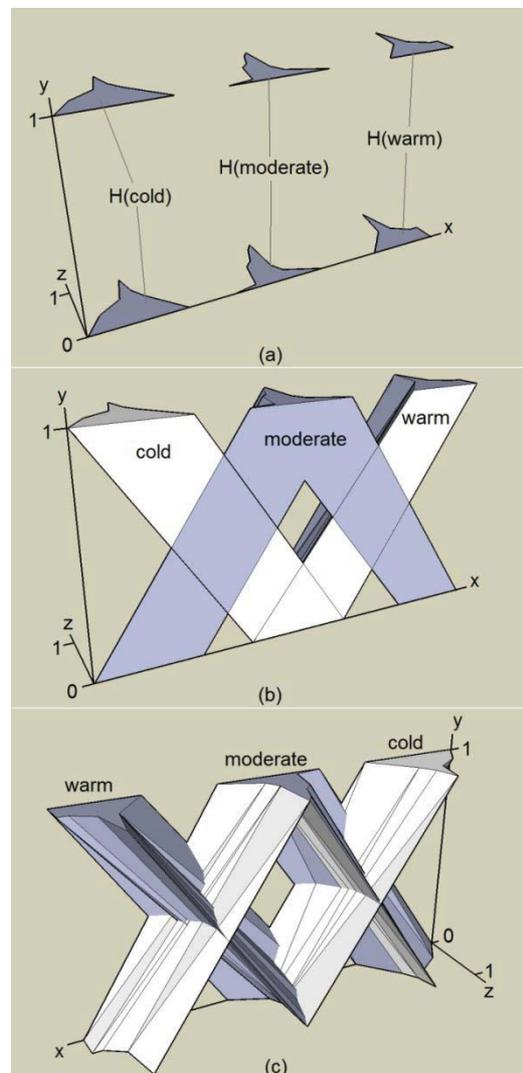


Figure 3: Schematic representation of the steps involved in creating the fuzzy set representation of linguistic labels. (a) “Histogram representations” of the linguistic labels associated with the respective levels on the yAxis before interpolation. (b) Fuzzy sets after interpolation from the front. (c) Fuzzy sets after interpolation from the rear.

- b. For *moderate*, we associate  $H(\text{cold})$  with the level 0 on the yAxis,  $H(\text{moderate})$  with level 1 on the yAxis and  $H(\text{warm})$  with yLevel 0 on the yAxis. We employ interpolation to recreate the general type-2 set for *moderate* as shown in Fig. 3a, Fig. 3b and Fig. 3c. The reasoning of the positioning on the yAxis of  $H(\text{cold})$ ,  $H(\text{moderate})$  and  $H(\text{warm})$  is analogous to Step a.
- c. For *warm*, we associate  $H(\text{moderate})$  with level 0 on the yAxis and  $H(\text{warm})$  with yLevel 1 on the yAxis. We employ interpolation to recreate the general type-2 set for *warm* as shown in Fig 3(a)-(c).

The reasoning of the positioning on the yAxis of  $H(\text{moderate})$  and  $H(\text{warm})$  is analogous to Step a.

We have shown in a series of steps how the data on a linguistic variable such as temperature can be used to generate a general type-2 representation of the given variable. It is straightforward to extract zSlices based general type-2 fuzzy sets as introduced in [11], interval type-2 and type-1 sets. The individual steps are shown for the general case in Section 4.

### 3 Design of fuzzy sets based on device or variable characteristics

In the context of Fuzzy Logic Controllers (FLCs), fuzzy logic sets are generally employed to model the uncertainty associated with the inputs and outputs of the FLC. In real world applications these inputs and outputs are commonly physical devices such as sensors and actuators which are susceptible to a wide range of sources for uncertainty, from wear and tear, to environmental conditions, signal noise etc. General type-2 fuzzy sets are well suited to model this uncertainty and in this section we are describing how information about the device or variable to be modelled can be directly incorporated into the fuzzy sets modelling. We will be using the illustrative example of generating the MFs for a sonar sensor from information which is directly retrieved from the device. While the example is referring to a sonar sensor, the overall method is applicable to every device or artefact that is to be modelled.

#### Generation of two MFs *Near* and *Far* which model the distance value measured using a Sonar sensor.

1. As a first step, we discretize the maximum sensor range into X steps. A finer level of discretization will result in a more accurate model of the uncertainty contained within the sensor outputs. The process of discretizing the sensor range of a Pioneer robot side sonar is shown in Fig. 4.
2. At every step, the device, in our case the sonar sensor is sampled (for example by presenting the sonar with an obstacle at the specified distance in our case) Y times. A larger number Y of samples will result in a more accurate model of the uncertainty contained within the

sensor outputs. The series of Y readings are analysed to create a mapping detailing the normalized frequency of each specific sample. As such, in our case, the reading (or close group of readings) which was measured the largest number of times out of Y samples is mapped to 1, readings that have never been measured are mapped to 0 etc. This approach can be likened to a continuous (high discretization level) histogram. An example of this can be seen in Fig. 4. At each sampling step x, Y samples are taken which are translated to the respective histograms  $H(x)$ .

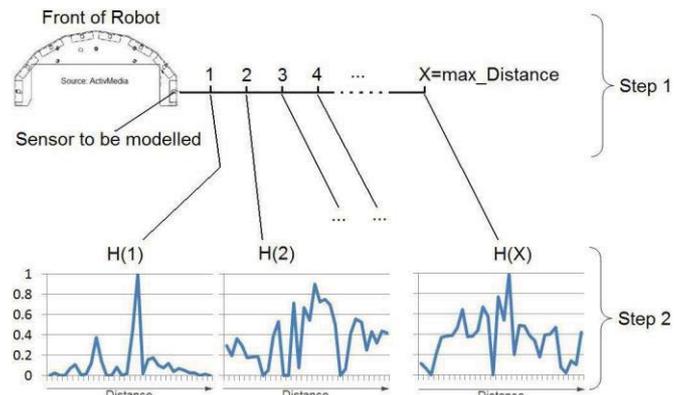


Fig. 4 Step 1 and 2 for the automatic generation of MFs for the sonar sensor mounted on a Pioneer robot.

3. The histograms generated in Step 2 are representative of the uncertainty encountered while sampling the specific distance x, i.e. while usually the majority of samples return the accurate reading, some will return a different reading because of sensor noise, environmental impact, etc. The information contained in the histograms can thus be employed to define the third dimension of general type-2 fuzzy sets which models the uncertainty within the sensor. In order to use the histograms to define the third dimension of general type-2 fuzzy sets, the histograms need to be strictly convex. The convexity of the third dimension of fuzzy sets is a general requirement in several general type-2 fuzzy logic operations such as the join and meet computed on the membership grade presented in [12]. While we are planning to address the direct handling of non-convex sets in the future, at this stage we are converting the existing histogram into a convex function. While this can be done through simple triangular approximation, this results in a very poor representation of the information contained in the initial histogram. We have devised a recursive algorithm which converts the histograms  $H(x)$  to their convex counterparts  $C(x)$  while preserving as much information as possible. An example comparing an initial histogram to its convex approximations using the triangular approach and our algorithm can be seen in Fig. 5 and an in depth

description of the algorithm and further comparisons can be found in Section 5.

It should be noted that the algorithm can be applied to any type-1 fuzzy set and as such to the histograms as presented here as well as any vertical slices of the general type-2 fuzzy set if required.

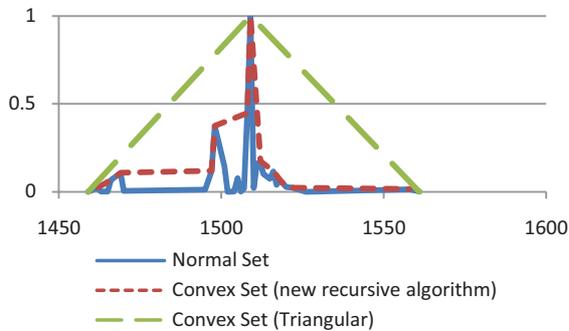


Figure 5: Comparison between an initial non-convex histogram (set) and its convex approximation using the triangular and our recursive algorithm approach.

- For each linguistic label, we will need to determine the position of the linguistic labels within the device input/output range. In our example we are using triangular MFs and as such would define three points for every linguistic variable: its start, its maximum and its end. Fig.6 shows the Near MF for which in our example we defined the start as  $N_s$ , the maximum as  $N_m$  and the end as  $N_e$ . All previously generated convex  $H(x)$  where  $x \in [\text{start}, \text{end}]$  are mapped to the triangle formed by  $\{\text{start}, \text{maximum}, \text{end}\}$ , “raising” them to the respective level on the yAxis. This process is illustrated in Fig. 6 for 5 histograms. In case there is uncertainty (for example due to linguistic uncertainties) about the positions of points  $N_s$ ,  $N_m$  and  $N_e$ , we will have various triangles, each one is associated with the relevant histograms as shown in Fig.6. These triangles are then interpolated (as explained in the following step) in the x-y and the y-z domain to generate the FOU and third dimension of the general type-2 fuzzy set.

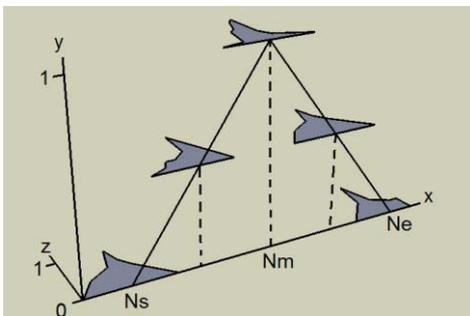


Fig. 6 The fuzzy MF *Near* is composed of the endpoints defined by the linguistic variable combined with the histograms of any sampling points which lie within the range of the linguistic variable. Here, five histograms are shown which have been “raised” to their respective level on the yAxis which is defined by the intersection of their sampling point  $x$  with the triangular  $\{N_s, N_m, N_e\}$ .

- In the final step towards the generation of a general type-2 MF, the different histograms are linked through interpolation. It should be noted that the choice of interpolation method and the number of histograms directly affect the quality of the uncertainty model of the MF. Fig. 7 illustrates the front and side view of our exemplary MF *Near*. If it is requirement for the vertical slices of the resulting set to be strictly convex, the same convexity algorithm as described above can be applied to the vertical slice.

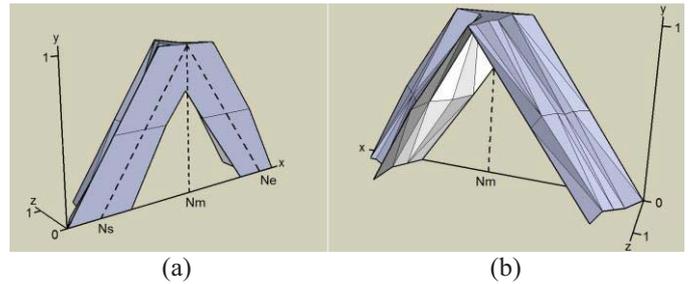


Figure 7: (a) Front view of the generated membership function for *Near*; (b) rear view of the same set.

The procedure for the automatic generation of general type-2 MFs presented in this section allows the generation of MFs which reflect and model the uncertainty encountered by or inherent to a specific device. The generated general type-2 MFs can easily be used to extract type-1, interval type-2 and zSlices based general type-2 fuzzy sets as we will be shown in Section 4.

#### 4 Extraction of zSlice based general, interval type-2 and type-1 fuzzy sets.

It is straightforward to extract type-1 MFs and interval type-2 MFs from the general type-2 MFs generated in Sections 2 and 3 by restricting the zLevel (i.e. the value on the zAxis) to 1 or 0 respectively (in the case of interval type-2 MFs, they are subsequently “raised” to a zLevel of 1). This allows the process to generate type-1 and interval type-2 MFs from data without altering the procedure, which should facilitate the performance comparison between different levels of complexity in FLCs. An example of the MFs extracted from the general type-2 fuzzy sets shown in Fig. 7 is shown in Fig. 8.

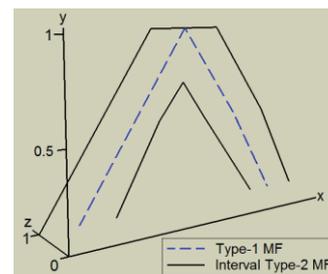


Figure 8: The type-1 and interval type-2 MFs extracted from the general type-2 MF shown in Fig. 7

Finally, it is worth noting that the fuzzy sets generated in Sections 2 and 3 can easily be transformed to zSlices based MFs which were introduced in [11]. Fig. 9 shows an example of the generated zSlice based MF from the general type-2 MF shown in Fig. 7.

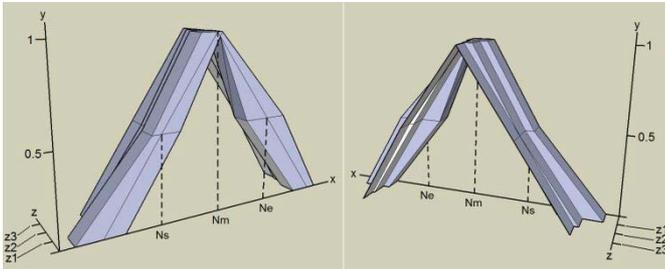


Figure 9: zSlice based general type-2 version (3 zSlices) of the general type-2 set shown in Fig. 15. It should be noted that only the zSlices 1-3 are shown, zSlice 0 has been omitted to improve visibility.

## 5 Approximation of a convex set from a non-convex set.

### 5.1 Description of the algorithm.

In order to generate a convex function from a non-convex function (more specifically, the discretized points from this function) while maintaining a high level of similarity to the original function, we have devised a recursive algorithm which is described in detail in this section.

The algorithm takes a discretized non-convex function (such as a type-1 fuzzy set)  $F$  as input. The discretization level along the xAxis is referred to as  $D$ . The function  $F$  matches each point on the xAxis  $x_d$  to a point  $y_d$  on the yAxis (where  $d \in [1, D]$ ). As such, the function is represented as a series of points  $P_d$  in two-dimensional space where the x and y coordinates of a point  $P_d$  are referred to as  $P_{d_x}$  and  $P_{d_y}$ .

In our application of the convex approximation of “continuous histograms” (as discussed in Sections 2 & 3), the x-Axis represents the input axis, for example the distance, if we are considering a distance sensor, while the yAxis represents the number of samples. We denote the convex output set as  $O$ .

The recursive algorithm performs the following steps:

0.
  - a. Traverse set from left to right (arbitrary- or from right to left) and find the coordinates of the discretized point  $P_{Max}$  ( $P_{Max}$  is defined in two-dimensional space by its coordinates  $P_{Max_x}$  and  $P_{Max_y}$ ), where
 
$$P_{Max} \in P_D \text{ and } P_{Max_y} = \max(P_{d_y}), d \in [1, D].$$
 If more than one point with the same maximum is found, the middle point (according to its x coordinates  $P_{d_x}$ ) is chosen. If the number of points with the same maximum is even, one of the middle points is chosen arbitrarily.
  - b. Add point  $P_{Max}$  to the output set  $O$ .

- c. Split the series of points into two series:  $\{P_1, \dots, P_{Max}\}$  and  $\{P_{Max}, \dots, P_D\}$ .
- d. Pass the series  $\{P_1, \dots, P_{Max}\}$  to Step 1 and the series  $P_{Max}, \dots, P_D\}$  to Step 2.
  1.
    - a. Refer to the given series of points as  $[P_{start}, P_{end}]$ .
    - b. *IF* ( $P_{start} == P_{end}$ ) *THEN STOP*.
    - c. Find the new maximum point  $P_{Max'}$  of the given series such that:
 
$$P_{Max'_y} > P_{start_y} \text{ AND } P_{Max'_y} < P_{end_y}$$

$$P_{Max'_y} = \max(P_{d_y}), d \in [start, end].$$
    - d. Add point  $P_{Max'}$  to the output set  $O$ .
    - e. Split the given series into two series:  $\{P_{start}, \dots, P_{Max'}\}$  and  $\{P_{Max'}, \dots, P_{end}\}$  and pass both series separately to Step 1 (recursion).
  2.
    - a. Refer to the given series of points as  $[P_{start}, P_{end}]$ .
    - b. *IF* ( $P_{start} == P_{end}$ ) *THEN STOP*.
    - c. Find the maximum point  $P_{Max'}$  of the given series with endpoints such that
 
$$P_{Max'_y} < P_{start_y} \text{ AND } P_{Max'_y} > P_{end_y}$$

$$P_{Max'_y} = \max(P_{d_y}), d \in [start, end].$$
    - d. Add point  $P_{Max'}$  to the output set  $O$ .
    - e. Split the given series into two series:  $\{P_{start}, \dots, P_{Max'}\}$  and  $\{P_{Max'}, \dots, P_{end}\}$  and pass both series separately to Step 2 (recursion).

Below we give an example of the application of the recursive algorithm.

### 5.2 Example application of the recursive algorithm.

In Table 2 we indicate the coordinates of a non convex set shown in Fig.10 taken from real world experiments and compare it to the coordinates computed by our algorithm and the triangular approximation.

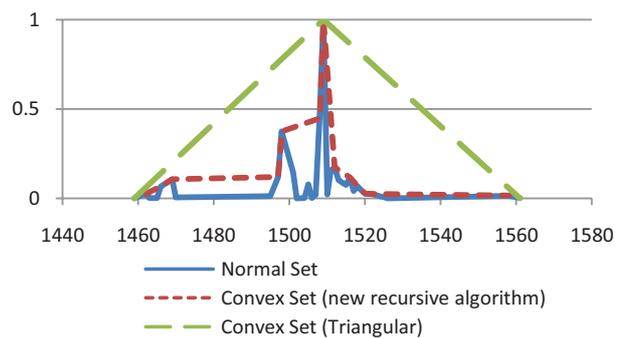


Fig. 10 A plot of the non convex input function as well as both the recursive and triangular approximated

From Table 2 it can be seen that both the recursive and the triangular algorithms omit a significant amount of detail in order to generate a convex approximation of the input set. Nevertheless the number of preserved samples and thus detail and information is much larger in the recursive approximation. This is also visible in Fig. 10 which shows that the recursive algorithm provides a vastly superior fit to

the original function in comparison to the triangular approximation.

Table 2 Comparison of z and y tuples preserved by our recursive algorithm and a triangular algorithm while converting the non-convex set shown in Fig. 10.

Non-convex Set				Convex Set (Recursive Algorithm)		Convex Set (Triangular Algorithm)	
x	y	x	y	x	y	x	y
0.00	1459	0.44	1508	0.02	1462	0	1459
0.02	1462	1.00	1509	0.06	1466	1	1509
0.00	1463	0.02	1510	0.10	1469	0	1561
0.00	1465	0.15	1511	0.11	1497		
0.06	1466	0.17	1512	0.37	1498		
0.10	1469	0.10	1513	0.44	1508		
0.00	1470	0.07	1515	1.00	1509		
0.01	1495	0.11	1516	0.17	1512		
0.11	1497	0.04	1517	0.11	1516		
0.37	1498	0.06	1518	0.06	1518		
0.14	1501	0.04	1519	0.04	1519		
0.00	1502	0.02	1520	0.02	1520		
0.00	1504	0.02	1522	0.02	1522		
0.07	1505	0.00	1526	0.01	1558		
0.00	1506	0.01	1558				
0.02	1507	0.00	1561				

### 6 Conclusions

In this paper we have addressed the reoccurring problem in fuzzy systems of specifying the parameters for fuzzy membership functions (MFs). The correct selection of parameter of fuzzy MFs is crucial in order for the fuzzy systems to model the uncertainty contained within the system correctly and thus to allow the generated fuzzy system to provide good performance. While the problem has been previously addressed to some extent for interval type-2 and type-1 fuzzy systems, the use of general type-2 fuzzy sets requires the specification of an even larger number of parameters for each MF and as such is set to exasperate the problem of choosing those parameters.

We have presented two novel methods which allow the design of general type-2 fuzzy MFs directly from data and as such eliminate the need for a manual design of the MFs or the use of artificial intelligence techniques in order to tune the MFs.

The first method allows the design of general type-2 fuzzy MFs using data on linguistic labels which is collected through a survey. The technique allows for an accurate representation of the perception of linguistic variables gathered through the survey without any loss of information. The second method specifies the general type-2 MFs based on data which is directly sampled from the device (such as a sensor) which is to be modelled and as such creates an accurate representation of the uncertainty associated with the specific device.

Both methods further allow the extraction of type-1, interval type-2 and the recently introduced zSlices based general type-2 fuzzy sets.

Additionally, we have presented a recursive algorithm that computes a convex approximation of fuzzy sets while preserving significantly more detail than simple triangular approximation which in turn enables a more accurate modelling of the uncertainty within the fuzzy MFs.

In the future, we are planning to further refine the algorithms for the automatic definition of fuzzy sets while investigating their performance in real world applications. Particularly, we are looking to extend the theoretical frameworks to non-convex fuzzy sets and to investigate non-singleton fuzzification in conjunction with the automatically created sets.

### Acknowledgment

This work was supported in part by the European Commission under Grant FP7-ICT-1-8.2/216837 entitled "Adaptive and Trusted Ambient Ecologies" (ATRACO).

### References

- [1] Mendel and R. John; "Type-2 Fuzzy Sets Made Simple" ; IEEE Transactions on Fuzzy Systems, Vol. 10, pp.117-127, April 2002.
- [2] H. Hagrass, "A Hierarchical type-2 fuzzy logic control architecture for autonomous mobile robots," *IEEE Transactions On Fuzzy Systems*, vol. 12 No. 4, pp. 524-539, August 2004.
- [3] J. Figueroa, J. Posada, J. Soriano, S. Rojas and M. Melgarejo, "A Type-2 Fuzzy Controller for Tracking Mobile Objects in the Context of Robotic Soccer Games", Proceedings of the International Conference of Fuzzy System, May 2005, pp.359 – 364.
- [4] J. Mendel, *Uncertain Rule-Based Fuzzy Logic Systems*, Prentice Hall 2001.
- [5] A. Homaifar and E. McCormick, "Simultaneous design of membership functions and rule sets for fuzzy controllers using genetic algorithms," *IEEE Transactions Fuzzy Systems*, vol. 3, No. 2, pp. 129-139 May 1995.
- [6] C. Lynch, H. Hagrass and V. Callaghan, "Embedded Interval type-2 Neuro-Fuzzy Speed Controller for Marine Diesel Engines," Proceedings of the 2006 Information Processing and Management of Uncertainty in Knowledge-based Systems conference, Paris, France, July 2006, pp. 1340-1347.
- [7] S. Coupland and R. John, "Geometric type-1 and type-2 fuzzy logic systems," *IEEE Transactions on Fuzzy Systems*, vol. 15, pp. 3-15, Feb. 2007.
- [8] S. Greenfield and R. John, "Optimised Generalised Type-2 Join and Meet Operations," Proceedings of the IEEE International Conference of Fuzzy Systems, London, UK, July 2007, pp. 141-146.
- [9] L. Lucas, T. Centeno and M. Delgado, "General type-2 fuzzy inference systems: analysis, design and computational aspects," Proceedings of the IEEE International Conference of Fuzzy Systems, London, UK, July 2007, pp. 1107-1112.
- [10] J. Mendel, Feilong Liu, "On new quasi-type-2 fuzzy logic systems," Proceedings of the IEEE International Conference of Fuzzy Systems, Hong Kong, China, June 2008, pp. 354-360.
- [11] C. Wagner and H. Hagrass, "zSlices – Towards bridging the gap between Interval and General Type-2 Fuzzy Logic," Proceedings of the IEEE International Conference of Fuzzy Systems, Hong Kong, China, June 2008, pp. 489-497.
- [12] N. Karnik and J. Mendel, "Operations on type-2 fuzzy sets," *Fuzzy Sets and Systems*, vol. 122, pp. 327-348, 2001.