Adaptive Non-singleton Type-2 Fuzzy Logic Systems: A Way Forward for Handling Numerical Uncertainties in Real World Applications

N. Sahab, H. Hagras

Nazanin Sahab, Hani Hagras
University of Essex, United Kingdom
The Computational Intelligence Centre
E-mail: nsahab@essex.ac.uk, hani@essex.ac.uk

Abstract: Real world environments are characterized by high levels of linguistic and numerical uncertainties. A Fuzzy Logic System (FLS) is recognized as an adequate methodology to handle the uncertainties and imprecision available in real world environments and applications. Since the invention of fuzzy logic, it has been applied with great success to numerous real world applications such as washing machines, food processors, battery chargers, electrical vehicles, and several other domestic and industrial appliances. The first generation of FLSs were type-1 FLSs in which type-1 fuzzy sets were employed. Later, it was found that using type-2 FLSs can enable the handling of higher levels of uncertainties. Recent works have shown that interval type-2 FLSs can outperform type-1 FLSs in the applications which encompass high uncertainty levels. However, the majority of interval type-2 FLSs handle the linguistic and input numerical uncertainties using singleton interval type-2 FLSs that mix the numerical and linguistic uncertainties to be handled only by the linguistic labels type-2 fuzzy sets. This ignores the fact that if input numerical uncertainties were present, they should affect the incoming inputs to the FLS. Even in the papers that employed non-singleton type-2 FLSs, the input signals were assumed to have a predefined shape (mostly Gaussian or triangular) which might not reflect the real uncertainty distribution which can vary with the associated measurement. In this paper, we will present a new approach which is based on an adaptive non-singleton interval type-2 FLS where the numerical uncertainties will be modeled and handled by non-singleton type-2 fuzzy inputs and the linguistic uncertainties will be handled by interval type-2 fuzzy sets to represent the antecedents' linguistic labels. The non-singleton type-2 fuzzy inputs are dynamic and they are automatically generated from data and they do not assume a specific shape about the distribution associated with the given sensor. We will present several real world experiments using a real world robot which will show how the proposed type-2 non-singleton type-2 FLS will produce a superior performance to its singleton type-1 and type-2 counterparts when encountering high levels of uncertainties.
Keywords: Type-2 Fuzzy logic, non-singleton Fuzzy Logic Systems, interval type-2 fuzzy logic systems, Type-2 non-singleton type-2 FLS.

1 Introduction

Professor Lotfi Zadeh introduced fuzzy sets in 1965 [1]. He has written several papers since then, but his comprehensive paper on the concept of linguistic variables [2] became very famous as a reference on fuzzy logic theory. He introduced fuzzy logic for the application of approximate
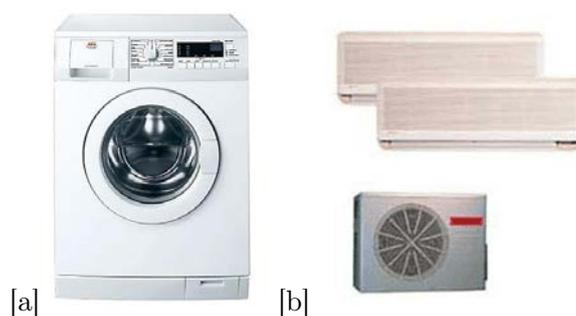
Figure 1: Two examples of domestic products that used fuzzy logic. (a) AEG washing machine [8]. (b) Hitachi air conditioner [9].

reasoning and mentioned the need of a humanistic system whose behavior is strongly influenced by human judgment, perception or emotion for the purpose of computing with words to solve the problems of human-oriented fields such as philosophy, psychology, politics, law, sociology and literature. Professor Zadeh also described the use of his fuzzy logic approach in [3] in describing the behaviors of too complex or too ill-defined systems. This has inspired control engineers from over the world to investigate the power of fuzzy logic to control applications which are difficult to mathematically model.

In 1975, the first Fuzzy Logic Controller (FLC) was developed by Mamdani and Assilian for controlling a steam engine [4]. Since then Fuzzy Logic Systems (FLSs) have been applied with great success to many real world applications. It was reported in 1995 that over the last two decades, the number of applications of FLSs have dramatically increased. Several industrial applications of fuzzy logic were reported from which we can mention cement kilns [5], steel furnaces [5], aircraft engine control, steam turbines, power supply controllers, etc [6]. FLSs have also been used domestically in elevators, washing machines [6], fridges [7], air conditioners, automobiles [5], automatic transmission, camera autofocus control, etc [6].

Figure 1 shows two examples of fuzzy logic usage in domestic appliances which are the AEG washing machine [8] (shown in Figure 1a) and the Hitachi air conditioner [9] (shown in Figure 1b). According to the John Lewis website about a top brand washing machine from AEG and reporting about the fuzzy logic washing machine, the website mentions "fuzzy logic circuit detects when the laundry is out of balance and re-jig it accordingly, ensuring minimum wear and tear to the drum bearings. The fuzzy logic also detects half loads, if too much detergent has been added and adds extra rinses if required" [8]. Hence, for the past thirty years fuzzy logic and its applications became embedded in our everyday environments.

Fuzzy logic laid the basis for a successful method to model uncertainty, vagueness, and imprecision [10]. The FLSs are general knowledge base systems [11] with linguistic rules that can be constructed using the knowledge of experts in a given field. During more than four decades from the invention of fuzzy logic, great progress in using FLS has been achieved. While traditionally type-1 FLSs have been widely employed in real world applications, recent years have shown a rapidly growing interest in the research and application of type-2 FLSs. This is because, it has become apparent that type-1 FLSs cannot fully handle the large amounts of uncertainties encountered in real world environments and applications. Interval type-2 FLSs have shown to provide better performance when compared to type-1 FLSs in applications with high levels of uncertainty. The difference in performance has been attributed to the nature of interval type-2 fuzzy sets which can better account for the uncertainties as they incorporate a Footprint of Uncertainty (FOU) and they can be assumed to embed a large number of type-

1 fuzzy sets. However, the majority of the type-2 FLSs [12] handle the linguistic and input numerical uncertainties using singleton interval type-2 FLSs that mix the numerical and linguistic uncertainties to be handled only by the linguistic labels type-2 fuzzy sets. Input numerical uncertainties are associated with the noise, imprecision and uncertainty associated with sensors and input devices. However, the linguistic uncertainties are associated with human words and perceptions. Hence, it seems paradoxical to use singleton type-2 FLSs to handle the input numerical uncertainties, as this ignores the fact that if input numerical uncertainties were present, they should affect the incoming inputs to the FLS. Thus we cannot treat the incoming FLS inputs as perfect signals as in the case of singleton FLSs. Hence, there is a need to employ non-singleton FLSs to handle the input numerical uncertainties by modeling the inputs as fuzzy inputs rather than crisp values.

Although, there are many papers on type-1 FLSs, quite limited number of papers considered the usage of type-1 non-singleton FLSs such as [13]- [25] and even fewer number considered type-2 non-singleton FLS [25]- [31]. The work done so far in (type-1 and type-2) non-singleton FLSs use predefined shapes for the uncertainty distribution affecting the FLSs inputs (mainly Gaussian and triangular) which might not reflect the real uncertainty distribution associated with the given sensor [32]- [35]. For example in the case of a sonar sensor, the numerical input uncertainties depend on many factors such as the shape and type of objects reflecting the sonar signal as well as wind speed, humidity and temperature. Moreover, the uncertainty distribution also varies with the measured values where for the case of a sonar sensor, the amount of uncertainty affecting measurement readings at 20cm distance could be much less than the uncertainties affecting the measurement at large distances such as 3m. Hence, it is not fair to consider that there is specific shape for the uncertainty distribution (triangular, Gaussian, etc.) with fixed parameters (average, standard deviation, etc.) for all the measured values. [12]

In this paper, we will present some of the theoretical basis for generating an adaptive type-2 fuzzy input which is better able to represent the encountered numerical uncertainty at a given measurement. The non-singleton type-2 fuzzy inputs are dynamic and they are automatically generated from data and they are piece-wise linear type-2 fuzzy sets which are changing for different measurements to model the encountered uncertainty. Piece-wise linear fuzzy sets were used in [36] for the representation of membership functions. However, we use piece-wise linear fuzzy sets to represent the type-2 fuzzy input variables. In this paper, we will present an overview on how the adaptive type-2 input based non-singleton interval type-2 FLS can operate in real time. We will present real world experiments using a mobile robot which will show how under high input uncertainty levels, the non-singleton type-2 FLS can give a good performance and outperform its singleton type-2 and type-1 FLSs counterparts.

Section 2 will review the singleton type-1 FLS. Section 3 will review the singleton interval type-2 FLS while Section 4 will review non-singleton FLSs. Section 5 will provide an overview on modeling a data based type-2 fuzzy input. Section 6 will provide an overview on the proposed approach for type-2 non-singleton type-2 FLS with the type-2 fuzzy input. Section 7 will provide an overview on the system operating in real-time in real world environments. Section 8 provides the experiments and results and Section 9 provides the conclusions and future work.

2   Singleton Type-1 FLS

The first generations of the FLSs were singleton type-1 FLSs. In the singleton type-1 FLS, both the antecedents and the consequents of the FLS are type-1 fuzzy sets and the input values are crisp and precise values. A type-1 fuzzy set, A, for a variable $x \in X$ is characterized by a type-1 membership function $\mu_A(x)$ which is limited between [0,1] [32]:

$$A = \{(x, \mu_A(x)) | \forall x \in X\} \tag{1}$$

In FLSs, membership functions are associated with terms that appear in antecedents or consequents of the FLS rules [32]. Figure 2 shows an example of the linguistic labels, Close, Moderate and Far modeled by type-1 fuzzy sets considered for the antecedent of a type-1 FLS.
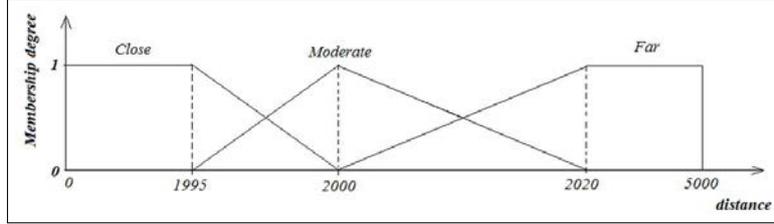
Figure 2: An example of type-1 fuzzy sets which is also used in our experiments as antecedent membership functions.

Figure 3 shows the structure of singleton type-1 FLS (the usual Mamdani type) where the crisp inputs are first fuzzified to input type-1 fuzzy sets which trigger the inference engine and rule base to generate output type-1 fuzzy sets which are then defuzzified to obtain a final crisp output of the FLS. In singleton FLSs, the incoming inputs to the FLS are considered as crisp numbers which assume that the incoming inputs to FLSs are not associated with noise or uncertainty.
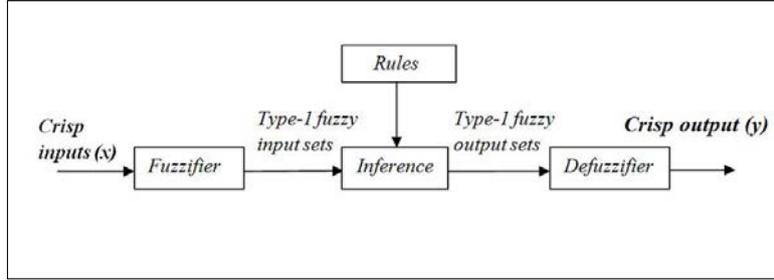
Figure 3: Structure of a singleton type-1 FLS [32].

Considering having p inputs $x_1 \in X_1, ..., x_p \in X_p$ and one output $y \in Y$, where F_k^l is the antecedent membership function for input k and rule l, and G^l is consequent membership function for rule l, the lth rule is shown as:

$$R^l : If\ x_1 is\ F_1^l\ and\ ...\ x_p\ is\ F_p^l,\ then\ y\ is\ G^l. \qquad l = l, ..., M \tag{2}$$

Singleton fuzzifier maps a crisp input $\overrightarrow{x} = (x_1, ..., x_p) \in X_1 \times X_2 \times ... \times X_p \equiv \overrightarrow{X}$ into a fuzzy set A_x in $\overrightarrow{X}$. Singleton fuzzifier is nothing more than a fuzzy singleton where the output of the singleton fuzzifier related to input k and rule l is $\mu_{F_k^l}(x_k\prime)$ when the incoming input is $x_k = x_k\prime$. The firing strength of rule l can be found as follows [34]:

$$\mu^l = \mu_{F_1^l}(x_1\prime) * \mu_{F_2^l}(x_2\prime) * ... * \mu_{F_p^l}(x_p\prime) \tag{3}$$

Where * represented the t-norm operation. The defuzzification produces a crisp output for the FLS from the fuzzy sets that appear at the output of inference engine. There are many different types of defuzzifiers such as centroid, height, and center of sets defuzzifiers [32]. For example for the center sets defuzzifier (which we have used in our experiments) when c^l is the

centroid of the related consequent fuzzy set to rule l and T is t-norm operation, the defuzzified output $y_{cos}(\overrightarrow{x})$ could be written as follows [32]:

$$y_{cos}(\overrightarrow{x}) = \frac{\sum_{l=1}^{M} c^l T_i^p = 1 \mu_{F_i^l}(x_i)}{\sum_{l=1}^{M} T_i^p = 1 \mu_{F_i^l}(x_i)} \tag{4}$$

There are many sources of uncertainty facing the FLS in dynamic unstructured real world environment and many real world applications; some of them are: [37]
- Uncertainties in inputs to FLS due to noise, change of the conditions of observation, etc.;
- Linguistic uncertainties as the meaning of words that are used in antecedent and consequent linguistic labels can have different meaning to different experts [32];
- Uncertainties associated with the use of noisy training data;
- Uncertainties associated with the change in the operation conditions of controllers;
- Uncertainties in control outputs which can be due to the wear and tear associated with the FLS actuators.

All of these uncertainties translate into uncertainties about membership functions. Type-1 FLS cannot handle linguistic and numerical uncertainties associated with dynamic situations, as they use precise membership functions that the user believes capture uncertainties. When the environment change, the chosen type-1 fuzzy sets might not be appropriate any more. This can cause degradation in the performance of the FLS and the designer might end up wasting time to frequently tune type-1 FLS to deal with various uncertainties [37], [32]. Hence, it was claimed that type-1 fuzzy sets might have limited capabilities to directly handle uncertainties related to dynamic uncertain situations. That sounds paradoxical because the word fuzzy has the connotation of uncertainty. The paradox about type-1 fuzzy set has been known for a long time, that a representation of fuzziness is made using membership values which are themselves crisp numbers [38].

A type-2 fuzzy set is characterized by a fuzzy membership function, i.e. the membership value for each element of this set is a fuzzy set in [0,1], unlike a type-1 fuzzy set where the membership value is a crisp number in [0,1] [32]. The membership functions of type-2 fuzzy sets are three dimensional and include a Footprint Of Uncertainty (FOU), it is the new third-dimension of type-2 fuzzy sets and the FOU that provide additional degrees of freedom that can make it possible to directly model and handle the uncertainties [32]. Therefore FLSs that use type-2 fuzzy sets have the potential to handle higher uncertainty levels than their type-1 counterparts [37], [38].

3   Singleton Interval Type-2 FLS

Type-2 fuzzy logic was also introduced by Zadeh in 1975 in [2], but little works [39] were published about them until the middle of previous century. Until then few people were studying on type-2 FLS, as people were learning what to do with type-1 fuzzy sets [39]. Type-2 FLSs are the extension of the type-1 FLS. In type-2 FLSs, the antecedent and/or consequent membership functions of type-2 FLS are type-2 fuzzy sets [40].

A type-2 fuzzy set $\widetilde{A}$ is characterized by a type-2 membership function $0 \leq \mu_{\widetilde{A}}(x, u) \leq 1$, where x ∈ X and u ∈ $J_x \subseteq$ [0,1], as [32]:

$$\widetilde{A} = \{((x, u), \mu_{\widetilde{A}}(x, u)) | \forall x \in X, \forall u \in J_x \subseteq [0, 1]\} \tag{5}$$

J_x is called primary membership of x, where $J_x \subseteq [0, 1]$ for $\forall x \in X$. The uncertainty in the primary membership grades of a type-2 fuzzy membership function consist of a bounded

region, that we call the Footprint of Uncertainty (FOU) of a type-2 membership function. It is the union of all primary membership grades [41]. Figure 4a shows the type-1 membership function which is totally crisp and precise. Figure 4b shows the primary membership function of a type-2 fuzzy set and related FOU. The FOU of a type-2 membership function handles the rich variety of choices that can be made for a type-1 membership function [37], [43]. A type-2 FLS can be thought of as a collection of a large number of type-1 FLS's. When the antecedent and consequent membership grades in the type-2 FLS have a continuous domain, the number of embedded type-1 FLS's is uncountable [40].

Interval type-2 FLS [33] is a special case of general type-2 FLS when the secondary membership functions are interval sets. The FOU can be described in terms of upper and lower membership functions. Figure 4b illustrates an upper membership function and a lower membership function that are two type-1 membership functions that are bounds for the FOU of a type-2 fuzzy set. The upper membership function is denoted by $\overline{\mu}_{\widetilde{A}}(x)$, and the lower membership function is denoted by $\underline{\mu}_{\widetilde{A}}(x)$ [32]. Figure 4c also shows the 3-D view of an interval type-2 fuzzy set.

In a type-2 FLS the rules are defined as Equation (2), however the antecedent and/or consequents are type-2 fuzzy sets. We can mark type-2 fuzzy sets with sign "$\sim$" to be different from type-1 fuzzy sets as [32]. The structure of singleton type-2 FLS is shown in Figure 5 where crisp inputs are first fuzzified into input type-2 fuzzy sets (in singleton fuzzification) which then activate the inference engine and the rule base to produce output type-2 fuzzy sets. These output type-2 fuzzy sets are then processed by the type-reducer which combines the output sets and then performs a centroid calculation, which leads to type-1 fuzzy sets called the type-reduced sets [32]. The defuzzifier can then defuzzify the type-reduced type-1 fuzzy outputs to produce crisp outputs to be fed to the actuators. More information about the singleton interval type-2 FLS can be found in [32], [37], [40].
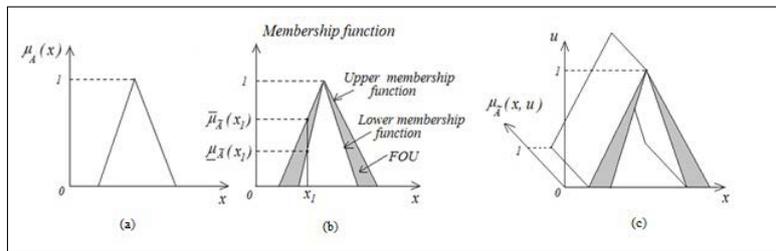
Figure 4:   (a) A type-1 fuzzy set. (b) The primary membership function of a type-2 fuzzy set. (c) A 3-D view of interval type-2 fuzzy set.
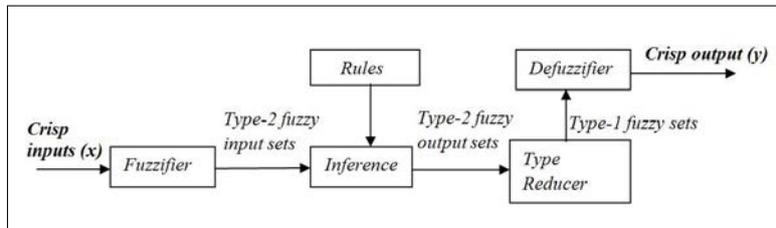
Figure 5:   Structure of singleton interval type-2 FLS [37], [32].

Using type-2 fuzzy sets to represent the inputs and outputs of a FLS has many advantages when compared to the type-1 fuzzy sets, we summarize some of these advantages as follows [37]:
   • The type-2 fuzzy sets of membership functions are themselves fuzzy and include a FOU, so

they can model and handle the linguistic and numerical uncertainty associated with the inputs and outputs of the FLS in changing and dynamic unstructured environments and handle the difficulty associated with determining the exact membership functions for the fuzzy sets [41]. Hence, FLSs based on type-2 fuzzy sets have the potential to produce better performance that type-1 FLS.

• Using type-2 fuzzy sets to represent the FLC inputs and outputs will result in the reduction of the FLC rule base when compared to using type-1 fuzzy sets as the uncertainty represented in the footprint of uncertainty in type-2 fuzzy sets lets us cover the same range as type-1 fuzzy sets with smaller number of labels and the rule reduction will be greater when the number of the FLC inputs increases [32].

• Each input and output will be represented by a large number of type-1 fuzzy sets which are embedded in the type-2 fuzzy sets [42], [41]. The use of such a large number of type-1 fuzzy sets to describe the input and output variables allows for a detailed description of the analytical control surface and gives much smoother control surface and response [37].

During the past decade, type-2 fuzzy sets and systems have become very popular [44]. There are many applications done using type-2 FLSs such as video streaming control [45], induction motor control [46], two-axis motion control [47], hot strip mill temperature control [19], marine diesel engines, DC-DC converter, mobile robots, ambient intelligent environment, etc. [37]. In these applications, the type-2 FLSs have outperformed their type-1 counterparts.

Although, there are many papers on interval type-2 FLSs, the vast majority of these papers used singleton FLSs which mix the numerical and linguistic uncertainties to be handled only by linguistic labels type-2 fuzzy sets. This looks paradoxical, as if numerical uncertainties were present, they should affect the incoming inputs to the FLS, and therefore we cannot treat the incoming FLS inputs as perfect signals in the case of singleton FLSs [48]. Hence, there was a need to consider non-singleton type-2 FLSs to handle the incoming numerical uncertainties.

4 Non-Singleton FLS

The major difference between a non-singleton FLS and a singleton FLS is in the fuzzification part. Zadeh introduced fuzzification in 1973 as "the operation of fuzzification (or, more specifically, support fuzzification) has the effect of transforming a non-fuzzy set into a fuzzy set or increasing the fuzziness of a fuzzy set" [3]. Since 1975 when Mamdani used FLS in a steam engine [4], the majority of the FLSs were using singleton FLS employing singleton fuzzification. The reason for widespread employment of singleton fuzzification is because of its simplicity and speed of computation which allows for real time operation. In singleton fuzzification, inputs are considered to be crisp data and the effect of noise, imprecision, and uncertainty is ignored, while the non-singleton fuzzification models the FLS inputs as fuzzy sets.

FLSs in real world environments [48] can face high levels of uncertainties, which can also be categorized into linguistic and numerical uncertainties. The linguistic uncertainties is related to human words and perceptions, and the input numerical uncertainties are related to noise, imprecision and uncertainty related to input devices and sensors. In non-singleton FLSs, the numerical uncertainties can be modeled and handled by non-singleton fuzzy inputs and the linguistic uncertainties will be handled by antecedent fuzzy sets to represent the linguistic labels. Dealing with numerical uncertainty is especially important in engineering applications in which measurement instruments, sensors, or input devices are employed to produce inputs to FLSs.

There has been some research from instrumentation and measurements researchers in modeling uncertainty measurements using type-1 fuzzy sets and even type-2 fuzzy sets. There are a number of non-ideal situations affecting measurement results such as measurement method, measurement condition, and operator [49]. This has caused the issue of measurement uncertainty

in which the true value is unknown. Ferrero [49] also mentioned that up to that date, the most widely known and assessed theory to deal with incomplete information was probability. But probability just covers random uncertainty which is a part of incomplete knowledge and it is needed to consider both random and unknown systematic effect [49].

Quite limited number of papers (compared with large number of paper in FLSs) used non-singleton type-1 fuzzification. Since 1997 Mouzouris and Mendel presented some closed forms of non-singleton fuzzifier [32], [34]- [35]. Liang and Mendel [32]- [33] in 2000 presented some closed form of non-singleton type-2 FLS for the case of Gaussian type-2 fuzzy inputs with uncertain standard deviation, and some limited number of papers have used type-1 non-singleton type-2 FLS in their applications such as [19]- [25] and more limited number of authors have used type-2 non-singleton type-2 [25]- [31]. However, all the papers assume a predefined shape (such as triangular or Gaussian) as the fuzzy input variable. However, these predefined shapes might not accurately represent the real shape of data distribution related to the input device or sensor. Three categories of non-singleton FLSs have been seen so far in papers which are: Non-singleton type-1 FLS, type-1 non-singleton type-2 FLS, and type-2 non-singleton type-2 FLS. In the first two categories, the input is modeled as a type-1 fuzzy set and in the third category type-2 fuzzy sets are used to model incoming inputs to the FLS.

In this paper, we will present a new approach to obtaining an interval type-2 fuzzy input variable [50] directly from data with no assumption of any predefined shapes for inputs. We will then present our novel adaptive type-2 non-singleton type-2 FLS.

5   Overview of Modeling a Data-Based Type-2 Fuzzy Input

We will start by presenting our data based method to model non-symmetric non-specified shapes of convex type-2 fuzzy inputs, which are directly obtained from data and we do not assume a specific shape about the uncertainty distribution with the given sensor [50].

A summary of the process is as follows:

1. Record 10,000 continuous measurements (at a specific measurement condition) and obtain a histogram for that condition to show how many times a measurement value is recorded.

2. Obtain a normal convex type-1 fuzzy set from the data distribution as follows [50]:

- Obtain the first point from the left side in which the distribution value is in its maximum (shown as max_l in Figure 6).

- Consider the graph from the starting point to the first left maximum (max_l) and connect the points at each distribution value if the value is increasing, otherwise keep the line horizontally at previous value.

- Continue the previous step until getting to the point max_l.

- Then Obtain the first point from right side in which the distribution value is in its maximum (max_r).

- Consider the graph from the right ending point to the first right maximum (max_r) and connect the points at each distribution value if the value is increasing or fixed, otherwise keep the line horizontally at the previous value.

- Continue the previous step until getting to the point max_r.

- To having a convex set, the value from max_l to max_r is kept constant equal to maximum distribution value ($distribution\_no(max_l) = distribution\_no(max_r)$).

- Divide the whole set by the maximum value to change the scale of the function to the acceptable interval [0,1] as a normal fuzzy set.

3. Repeat the first and second steps whilst changing different possible conditions which could affect the input measurements. As for the case of a sonar sensor [50], we considered changing temperature, sound noise, and wind while recording measurements at a fixed distance.
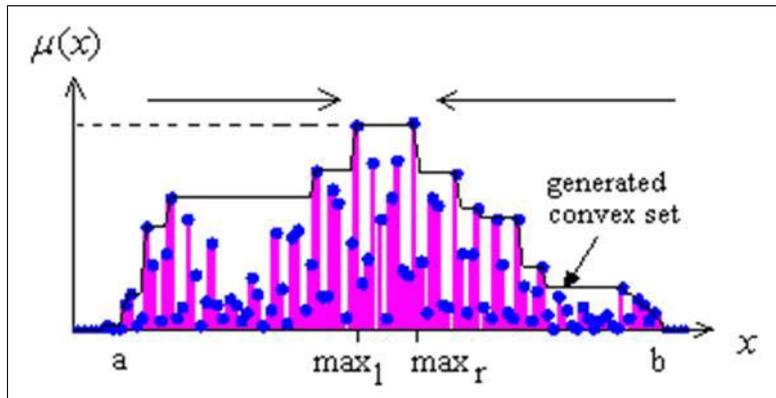
Figure 6: Obtaining a convex function from data distribution [50].

4. Create a convex type-2 fuzzy set from the various type-1 fuzzy input variables, as shown in Figure 7. The method to generate type-2 fuzzy sets from type-1 fuzzy sets is similar to the approach in [51], but we considered non-specified shapes instead of triangular type-1 fuzzy sets. The details of the approach were described in [50]. Another method (which can be done in a discrete form) is to obtain the upper membership function by taking the maximum t-conorm of all the type-1 fuzzy sets, whilst keeping the area from the left point with maximum membership value $\overline{(max_l)}$ to the right point with maximum membership value $\overline{(max_r)}$ constant at one. Lower membership function could be also considered as the minimum of all the type-1 fuzzy sets. [50]
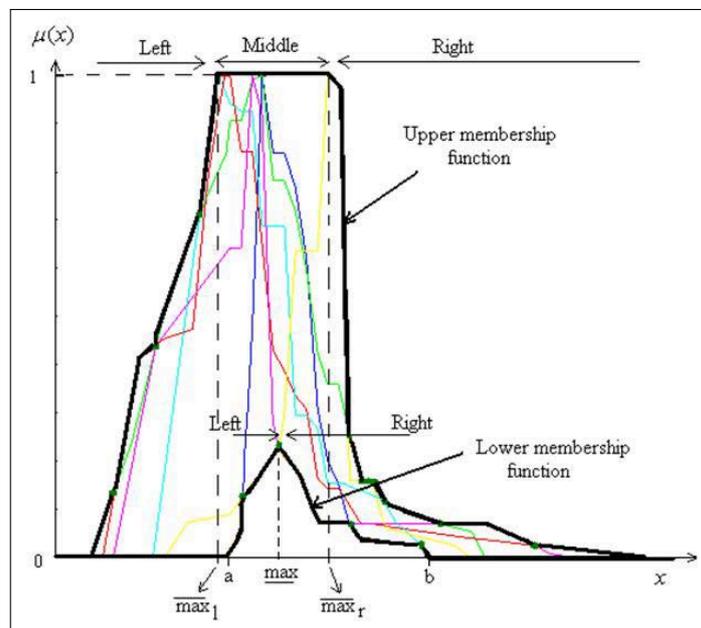
Figure 7: Obtaining type-2 fuzzy input from type-1 fuzzy inputs, the colorful lines show each type-1 fuzzy set and upper membership function and lower membership functions of the resulted interval type-2 fuzzy input is shown by black thick lines [50].

Figure 8 illustrates a type-2 fuzzy input variable modeled from the case of one of the sonar sensors at around 3000mm from a cylindrical object. The black lines define type-2 fuzzy input variable, and the colorful dash lines show the type-1 fuzzy sets from which the type-2 fuzzy input is created.
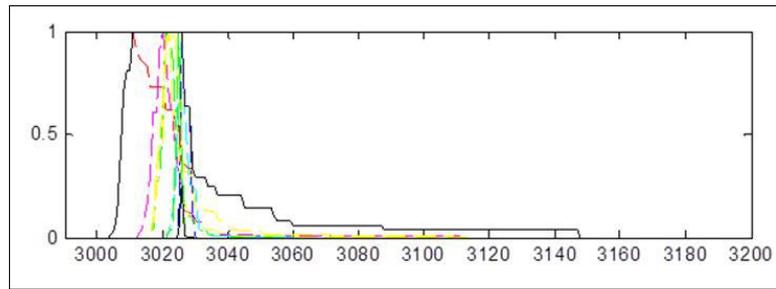
Figure 8: A sample of the modeled type-2 fuzzy inputs at the distance of around 3000mm [12].

In the following section, we present a summarized view on how to employ the type-2 fuzzy input in a type-2 non-singleton type-2 FLS.

6 Type-2 Non-Singleton Type-2 FLS

The type-2 non-singleton type-2 FLS employs the created type-2 fuzzy input variables to handle the encountered numerical uncertainty, and uses type-2 fuzzy sets to model the linguistic labels like "close" , "moderate", and "far" which represent membership functions of antecedents to handle the encountered linguistic uncertainty. Figure 9 illustrates the structure of the type-2 non-singleton type-2 FLS [48]. The following subsections presents the detailed operation of the type-2 non-singleton type-2 FLS.
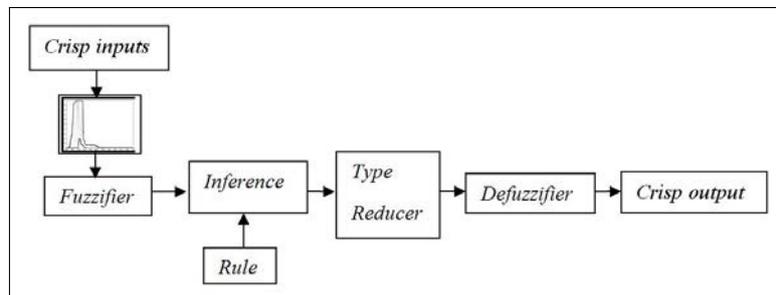
Figure 9: The structure of the type-2 non-singleton type-2 FLS [50].

6.1 Fuzzification

The antecedent type-2 fuzzy sets to represent the linguistic labels were considered as triangular, left shoulder, and right shoulder membership functions (as shown in Figure 10). The input type-2 fuzzy input variables were considered to be a non-specified piece-wise linear shape as explained in Section 5 (as shown in Figure 11). The fuzzifier converts the inputs into linguistic labels with given membership value to activate the rule base and inference engine. The operation between the linguistic labels antecedent fuzzy sets (defined by the interval type-2 fuzzy set) and the fuzzy input variable to provide the lower and upper membership values of input k within rule l is given by Equation (6) and Equation (7) respectively as follows: The antecedent type-2 fuzzy sets to represent the linguistic labels were considered as triangular, left shoulder, and right shoulder membership functions (as shown in Figure 10). The input type-2 fuzzy input variables were considered to be a non-specified piece-wise linear shape as explained in Section 5 (as shown in Figure 11).

The fuzzifier converts the inputs into linguistic labels with given membership value to activate the rule base and inference engine. The operation between the linguistic labels antecedent fuzzy sets (defined by the interval type-2 fuzzy set $\widetilde{F}_k^l$) and the fuzzy input variable to provide the lower and upper membership values of input k within rule l is given by Equation (6) and Equation (7) respectively as follows:

$$\overline{\mu}_k^l(x_k) = \sup_{x_k \in X_k} (\overline{\mu}_{x_k}(x_k) * \overline{\mu}_{F_k^l}(x_k)) \tag{6}$$

$$\underline{\mu}_k^l(x_k) = \sup_{x_k \in X_k} (\underline{\mu}_{x_k}(x_k) * \underline{\mu}_{F_k^l}(x_k)) \tag{7}$$

The $\overline{\mu}_{x_k}(x_k)$ and $\underline{\mu}_{x_k}(x_k)$ are the upper and lower membership functions of the input fuzzy variable, and $\overline{\mu}_{F_k^l}(x_k)$ and $\underline{\mu}_{F_k^l}(x_k)$ are the upper and lower membership functions of the antecedent fuzzy sets representing the linguistic labels respectively. In this work, for the sup operation, we have used the maximum t-conorm, and for the $*$ operation, we have used minimum t-norm. As we have considered three forms of antecedent type-2 fuzzy sets, therefore six algorithms are required to obtain the upper membership value and lower membership values of the output of the fuzzier for the different antecedents.
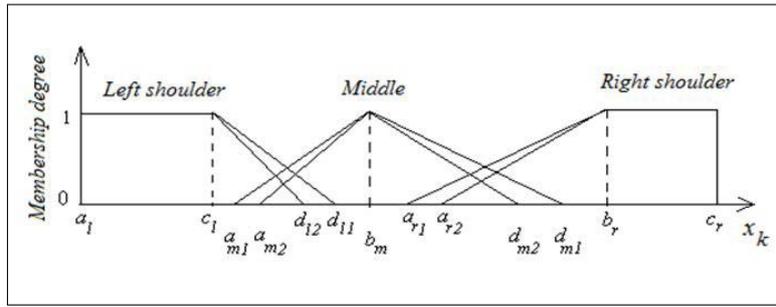
Figure 10: The considered antecedent type-2 fuzzy sets [48].
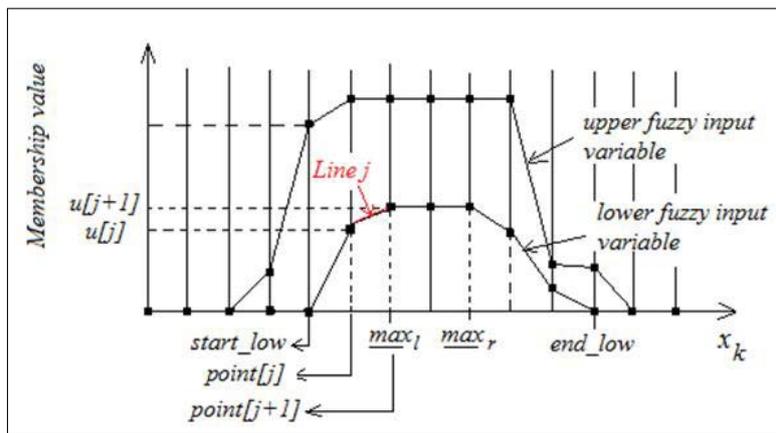
Figure 11: A considered shape of type-2 fuzzy input variable for the non-singleton fuzzification [12].

Due to the limited space, we will present the algorithms to obtain the upper and lower membership values of the type-2 non-singleton fuzzification when only right shoulder antecedent is considered. The rest of the algorithms follow the same logic.

The first example of fuzzification is on finding upper membership value of the fuzzifier, when antecedent membership function is a right shoulder (shown in Figure 12 with green and black colors). Following Equation (6), we need to perform the max-min operation between the upper membership functions of the type-2 fuzzy input variable and the right shoulder antecedent type-2 fuzzy set. The minimum operation between the type-1 membership functions and the type-1 fuzzy input variable will resulted in a type-1 fuzzy set which is shown in Figure 12 by red line, where we need to find its sup (maximum) to find $\overline{\mu}_k^l(x_k)$.

When the ending point of the upper membership function of the fuzzy input variable is larger than a_{r1}, it is required to consider the operations involving the right shoulder antecedent type-2 fuzzy set, as it is not zero. Two cases can happen, where case 1 is shown in Figure 12a and case 2 is shown in Figure 12b and Figure 12c. Case1 happens when $\overline{max}_r$ (shown in Figure 12) is smaller than b_r. In that case, the line of antecedent from a_{r1} to b_r would be considered.
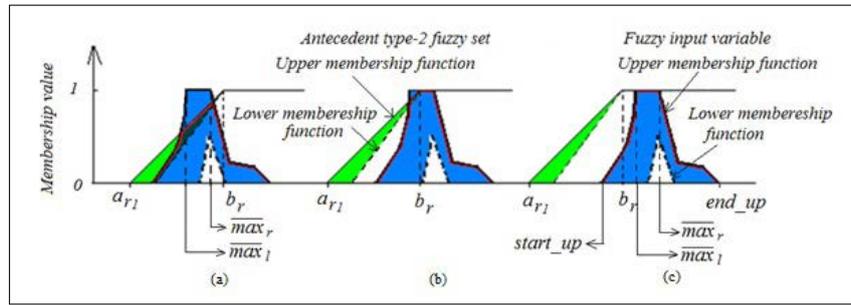
Figure 12: Different cases for obtaining the upper membership value of the fuzzifier output when the right shoulder antecedent membership function is considered in rule (a) case1. (b) and (c) case 2.

The algorithm in Figure 13 shows how to obtain the $\overline{\mu}_k^l(x_k)$ where we find the intersection of the sets of lines j (which form the piece-wise linear type-2 fuzzy input variable) and the antecedent type-2 fuzzy sets.

The given line j (of fuzzy input) ranges from *point[j]* to *point[j+1]* in the x-axis line where j+1 is the line connecting to the line j and located at its right in the piecewise linear function. The *u[j]* and *u[j+1]* are also membership value at *point[j]* and *point[j+1]* respectively. The membership value for any point x_k located on line j is obtained from (8) as follows:

$$\frac{\mu_{x_k}(x_k) - u[j]}{x_k - point[j]} = \frac{u[j+1] - u[j]}{point[j+1] - point[j]}$$

$$\mu_{x_k}(x_k) = \frac{u[j+1] - u[j]}{point[j+1] - point[j]} x_k + \frac{point[j+1]u[j] - point[j]u[j+1]}{point[j+1] - point[j]} \tag{8}$$

The membership value $\mu_{F_k^l}(x_k)$ of any point x_k on a line of the antecedent fuzzy set (starting from an extreme left point "left" (a_{r1} in our case) and ending at an extreme right point "right" (b_r in our case)) is obtained as follows:

$$\frac{\mu_{F_k^l}(x_k) - \mu_{F_k^l}(left)}{x_k - left} = \frac{\mu_{F_k^l}(right) - \mu_{F_k^l}(left)}{right - left}$$

$$\mu_{F_k^l}(x_k) = \frac{\mu_{F_k^l}(right) - \mu_{F_k^l}(left)}{right - left} x_k + \frac{right\mu_{F_k^l}(left) - left\mu_{F_k^l}(right)}{right - left} \tag{9}$$

At the intersection point $x_{k,c}$ between an antecedent line and a line j of the fuzzy input variable, $\mu_{x_k}(x_k)$ in Equation (8) is equal to $\mu_{F_k^l}(x_k)$ in Equation (9). Hence $x_{k,c}$ could be found as follows:

$$x_{k,c} = \frac{\left[\begin{array}{c}(right\mu_{F_k^l}(left) - left\mu_{F_k^l}(right))(point[j+1] - point[j])+ \\ (right - left)(point[j]u[j+1] - point[j+1]u[j])\end{array}\right]}{\left[\begin{array}{c}(right - left)(u[j+1] - u[j])+ \\ (\mu_{F_k^l}(left) - \mu_{F_k^l}(right))(point[j+1] - point[j])\end{array}\right]} \tag{10}$$

Subsistuting with the value of $x_k = x_{k,c}$ in Equation (8) or (9), we can find the intersection points when it is needed as $\overline{\mu}_k^l(x_k) = \mu_k^l(x_k)$.

Case 2 happens when $\overline{max}_r$ is larger or equal b_r, which is the case shown in Figure 12b and Figure 12c. Hence the output of fuzzifier for the upper membership value is one. The algorithm given in Figure 13 summarizes the method to obtain $\overline{\mu}_k^l(x_k)$ in the case that the antecedent type-2 fuzzy set is a right shoulder.

Step1: Initially consider $\overline{\mu}_k^l(x_k) = 0$ (which means no input is fired when right shoulder antecedent and input k is considered.)

Step2: Check if $end\_up > a_{r1}$. If that was true, continue; otherwise exit the algorithm.

Step3: (case1) Check if $\overline{max}_r < b_r$. If that was true continue; otherwise go to case2.

Step4: Consider the line connecting a_{r1} and b_r of the upper membership function of the antecedent type-2 fuzzy set and set the parameters of antecedent line as ($left = a_{r1}$ and $right = b_r$, and hence $\mu_{F_k^l}(left) = 0$ and $\mu_{F_k^l}(right) = 1$).

Step5: Consider the first line j (which ends at point[j+1]=end_up) of the upper membership function of the type-2 fuzzy input variable.

Step6: Check if there were no more lines j exists ($point[j] < \overline{max}_r$). If that was true, exit the algorithm; otherwise continue.

Step7: Obtain the point of intersection from Equation (10).

Step8: Check if the obtained intersection point is acceptable ($point[j] \leq x_{k,c} \leq point[j+1]$). If that was accepted, continue; otherwise consider the next line (decrease j by one) and come back to step6.

Step9: Obtain $\overline{\mu}_k^l(x_k) = \mu_{xk}(x_{k,c})$ for the point $x_{k,c}$ from equation (8). Then exit the algorithm.

Step10: (case2) Check if $\overline{max}_r \geq b_r$, if that was true continue; otherwise exit the algorithm.

Step11: Consider $\overline{\mu}_k^l(x_k) = 1$. Then exit from the algorithm.

Figure 13: The algorithm to obtain the upper membership value of the fuzzifier for input k and rule l (when right shoulder antecedent is considered in the rule).

The second example of fuzzification is on finding lower membership value of the fuzzier, when antecedent membership function is a right shoulder (shown in Figure 14 with green and black colors). Following the Equation (7), we need to obtain the t-norm (minimum) between the type-1 membership function $\underline{\mu}_{x_k}(x_k)$ and $\underline{\mu}_{F_k^l}(x_k)$. This will result in a type-1 fuzzy set (shown by red line in Figure 14), where we need to find its sup (maximum) to find related $\underline{\mu}_k^l(x_k)$.

If the ending point of lower membership function of the type-2 fuzzy input is smaller or equal a_{r2}(the start of the lower membership function of right shoulder antecedent), the lower membership value of the fuzzifier for the right shoulder is zero. If it is not zero, then two cases can happen, where case 1 is shown in Figure 14a and case 2 is shown in Figure 14b. Case 1 happens when the lower membership value of fuzzy input at point $\underline{max}_r$(shown in Figure 14a) is larger than the membership value of the antecedent point $\underline{max}_r$, which can be considered as:
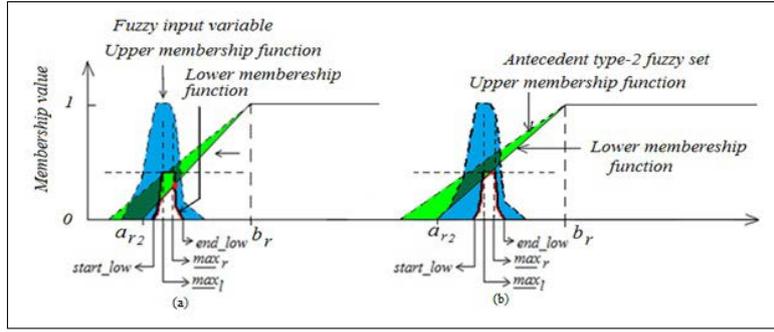
Figure 14: Different cases for obtaining the lower membership value of the fuzzifier output when the right shoulder antecedent membership function is considered in rule (a) case1 (b) case 2.

$$\underline{\max}_r < \frac{right(\mu_{x_k}(\underline{\max}_r) - \mu_{F_k^l}(left)) + left(\mu_{F_k^l}(right) - \mu_{x_k}(\underline{\max}_r))}{\mu_{F_k^l}(right) - \mu_{F_k^l}(left)} \tag{11}$$

In this case, the max-min will be located on the antecedent line from a_{r2} to b_r. Because of the positive slope of the antecedent line, the most right intersection between the lower membership function of the type-2 fuzzy input and the antecedent line has the highest membership value, and its membership value is $\underline{\mu}_k^l(x_k)$, when the right shoulder is considered in rule.

Case 2 happens when the lower membership value of fuzzy input at point $\underline{max}_r$ is not larger that the membership value of the antecedent at point $\underline{max}_r$ (when case1 does not happen). In this case, $\underline{\mu}_k^l(x_k)$ is equal to the maximum membership value of the lower membership function of the type-2 fuzzy input. Figure 15 illustrates the algorithm to obtain the lower membership value of the fuzzifier for input k and rule l when right shoulder antecedent is considered in rule.

Figure 15: The algorithm to obtain the lower membership value of the fuzzifier for input k and rule l (when right shoulder antecedent is considered in the rule).

6.2   Rule Base and Inference Engine

The outputs of fuzzifier trigger the inference engine and the rule base to generate type-2 fuzzy outputs which are then type-reduced and defuzzified to generate the final crisp output of the FLS. The same approach as singleton FLS is applicable to the type-2 non-singleton type-2 FLS after obtaining the output of fuzzifier. The rule base, inference engine, type-reduction and defuzzification follow as:

• Rule Base: The rule base of singleton and non-singleton FLS are similar. Considering a FLS having inputs $x_1 \in X_1, ..., x_p \in X_p$ and one output $y \in Y$, the rule l is defined as [32]:

$$If\ x_1\ is\ \widetilde{F}_1^l\ ...\ and\ x_p\ is \widetilde{F}_p^l,\ then\ y\ is\ \widetilde{G}^l \tag{12}$$

Where l=1,..., M and M is the total number of rules in the rule base. The $\widetilde{F}_k^l$ is the antecedent membership function for input k and rule l, and $\widetilde{G}^l$ is consequent membership function for rule l.

• Inference Engine: We have shown in the fuzzifier subsection, how to find the upper and lower membership values of a given type-2 fuzzy input variable to a given linguistic label. The process needs to be done for all inputs (k=1,..., p). At the end, the firing strength of each rule $F^l = [\underline{f}^l, \overline{f}^l]$ could be found where $\underline{f}^l$ and $\overline{f}^l$ will be obtained as follows:

$$\underline{f}^l = T_{k=1}^p \{\underline{\mu}_k^l(x_k)\} \tag{13}$$

$$\overline{f}^l = T_{k=1}^p \{\overline{\mu}_k^l(x_k)\} \tag{14}$$

6.3   Type-Reduction

After obtaining the lower and upper firing strengths for each rule from the inference engine and obtaining the centroid of consequent membership function, the center of sets type-reducer [32] is used to produce the type-reduced sets.

6.4   Defuzzification

The final defuzzified crisp output is obtained by taking the average of the type-reduced set.

7   Adaptive Type-2 Input Based Non-Singleton Type-2 FLS

We considered the non-singleton type-2 fuzzy inputs to be dynamic and automatically generate from data and the shapes are changing for different measurement values. As we cannot generate the type-2 fuzzy sets for all the measurement values, we need to interpolate the type-2 fuzzy set for an incoming measurement value by interpolating the type-2 fuzzy sets representing the measurements to the left and right of incoming measurement value. Figure 16 shows the process of the on-line singleton type-2 FLS. [12]. We obtained some sample of original type-2 fuzzy input variables for different measurement values within considered range according to the instructions given in Section 5, and then obtained the rest of the type-2 fuzzy input variables by a linear interpolation defined for type-2 fuzzy sets. The interpolated membership value for the desired crisp incoming input (d) can be obtained as follows:

$$u_d[i] = \frac{(d - d_2)}{(d_1 - d_2)} u_1[i] + \frac{(d - d_1)}{(d_2 - d_1)} u_2[i] \tag{15}$$

The $u_1[i]$ is the membership value of the left original fuzzy input variable, and $u_2[i]$ is the right original fuzzy input variable. The d_1 is the given measurement input for the left fuzzy input variable and d_2 is the given measurement input for the right neighboring fuzzy input variables, and d is the crisp value (new measurement data) in question which is to be modeled by an interpolated type-2 fuzzy input variable.
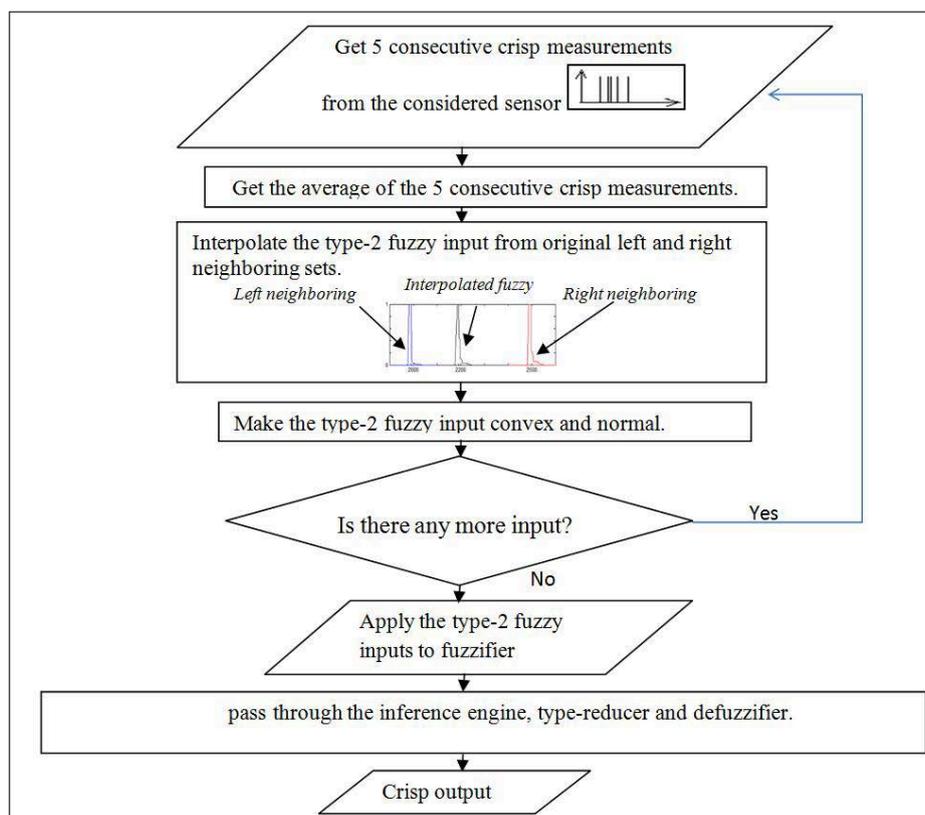
Figure 16: The process of on-line non-singleton type-2 FLS [12].

The interpolation is considered to be done separately for the lower membership functions and the upper membership functions for the fuzzy input. The resulting interval type-2 fuzzy sets need to be convex and normal and if this is not the case, we need to convert the type-2 fuzzy set to a normal convex fuzzy set [48].

8 Experiments and Results

In this section, we will present real world experiments in a real world environment and we will describe the characteristics of the considered FLSs and test environment. We will then present control surface analysis and the real world experiments results.

8.1 Experimental Setup

In this Section, we will present real world experiments which were carried using a real world robot navigating in a real world environment. We will compare the performance of the adaptive type-2 non-singleton type-2 FLS against its singleton type-1 and type-2 FLSs counterparts. In our experiments, we have employed a four-wheel mobile robot (shown in Figure 17) to do a left edge following behavior using two sonar sensors. A hair dryer was located on robot to blow hot

air through the path of front sonar sensor to create larger measurement uncertainty while robot is moving.
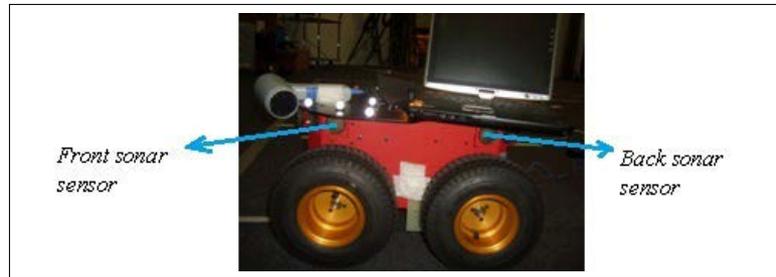
Figure 17: The four wheel mobile robot used in the experiments.

In order to test the robot under unseen input uncertainties [12], we have applied various sources of uncertainties to the input sonar sensors (as shown in Figure 18a, Figure 18b) where we have used a hair dryer stuck on robot to blow hot air while a fan fixed at a certain position on ground to create cold wind. These employed uncertainties are introduced during the online operation of the robot and they have not been seen by the robot during the FLS design. We chose changing the wind speed (through the fan and dryer) and temperature (through the hair dryer) to introduce input uncertainties to the sonar sensors. This is because the wind, temperature variations and atmospheric turbulences are all meteorological conditions that have an effect on sound propagation (as we used sonar sensors) [52]. The speed of sound in air is proportional to the square root of absolute temperature as follows:

$$c = 20.05\sqrt{T + e/p} \qquad (16)$$

Where T is absolute temperature in Kelvin, and e is partial pressure of water vapor, and p is barometric pressure. Wind is also another factor affecting the sound travel. The wind in the same direction of sound will cause the wave propagation to be bent downward, and in the upwind direction sound waves move upward which cause no direct sound penetration [52].

To obtain the position of the robot in a precise way, the VICON system is used in which six cameras are employed. To be able to recognize the robot, four markers are used (as shown in Figure18). The top of the cylinders are also marked to be able to check if the edges of the cylinders are in a same line where the starting and ending cylinders are modeled in VICON to be able to distinguish the certain position of wall (one of them is shown in Figure18c) [12].

In our experiments, we considered a type-1 FLS which was able to give a good performance under the normal lab conditions. The performance of type-1 FLS was considered to be acceptable in a usual room temperature without applying external uncertainty. Two main factors were considered to evaluate the performance of the FLSs, the first one is the Root Mean Square Error (RMSE) which is considered as a measurement factor for the average of error for the distance between the robot's path from the desired path. The second factor is standard deviation (STD) of error in following edge to check the variance of the error around the average error value.

The shapes of the type-2 fuzzy sets used to represent the linguistic labels are shown in Figure 19 where we blurred the type-1 membership functions (shown in thin lines in Figure 19) to obtain the FOU of the type-2 fuzzy sets (shown in Figure 19 by thick line) for the interval (singleton and non-singleton) type-2 FLS. The consequent membership functions (for all FLSs) are type-1 fuzzy sets (shown in Figure 20) as there are no major uncertainties associated with robot outputs. The rule-base (employed for all the FLSs) is shown in Table 1.

<center>(a)   (b)   (c)</center>
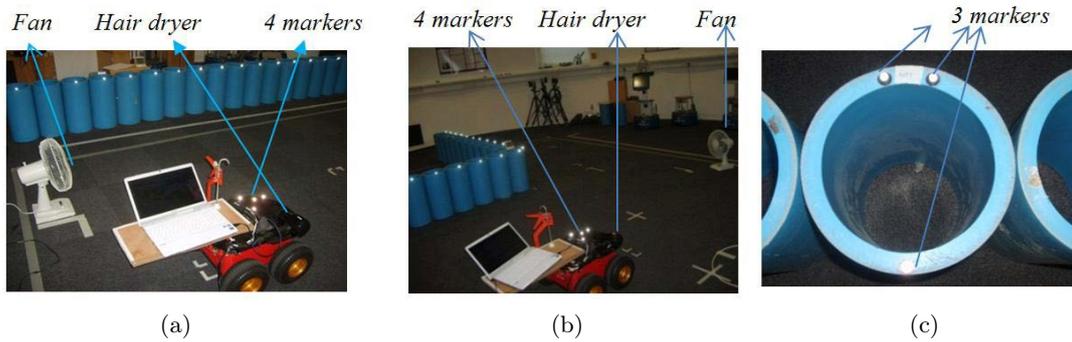
Figure 18: Mobile robot experiments when (a) Left edge following set in a straight line. (b) Left edge following set in straight line with a right angle in the way. (c) One of the cylinders modeled in Vicon system using three markers to know the exact position of wall.
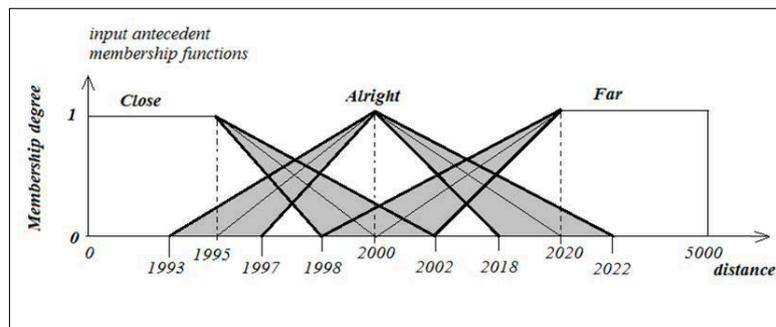
Figure 19: Antecedent membership functions for both the front and back senors. The type-1 fuzzy sets are shown in thin line while the interval type-2 fuzzy sets are shown in thick lines [12].
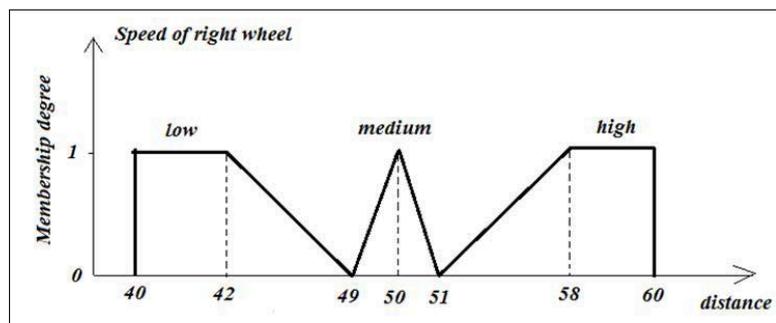
Figure 20: The consequent type-1 fuzzy sets for all the used FLSs [12].

Front sonar	Back sonar		
	Close	Alright	Far
Close	low	low	low
Alright	low	medium	low
Far	high	high	high

Table 1: The rule base for all the FLSs; the output in the rule base is the speed of right wheel [12].

8.2   Control Surface Analysis

The control surface graphically represents the unknown function articulated by the FLS. Figure 21 shows the control surfaces for the singleton type-1 and type-2 FLSs and the type-2 non-singleton type-2 FLS. The control surfaces show the inputs of front and back sensors plotted against the speed of right wheel as output of the FLSs. Note that a smooth shape of a control surface translates to a smooth control response that can deal with uncertainty and imprecision.
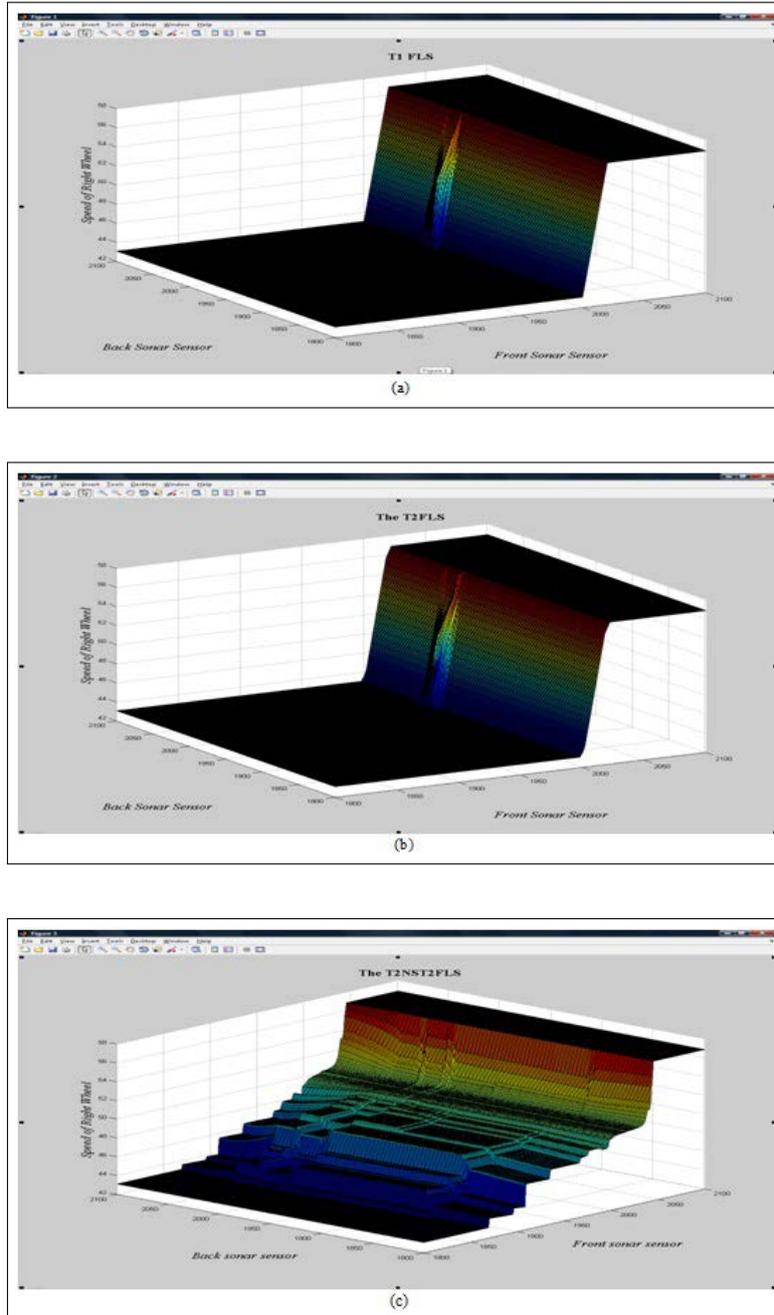
Figure 21: The control surface of (a) The singleton type-1 FLS. (b) The singleton interval type-2 FLS. (c) The type-2 non-singleton intreval type-2 FLS.

As seen from Figure 21a, the control surface of the type-1 FLS is rather steep which means it can be affected by high levels of uncertainties in the inputs where small change in the inputs

can result in large output changes which might cause high overshoots/undershoots and steady state errors in addition to moving towards an unstable situation. On the other hand the control surface of the singleton type-2 FLS (shown in Figure 21b) is smoother than the type-1 FLS, which means it can handle higher level of uncertainties. However, the control surface of the type-2 non-singleton type-2 FLS (shown in Figure 21c) is much smoother where there are smooth transitions at all parts of the control surface where any small change in inputs will result in a small change in the outputs. Thus using the type-2 input based non-singleton type-2 FLS result in a control surface which is smoother than the singleton type-1 and type-2 FLSs. This smooth response will consequently give a very good control performance that can handle the high levels of uncertainties and disturbances where small variations in inputs will not cause significant changes to outputs. Thus, it is expected that the type-2 non-singleton type-2 FLS will be able to handle higher levels of uncertainties than its singleton type-1 and type-2 FLSs counterparts. This will be evidenced in the real world experiments reported in the following subsection.

8.3   Real World Experiments

Figure 22a and Figure 22b show the results of the robotic experiments illustrated in Figure 18a, which compare the performance of singleton type-1, singleton type-2 and type-2 input based non-singleton type-2 FLS to realize an edge following behavior at a desired distance of 2m. The robot starts from the distance of 1.8m from the nearest edge. Figure 22a shows the robot's paths of singleton type-1, type-2 and the non-singleton type-2 FLS under no induced uncertainty. As shown in Figure 22a and Table 2 as expected under no uncertainty, the performance of all the FLSs will be quite similar where the type-1 FLS will be slightly better. However, as high levels of uncertainties are introduced (by blowing hot wind through the way of one sonar) as shown in Figure 22b and Table 2, the type-2 non-singleton type-2 FLS gives the best performance in terms of the RMSE and the STD (which could represent the size of the overshoots/undershoots from the average). In terms of control performance, the type-2 non-singleton type-2 FLS provides the best performance followed (by a large margin) by the singleton type-2 FLS and then the singleton type-1 FLS which performs poorly.

Figure 22b evidences the control surface analysis in Figure 21 where the type-2 nonsingleton type-2 FLS recovers quickly from the uncertainties with relatively smaller overshoot/undershoot.

In the other experiments we are presenting in the paper, we consider the same type-1 fuzzy antecedent while we increase the FOU as illustrated in Figure 23 to have larger difference between type-1 FLS and type-2 FLS. The rule base and consequent membership functions were considered the same as Table 1 and Figure 20 respectively. The other differences are introducing a right angle in the middle of the edges set in straight line and changing the position of fan (shown in Figure 18b).

Different Conditions	Different Types of FLS		
	Type-1 FLS	Type-2 FLS	Type-2 nonsingleton Type-2
No uncertainty	STD = 45.5617	STD =46.5435	STD= 47.9173
	RMSE =45.5695	RMSE=46.5823	RMSE= 51.2167
large uncertainty	(Stopped) STD = 624.8140	STD =477.2452	STD =244.2411
	RMSE=1268.2	RMSE=1037.8	RMSE=634.4

Table 2: Results of following a straight wall for the singleton type-1 FLS, singleton type-2 FLS, and adaptive type-2 input based non-singleton type-2 FLS under different conditions (related to Figure 22).
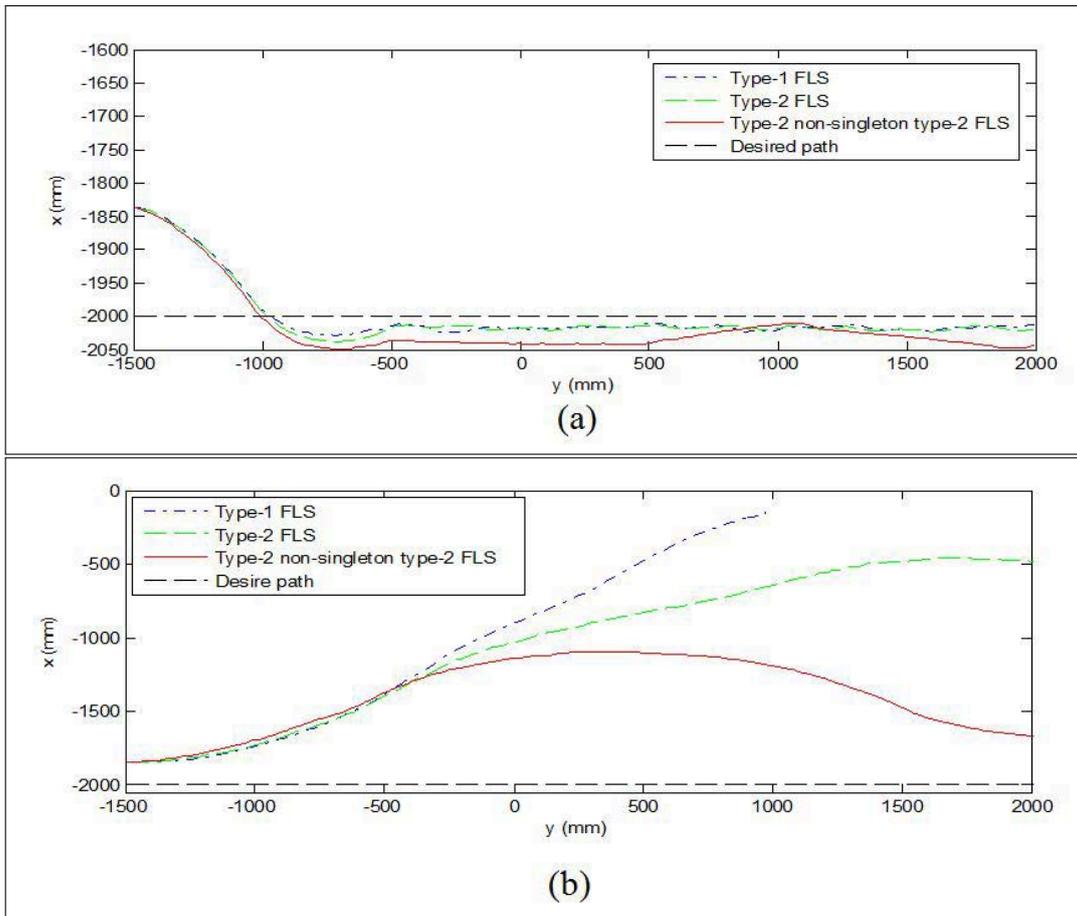
Figure 22: The robot's path when employing singleton type-1 FLS, singleton type-2 FLS, and the adaptive type-2 input based type-2 FLS, while following a straight wall under different conditions (a) No external uncertainty. (b) Large external uncertainty.
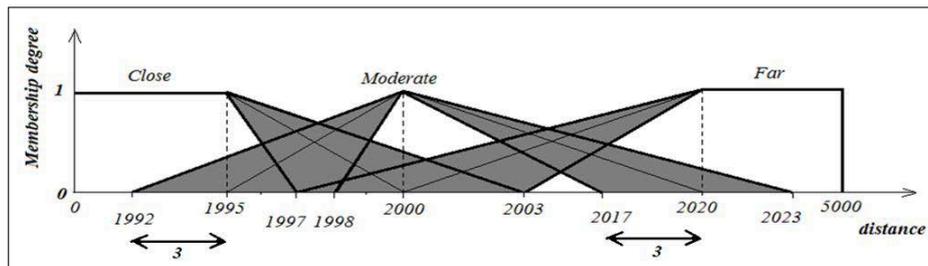
Figure 23: The type-2 fuzzy sets for the antecedent membership functions considered in the experiments whose results are shown in Figure 24 and Figure 25, which is actually considering uncertainty bound of 3 around type-1 fuzzy sets.

Figure 24 shows the robot's paths when employing the type-2 non-singleton type-2 FLS with its singleton type-1 FLS and singleton type-2 FLS counterparts to do left edge following at a desired distance of 2m. The robot was started from the distance of 1.8m from the nearest edge.

Figure 24a shows that the robot's paths under no induced uncertainty for the three FLS are quite similar. Looking at Table 3, it can be seen that the performance of the type-2 input based non-singleton type-2 FLS is slightly better than the other FLSs because it has the least

STD and RMSE compared with its singleton type-1 FLS and singleton type-2 FLS counterparts. When considering moderate uncertainty (the related robot's paths are shown in Figure 24b), the difference between the performance of the adaptive type-2 non-singleton type-2 FLS and the singleton type-1 FLS and singleton type-2 FLS are becoming more distinguishable. The robot's path for the non-singleton type-2 FLS is closer to the desired path specified by black dashed lines, so it has the best performance while the type-2 FLS has also outperformed the type-1 FLS. Table 3 also shows the least STD and RMSE for the non-singleton type-2 FLS, then the singleton type-2 FLS which outperform the type-1 FLS by having less STD and RMSE. When considering large uncertainty (the related robot's paths are shown in Figure 24c), the non-singleton type-2 FLS outperforms its singleton type-2 and singleton type-1 FLS counterparts, while the type-1 FLS is performing very poorly. Table 3 also shows the least STD and RMSE for the non-singleton type-2 FLS.

Figure 25 shows the robot's paths when employing the type-2 non-singleton type-2 FLS and its singleton type-1 FLS and singleton type-2 FLS counterparts to do left edge following at a desired distance of 2m. The robot was started from the distance of 1.9m from the nearest edge. Figure 25a shows the robot's path when employing the FLSs under no induced uncertainty. The performances of the FLSs are quite similar as it can be seen from the Figure 25a while the adaptive type-2 non-singleton type-2 FLS was slightly better as it has the least STD and RMSE (illustrated in Table 4). Figure 25b and Figure 25c also shows that the type-2 input based non-singleton type-2 FLS outperforms its singleton type-2 and type-1 FLS counterparts when moderate and large uncertainty are applied respectively. Table 4 also confirms that the type-2 input based non-singleton type-2 FLS outperforms its counterparts FLSs in the experiments as the type-2 input based non-singleton type-2 FLS has the least STD and RMSE, while the type-1 FLS performs poorly.

Different Conditions	Different Types of FLS		
	Type-1 FLS	Type-2 FLS	Type-2 nonsingleton Type-2
No uncertainty	STD =202.5274 RMSE =202.7028	STD =203.8221 RMSE=204.0164	STD =191.3634 RMSE=191.5175
Moderate uncertainty	STD =907.7865 RMSE =910.7622	STD =700.8891 RMSE=703.5157	STD =533.0486 RMSE=546.5797
Large uncertainty	(Stopped)	STD =694.8356 RMSE=699.5279	STD =669.3059 RMSE=676.0831

Table 3: Summary results of the edge following behviour for the singleton type-1 FLS, singleton type-2 FLS, and adaptive type-2 input based non-singleton type-2 FLS under the different three conditions related to Figure 24.

Different Conditions	Different Types of FLS		
	Type-1 FLS	Type-2 FLS	Type-2 nonsingleton Type-2
No uncertainty	STD = 201.7354 RMSE = 201.7395	STD = 201.7226 RMSE=201.7245	STD = 193.5605 RMSE=193.5647
Moderate uncertainty	STD =562.0358 RMSE=588.3824	STD=548.4835 RMSE=584.5484	STD= 345.9873 RMSE=352.3825
Large uncertainty	(Stopped)	(Stopped)	STD =747.6389 RMSE=748.5229

Table 4: Summary results of the edge following behavior for the singleton type-1 FLS, singleton type-2 FLS, and adaptive type-2 input based non-singleton type-2 FLS under the different three conditions related to Figure 25.
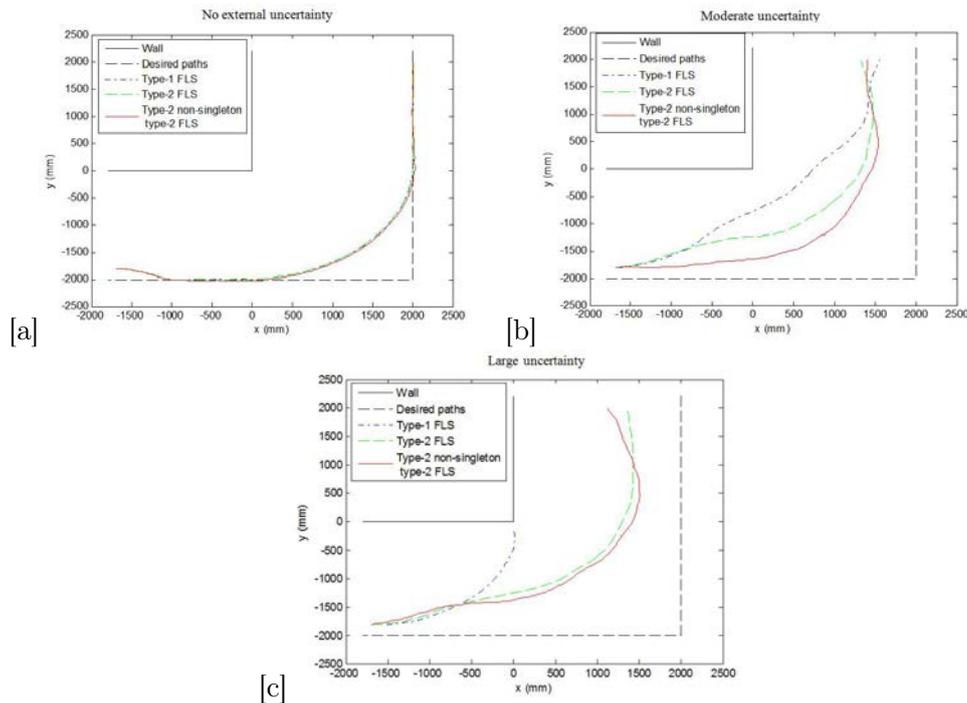
Figure 24: Comparison between singleton type-1 FLS, singleton type-2 FLS, and the adaptive type-2 input based non-singleton type-2 FLS when the robot started from 1.8 meter distance from the edge of wall while applying (a) No external uncertainty (b) Moderate uncertainty (c) Large uncertainty.

9   Conclusions and Future Directions

In this paper, we have presented an adaptive type-2 fuzzy input based non-singleton type-2 FLS which is better able to handle the encountered input uncertainties. The non-singleton type-2 fuzzy inputs are dynamic and they are automatically generated from data and they do not assume a specific shape about the uncertainty distribution associated with the given sensor. We have presented an overview on how the adaptive type-2 input based non-singleton interval type-2 FLS can operate in real time. We have shown through control surface analysis, how the non-singleton type-2 FLS can produce much smoother control surface than the singleton type-1 and type-2 FLSs. This should result in a better control response that handles high uncertainty levels. This was evidenced through the real world experiments using mobile robots where under high uncertainty levels, the non-singleton type-2 FLS can outperform its singleton type-2 and type-1 FLSs counterparts. This work has shown the benefits achieved when using type-2 non-singleton type-2 FLSs in applications characterized by high input uncertainty levels.

In our future work, we are going to tune the parameters of the type-2 FLS by neural networks. We aim also to work towards modeling more generalized type-2 fuzzy input variable or even doing non-singleton fuzzifier in general type-2 FLSs. Last but not least, we are also considering developing a self-tuning adaptive type-2 input based non-singleton interval type-2 FLS, which is updating the shape of the fuzzy input variable whenever the performance of the FLS was deteriorated.
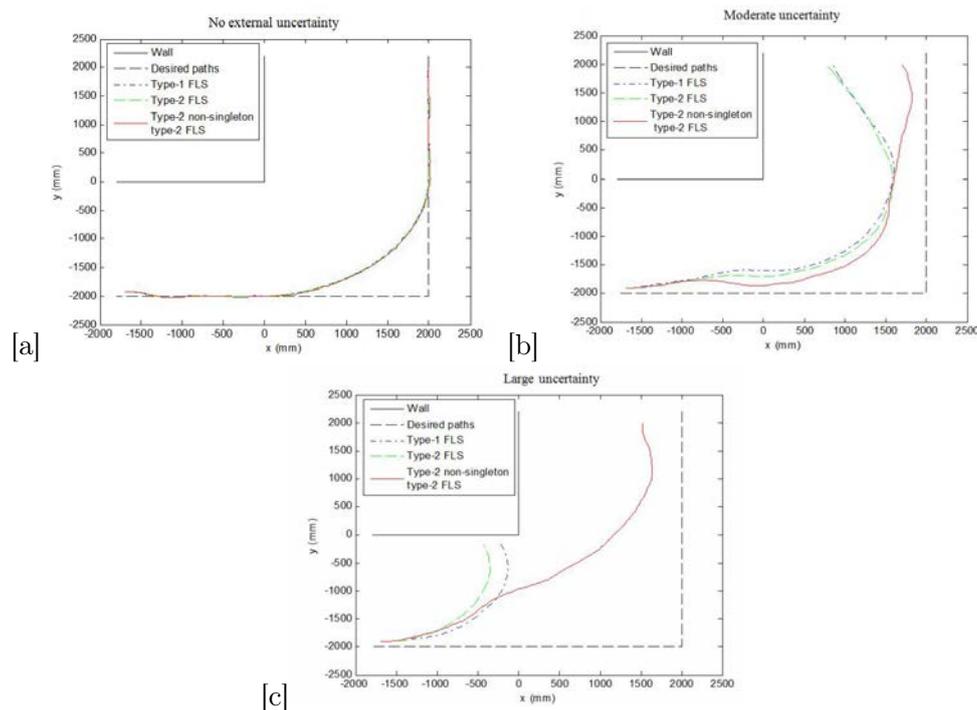
Figure 25: Comparison between singleton type-1 FLS, singleton type-2 FLS, and the adaptive type-2 input based non-singleton type-2 FLS when the robot started from 1.9 meter distance from the edge of wall while applying (a) No external Uncertainty. (b) Moderate uncertainty. (c) Large uncertainty.

Bibliography

[1] L. A. Zadeh, Fuzzy Sets, Information and Control, 8, 338-353, 1965.

[2] L. A. Zadeh, The Concept of a Linguistic Variable and its Application to Approximate Reasoning-I, Information Science 8, 199-249, 1975.

[3] L. A. Zadeh, Outline of a New Approach to the Analysis of Complex Systems and Decision Processes, IEEE Transactions on Systems, Man, and Cybernetics, pp.28-44, 1973.

[4] E. H. Mamdani, S. Assilian, An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller, International Journal of Man-Machine Studies (1975), Vol. 7, No. 1, pp.1-13, 1975.

[5] G. Feng, A Survey on Analysis and Design of Model-Based Fuzzy Control Systems, IEEE Transactions on Fuzzy Systems, Vol. 14, No. 5, pp.676-697, 2006.

[6] P. P. Bonissone, V. Badami, K. H. Chiang, P. S. Khedkar, K. W. Marcelle, M. J. Schutten, Industrial Applications of Fuzzy Logic at General Electric, Proceedings of IEEE, Vol. 83, No. 3, pp.450-465, 1995.

[7] B. Choi, S. Han, S. Hong, Refrigerator Temperature Control Using Fuzzy Logic and Neural Network, IEEE International Symposium on Industrial Electronics, Vol. 1, pp.186-191, 1998.

[8] http://www.johnlewis.com/

[9] http://www.gandhiappliances.com/index.php?main_page=product_info&cPath=10_224&products_id=5847

[10] R. John, S. Coupland, Type-2 Fuzzy Logic: A Historical View, IEEE Computational Intelligence Magazine, Vol. 2, No. 1, pp.57-62, 2007.

[11] D. Wu, W. W. Tan, Genetic Learning and Performance Evaluation of Interval Type-2 Fuzzy Logic Controllers, Engineering Applications of Artificial Intelligence, Vol. 19 No. 8, pp. 829-841, 2006.

[12] N. Sahab, H. Hagras, An Adaptive Type-2 Input Based Non-Singleton Type-2 Fuzzy Logic System for Real World Applications, IEEE International conference on Fuzzy Systems 2011.

[13] A. Perez-Neira, J. Sueiro, and J. Roca and M. Lagunas, A Dynamic Non-singleton Fuzzy Logic System for DS/CDMA Communications, IEEE International Conference on Fuzzy Systems, Vol. 2, pp. 1494-1499, 1998.

[14] M. Benedetti, A. Casagranda, and M. Donelli, and A. Massa, An Adaptive Multiscaling Imaging Technique Based on a Fuzzy-logic Strategy for Dealing with the Uncertainty of Noisy Scattering Data, IEEE Transactions on Antennas and Propagation, Vol. 55, No. 11, pp. 3265-3278, 2007.

[15] C. K. Makropoulos and D. Butler, Spatial decisions under uncertainty: fuzzy inference in urban water management, Journal of Hydroinformatics, IWA publishing, pp. 3-18, 2004.

[16] C. Quek and R. Zhou, Structure and learning algorithms of a nonsingleton input fuzzy neural network based on the approximate analogical reasoning schema, International Journal on Information Science and Engineering Science (Fuzzy Sets and Systems), Elsevier, vol.157, no. 13, pp. 1814-1831, 2006.

[17] E. Cavallaro, S. Micera, and P. Dario, and W. Jensen, and T. Sinkjaer,On the Intersubject Generalization Ability in Extracting Kinematic Information from Afferent Nervous Signals, IEEE Transactions on Biomedical Engineering, Vol.50, No. 9, pp. 1063-1073, 2003.

[18] T. Chua and W. Tan, Non-singleton genetic fuzzy logic system for arrhythmias classification, Engineering Applications of Artificial Intelligence, Elsevier, Vol. 24, No. 2, Pages 251-259, 2011.

[19] G. M. Mendez, L. L. Lezma, R. Colas, G. M. Perez, J. R. Cuellar, J. J. Lopez, Application of Interval Type-2 Fuzzy Logic Systems for Control of Coiling Entry Temperature in Hot Strip Mill, International conference on Hybrid Artificial intelligent Systems, pp.352-359, Springer-Verlag Berlin 2009.

[20] G. M. Mendez, M. A. Hernandez, Interval Type-1 Non-Singleton Type-2 TSK Fuzzy Logic Systems Using the Hybrid Training Method RLS-BP, IEEE Symposium on Foundations of computational Intelligence, pp.370-374, 2007.

[21] G. M. Mendez, A. Hernandez, Hybrid IT2 NSFLS-1 Used to Predict the Uncertain MXNUSD Exchange Rate, Springer-Verlag Berlin Heidelberg 2008, pp. 575-582, 2008.

[22] G. M .Mendez, A. Hernandez, A. Cavazos, M. T. M. Jimenez, Type-1 Non-singleton Type-2 Takagi-Sugeno-Kang Fuzzy Logic Systems Using the Hybrid Mechanism Composed by a Kalman Type Filter and Back Propagation Methods, Springer-Verlag Berlin Heidelberg 2010, pp.429-437, 2010.

[23] G. M. Mendez, L. A. Leduc, First-Order Interval Type-1 Non-singleton Type-2 TSK Fuzzy Logic Systems, Springer-Verlag Berlin Heidelberg ,pp.81-89, 2006.

[24] G. M. Mendez, Interval Type-1 Non-singleton Type-2 Fuzzy Logic Systems are Type-2 Adaptive Neuro-Fuzzy Inference Systems, International Journal of Reasoning-based Intelligent Systems, Vol. 2, No. 2, pp.95-99, 2010.

[25] G. M. Mendez, I. L. Juarez, L. A. Leduc, R. Soto, A. Cavazos, Temperature Prediction in Hot Strip Mill Bars Using a Hybrid Type-2 Fuzzy algorithm, International Journal of Simulation Systems, Science & Technology, pp.33-43, 2005.

[26] G. M. Mendez, M. A. Hernandez, Hybrid Learning for Interval Type-2 Fuzzy Logic Systems Based on Orthogonal Least-squares and Back-propagation Methods, International Journal on Information Science, Vol. 179, No. 13, pp.2146-2157, 2009.

[27] G. Mendez, A. Hernandez, Hybrid Learning of Interval Type-2 Fuzzy Systems Based on Orthogonal Least Squared and Back Propagation for Manufacturing Applications, Journal of Automation, Mobile Robotics & Intelligent Systems, pp.23-32 , 2008.

[28] H. Wu, J. M. Mendel, "Classification of battlefield ground vehicles using acoustic features and fuzzy logic rule-based classifiers", IEEE transactions on fuzzy systems, Vol. 15, No. 1, pp.56-72, 2007.

[29] G. Mendez, M. Hernandez, IT2 TSK NSFLS2 ANFIS, Ninth Mexican International Conference on Artificial Intelligence, IEEE Computer Society, pp.89-93, 2010.

[30] G. Mendez, M. Hernandez, Interval Type-2 Non-singleton Type-2 Takagi-Sugeno-Kang Fuzzy Logic Systems Using the Hybrid Learning Mechanism Recursive-Least-Square and Back-Propagation Methods, IEEE International Conference on Control, Automation, Robotics and Vision, Singapore, pp.710-714, 2010.

[31] P. Torres, D. Saez, Type-2 Fuzzy Logic Identification Applied to the Modeling of a Robot Hand, IEEE International conference on Fuzzy Systems, pp.854-861, 2008.

[32] J. M. Mendel, Uncertain Rule-based Fuzzy Logic Systems, introduction and new direction, Prentice Hall PTR Upper Saddle river, NJ, 2001.

[33] Q. Liang, J. M. Mendel, Interval Type-2 Fuzzy Logic Systems: Theory and Design, IEEE Transactions on Fuzzy Systems, Vol.8, No.5, pp.535-550, 2000.

[34] G. C. Mouzouris, J. M. Mendel, Nonsingleton Fuzzy Logic Systems: Theory and Application, IEEE Transactions on Fuzzy Systems, Vol.5 , No.1, pp.56-71, 1997.

[35] J. L. S. Neto, L. H. Hoang, G. C. Mouzouris, J. M. Mendel, Comments on "Nonsingleton fuzzy logic systems: theory and applications" and Reply, IEEE Transactions on Fuzzy Systems, Vol.6, No.2, pp.325-326 , 1998.

[36] S. Coupland, R. John, Geometric Interval Type-2 Fuzzy Systems, Proceedings of the joint 4th Conference of the European Society for Fuzzy Logic and Technology (EUSFLAT 2005) and the 11th LFA (joint EUSFLAT-LFA 2005), pp.449-454, 2005.

[37] H. Hagras, Type-2 FLCs: A New Generation of Fuzzy Controllers, IEEE Computational Intelligence Magazine, Vol. 2, No. 1, pp.30-43, 2007.

[38] J. M. Mendel, Type-2 Fuzzy Sets: Some Questions and Answers, Feature article in IEEE Neural Network Society, pp.10-13, 2003.

[39] J. M. Mendel, Advances in Type-2 Fuzzy Sets and Systems, Information Sciences 177 (2007) 84-110, 2006.

[40] N. N. Karnik, J. M. Mendel, Q. Liang, Type-2 Fuzzy Logic Systems, IEEE Transactions on Fuzzy Systems, Vol. 7, No. 6, pp. 643-658, 1999.

[41] Q. Liang, N. N. Karnik, J. M. Mendel, Connection Admission Control in ATM Networks Using Survey-Based Type-2 Fuzzy Logic Systems, IEEE Transactions on Systems, Man, and Cybernetics- Part-C: Applications and Reviews, Vol. 30, No. 3, pp.329-339, 2000.

[42] J. M. Mendel, R. B. John, Type-2 Fuzzy Sets Made Simple, IEEE Transactions on Fuzzy Systems, Vol. 10, No. 2, pp.117-127, 2002.

[43] J. Mendel, H. Wu, Uncertainty Versus Choice in Rule-based Fuzzy Logic Systems , IEEE International Conference on Fuzzy Systems, pp.1336-1342, 2002.

[44] J. M. Mendel, Type-2 Fuzzy Sets and Systems: How to Learn About Them, IEEE SMC eNewsletter, 2009.

[45] E. A. Jammeh, M. Fleury, C. Wagner, H. Hagras, M. Ghanbari, Type-2 Fuzzy Logic Congestion Control for Video Streaming Across IP Networks, IEEE Transactions on Fuzzy Systems, Vol. 17, No. 5, pp.1123-1142, 2009.

[46] S. Barkati, E. M. Berkouk , M. S. Boucherit, Application of Type-2 Fuzzy Logic Controller to An Induction Motor Drive with Seven-level Diode-clamped Inverter and Controlled Infeed, Electrical Engineering Journal- Spring Berlin, Vol.90, No. 5, pp.347-359, 2008.

[47] F. J. Lin, P. H. Chou, Adaptive Control of Two-axis Motion Control System Using Interval Type-2 Fuzzy Neural Network, IEEE Transactions on Industrial Electronics, Vol. 56, No. 1, pp.178-193, 2009.

[48] N. Sahab, H. Hagras, A Type-2 Nonsingleton Type-2 Fuzzy Logic System to Handle Linguistic and Numerical Uncertainties in Real World Environments, IEEE International Symposium on Advances in Type-2 Fuzzy Logic Systems 2011.

[49] A. Ferreoro, S. Salicone, Fully comprehensive mathematical approach to the expression of uncertainty in measurement, IEEE Transactions on Instrumentation and Measurement, Vol. 55, No. 3, pp.706-712, 2006.

[50] N. Sahab, H. Hagras, A Hybrid Approach to Modeling Input Variables in Non-Singleton Type-2 Fuzzy Logic Systems, The 10th Annual UK Workshop in Computational Intelligence (UKCI 2010), pp. 1-6, 2010.

[51] F. Liu, J. M. Mendel, An Interval Approach to Fuzzistics for Interval Type-2 Fuzzy Sets, IEEE International Conference on Fuzzy Systems, pp.1-6, 2007.

[52] J. Lamancusa (2000), Outdoor Sound Propagation, Noise Control, ME 458: Engineering Noise Control, Penn State University, pp.10-19, 2009.