Improved Uncertainty Capture for Non-Singleton Fuzzy Systems

Amir Pourabdollah, *Member, IEEE,* Christian Wagner, *Senior Member, IEEE,*
Jabran Hussain Aladi, *Student Member, IEEE* and Jonathan M. Garibaldi, *Member, IEEE*

Abstract—In non-singleton fuzzy logic systems (NSFLSs), input uncertainties are modelled with input fuzzy sets in order to capture input uncertainty (e.g., sensor noise). The performance of NSFLSs in handling such uncertainties depends on both: the appropriate modelling in the input fuzzy sets of the uncertainties present in the system's inputs, and on how the input fuzzy sets (and their inherent model of uncertainty) interact with the antecedent and thus affect the inference within the remainder of the NSFLS. This paper proposes a novel development on the latter. Specifically, an alteration to the standard composition method of type-1 fuzzy relations is proposed, and applied to build a new type of NSFLS. The proposed approach is based on employing the centroid of the intersection of input and antecedent sets as origin of the firing degree, rather than the traditional maximum of their intersection, thus making the NSFLS more sensitive to changes in the input's uncertainty characteristics. The traditional and novel approach to NSFLSs are experimentally compared for two well-known problems of Mackey-Glass and Lorenz chaotic time series predictions, where the NSFLSs' inputs have been perturbed with different levels of Gaussian noise. Experiments are repeated for system training under noisy and noise-free conditions. Analyses of the results show that the new method outperforms the traditional approach. Moreover, it is shown that while formally more complex, in practice, the new method has no significant computational overhead compared to the standard approach.

Index Terms—non-singleton, fuzzy logic systems, uncertainty, time series prediction

I. INTRODUCTION

FUZZY Logic Systems (FLSs) have shown their effectiveness in a wide range of applications including engineering, natural science and time-series prediction [1]–[3]. Handling uncertain and vague information has been at the forefront of FLSs since the introduction of fuzzy sets (FSs) by Zadeh in 1965 [4]. FLSs have been employed to handle uncertainties in different forms such as uncertainty associated with words (e.g., *Computing with Words* [5]) and with the input data (e.g., in the form of sensor noise).

While Singleton FLSs (SFLSs) are the most common types of FLSs, Non-Singleton FLSs (NSFLSs) [6], which are specifically designed for handling the uncertainties associated with the inputs to a FLS, also exist. NSFLSs are types of FLSs where the input uncertainty is modelled by distributions (input FSs) [7], rather than singleton FSs as is the case for SFLSs. For example, a FS with a Membership Function (MF) *around*

The authors are with the Lab for Uncertainty in Data and Decision Making (LUCID), the Intelligent Modelling and Analysis Research Group (IMA) and the Horizon Digital Economy Research Institute, School of Computer Science, the University of Nottingham, Nottingham, UK. (email: amir.pourabdollah| christian.wagner|itxjha|jon.garibaldi@nottingham.ac.uk)

x (e.g., a Gaussian distribution centred on x or a sensor-specific MF [8]) is used as the *input FS* [9]. In fact, SFLSs are special cases of NSFLSs where the input is represented by a singleton FS (i.e. a FS with a membership of 1 at x and 0 elsewhere). Although the theory of NSFLSs has been established for many years (e.g., in [6]), and the capacity of NSFLSs to deliver superior performance in comparison to SLFSs has been shown repeatedly [10], [11], the application of NSFLSs is still rare in comparison to SFLSs. The commonly cited reason for this is the additional computational and design complexity of NSFLSs compared to SFLSs [10], [11].

One way of interpreting the low use of NSFLSs is that the balance between performance gain and additional complexity has not been sufficient to warrant the additional complexity of their implementation as part of real world applications (some of the user-related challenges and practical solutions in a case of environmental management are discussed in a previous work [12]). This paper seeks to further extend the capacity of NSFLSs to deliver improved performance without increasing their design complexity, thus hopefully supporting the increased applicability of NSFLSs in applications.

We propose a novel inference method to be used in NS-FLSs to improve their performance without adding significant complexity/computation cost, by introducing a new method of interaction between the antecedent(s) and the fuzzy input set(s). The new method is based on altering the mathematics of the standard *composition of fuzzy relations* method, such that the centroid of the fuzzy sets' intersection is used for calculating each rule's firing strength instead of their intersection's maximum. While this change is not complex, it fundamentally affects the whole inference process and enables NSFLSs to more closely model (and react to) input uncertainties. The applied change in the composition method will be formally detailed in Section 3 after the standard method of composition is described in Section 2.

In order to examine the proposed method in practice and to assess its overloaded complexity, we apply it to two well-known problems of Mackey-Glass and Lorenz chaotic time series prediction which provide verifiable ground truth, enable easy reproduction of our results by other researchers, and enable us to easily modulate the different characteristics such as the level of uncertainty/noise in the experiments. Moreover, standard NSFLS solutions have already been designed for such time series prediction applications (in [6], [7]) and thus can serve as benchmarks for our experiments.

The structure of this paper is as follows: First, we review the background and the related works. We proceed by proposing

an alternative composition method for application in NSFLSs and theoretically compare it with the traditional approach. In Section IV, we experimentally compare new and traditional NSFLSs using the application of Mackey-Glass and Lorenz time-series prediction. Section V provides an analysis of the computational complexity of the proposed methods in relation to traditional NSFLSs. Finally, we conclude and provide the direction of future work in Section VI.

II. BACKGROUND

In this section we briefly introduce background material, including the composition of fuzzy relations, NSFLSs and the related works.

A. Fuzzy Relations and Their Composition

As the name implies, fuzzy relations describe the relation between two or more FSs. The concepts of fuzzy relations and their compositions are briefly reviewed here as they play an important role in NSFLSs' input-to-output mapping. Specifically, it will be shown in the next sub-section, how the method of fuzzy relations composition determines the inference method used in NSFLSs.

In order to have a formal review, we briefly describe the mathematical backgrounds of the standard method of fuzzy relations composition. If X and Y are two FSs with two universes of discourse, and $(X \times Y)$ is their Cartesian Product over the two universes, a *fuzzy relation* [7] between the two FSs is another FS called R, defined as:

$$R(X,Y) = \{[(x,y), \mu_R(x,y)] \mid (x,y) \in (X \times Y)\} \quad (1)$$

If, for three FSs X, U and Y, two fuzzy relations $P(X,U)$ and $Q(U,Y)$ are defined, the *fuzzy composition* of the two relations is another fuzzy relation denoted as $P \circ Q(X,Y)$ with a MF defined as:

$$\mu_{P \circ Q}(x,y) = sup_{u \in U}[\mu_P(x,u) \star \mu_Q(u,y)] \quad (2)$$

where sup is the supremum operator and $\star$ is any t-norm operator [13]. Equation (2) is also known as *sup-star* composition in the literature.

As a special case, a FS and a fuzzy relation can also be the two parts of a fuzzy composition. If unlike Q, P is a FS (not an explicit relation between two FSs), it can be considered as a relation between X and X, i.e., $X = U$ and $\mu_P(x,u)$ is reduced to $\mu_X(x)$. In this case, (2) is rewritten as:

$$\mu_{X \circ Q}(x,y) = sup_{x \in X}[\mu_X(x) \star \mu_Q(x,y)] \quad (3)$$

Equation (3) is particularly useful for input-to-output mapping in the inference engine of NSFLSs. This will be described in the next sub-section after NSFLSs are introduced.

B. Non-Singleton FLSs

1) Overview: In order to address uncertainty in the actual inputs to the FLS, the fuzzifier in NSFLS maps a given crisp input to a fuzzy input set, rather than to a fuzzy singleton as is the case in SFLSs. Thus, as shown in Fig. 1, the overall
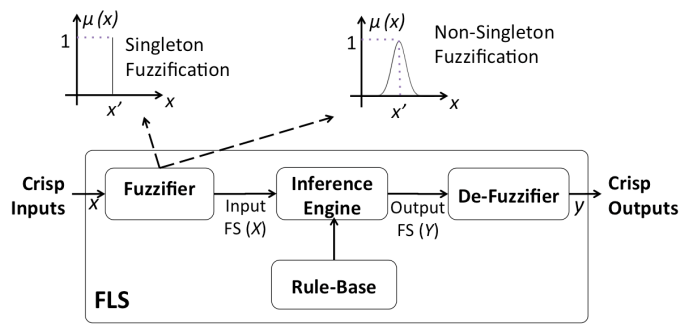

Fig. 1. FLS components and the illustration of different fuzzification methods. Singleton and non-singleton fuzzifications of a sample input x' are also shown.

components of a SFLS and a NSFLS are identical, the only difference is the handling of the crisp inputs in the fuzzifier.

In NSFLSs, the appropriate type of membership function to be employed for the input FSs is application dependent, i.e. it is dependent on the characteristics of the uncertainty affecting the system's inputs, with the most common being a type of fuzzy number, i.e. a convex, normal FS [7]. In Fig. 1, a Gaussian distribution is shown as an example.

2) The Composition of Fuzzy Relations used in NSFLSs: FLSs use different FSs (input, antecedent, and consequent -for Mamdani FLSs- FSs) and their compositions to establish the mathematical relationships between inputs and outputs. For Mamdani's inference method [14] in SFLSs for example, the firing strength of each rule depends on the firing degree of each antecedent MF for each crisp (singleton FS) input. In NSFLSs, the computation is more complex since each antecedent has to be combined with (non-singleton) FSs in order to work out the firing strength of each rule.

The general mapping between NSFLSs' inputs and outputs, i.e. between input set X and output set Y in Fig. 1, is comprehensively detailed in [7]. We do not repeat the details of how the comprehensive formula for the mapping is derived, instead we focus on the results at an abstract level.

For illustration, consider a single-input, single-rule and single-output system of Fig. 1 where Mamdani implication is used. Let x and y be members of input and output FSs (X, Y) and let A and C be two FSs representing an antecedent and a consequent. The only defined rule is *If x is A then y is C*. We also define $\mu_X(x)$, $\mu_A(x)$, $\mu_C(y)$ and $\mu_Y(y)$ as the MFs of X, A, C and Y respectively. We notice that the output set of a one-rule inference engine is a composition between its input FS and a FS that is determined by a rule (each rule is considered as a fuzzy relation between antecedent(s) and a consequent, i.e. between input space and output space - more details can be found in [7]).

Equation (3) has already formulated the composition between a fuzzy set X and a fuzzy relation Q. Thus in (3), X can represent the input FS while the relation Q represents a rule. In such NSFLS, $\mu_Q(x,y)$ can also be written as $\mu_A(x) \star \mu_C(y)$. Equation (3) can then be written for the described NSFLS as:

$$\mu_{X \circ Q}(x,y) = sup_{x \in X}[\mu_X(x) \star \mu_A(x) \star \mu_C(y)] \quad (4)$$

Since $\mu_C(y)$ is independent of x, it can be taken out of the supremum operator, leading to an input-to-output mapping
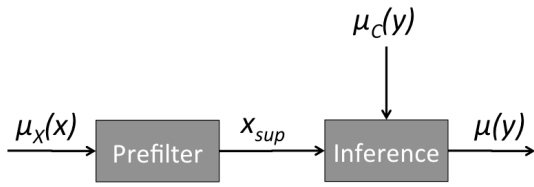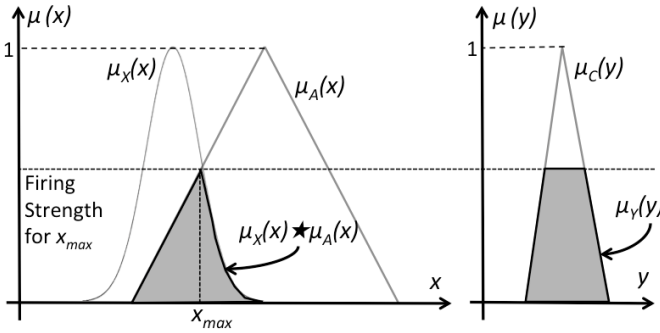
Fig. 2. Prefiltering of input FS to a NSFLS

Fig. 3. The illustration of how a standard NSFLS calculates its output (Y) according to its input (X), antecedent (A) and consequent (C) FSs.

expression:

$$\mu_Y(y) = \mu_C(y) \star sup_{x \in X}[\mu_X(x) \star \mu_A(x)] \qquad (5)$$

or equally;

$$\mu_Y(y) = \mu_C(y) \star \mu_X(x_{sup}) \star \mu_A(x_{sup}) \qquad (6)$$

where x_{sup} is the value of x at which $\mu_X(x) \star \mu_A(x)$ takes its maximum. Equation (6) is a reduced version of the general input-to-output mapping derived in [7] for the described simplified NSFLS. The inference engine in a NSFLS can then be imagined as a *prefilter* unit added to an inference unit, in which the prefilter unit transforms the uncertain input set to a representative numerical value x_{sup} (Fig. 2) [7]. Handling the input uncertainty in NSFLSs is concentrated in the prefiltering unit, i.e. the rest of the FLS acts identically to a singleton FLS.

In discrete systems (including the majority of FLS applications) the supremum operator is replaced by the maximum operator and x_{sup} in equation (6) is replaced by x_{max}. The most commonly used forms of sup-star composition in discrete systems are *max-min* and *max-product* compositions. In this paper we focus on the max-min composition, so $\mu_X(x) \star \mu_A(x)$ is the intersection of X and A, and $\mu_X(x_{max})$ and $\mu_A(x_{max})$ are equal. As such, (6) can be written as:

$$\mu_Y(y) = min[\mu_A(x_{max}), \mu_C(y)] \qquad (7)$$

Briefly, this formula tells us that *the firing level of an antecedent is the peak of its intersection with the input set* (Fig. 3). The above formula is for the simplified form of NSFLS. For the general NSFLS case, the same formulation is iterated for an arbitrary number of inputs and rules (detailed in [7]).

We have briefly reviewed the origins of using sup-star and max-min compositions for NSFLS input-to-output mappings. Selecting x_{sup} (and x_{max}) to be the output of prefiltering is

a direct result of utilizing the sup-star composition of fuzzy relations. However, the introduced methods are not the only possible methods. Section 3 is about considering the possible benefits of using an alternative method.

Before introducing the alternative in section 3, we briefly review in the next sub-section, the other related works about the theory, applications and performances of NSFLSs, with a focus on our selected benchmark problem of noisy time-series prediction.

C. Other Related Work

In addition to the classical literature about the NSFLS fundamentals and using the standard composition method (e.g., [13], [15]), there are a number of other research works that are connected to the focus of this paper.

Regarding NSFLS application for uncertain reasoning and prediction, in [16] NSFLSs are utilized to forecast time series from the domain of financial markets. After establishing the theory and application of NSFLS and its application in noisy time-series prediction problems ([6], [7], [17]), some practical prediction applications were also examined. For example in [18], the Box-Jenkin's gas furnace time series data is used in conjunction with NSFLS-based prediction, showing that the employed NSFLS provides better prediction in comparison to the SFLS. Among the different time-series prediction problems, predicting Mackey-Glass time series [19] has been a benchmark problem in the literature which has been considered by a number of researchers, e.g., in [20]–[23].

Beyond the development and application of NSFLSs in type-1 fuzzy systems, there have been substantial advances in the application of non-singleton fuzzification as part of type-2 FLSs [24]. The recent developments of the theory and applications of type-2 FLSs have shown the advantages of using type-2 FLSs (both general and interval type-2 FLSs) over type-1 FLSs when the uncertainties at different levels increase (e.g., in noisy time series prediction [25], [26] and in noisy image processing [27]). Further, non-singleton inputs have also been applied to type-2 FLSs, resulting in hybrid solutions for uncertainty handling at different uncertainty levels. For example in [28] and [29], non-singleton type-2 FLSs are shown to effectively handle different levels of uncertainties in industrial use cases. It has also been shown that in the presence of perturbations, the generalized type-2 fuzzy controllers outperform their type-1 and interval type-2 counterparts [30], [31], and that interval type-2 NSFLSs outperform type-1 NSFLSs in different application domains [32], [33]. Incorporating other AI techniques with type-2 FLSs has also resulted in hybrid solutions for predicting the behaviour of non-linear complex systems (e.g., using neural networks in [34] and genetic algorithms in [35]).

For optimizing the non-singleton fuzzification performance, the approaches taken in [17], [36]–[38], are to dynamically tune the parameters of the input fuzzy sets of type-2 FLSs according to the characteristics of the input uncertainties. While some researchers have focused on comparing non-singleton type-1 with singleton type-2 FLSs (e.g., in [9], [39]), others have taken the advantages of the both systems by

developing type-2 non-singleton type-2 FLSs (i.e. type-2 FLSs that use non-singleton type-2 input fuzzy sets), e.g., in [11], [40]. The current paper focuses specifically on type-1 non-singleton type-1 FLSs but the proposed concepts are equally applicable to type-2 FLSs as the proposed changes are at the inference level of NSFLSs applicable to both type-2 and type-2 NSFLSs. An analysis of their application in the type-2 FLS context will be addressed in a future paper.

III. A Novel Approach for NSFLSs

In this section we investigate whether or not x_{sup} (see Fig. 2) as a result of the prefiltering stage is the only possible approach or whether an alternative to the sup-star composition can provide superior results. We discuss the motivation both from the perspective of modelling the input uncertainty with high fidelity as well as from a formal perspective of the underlying composition. Specifically, we will highlight that following the traditional approach to NSFLSs, selecting x_{sup} may lead to the loss of valuable information encoded in the input FS membership function (which captures the uncertainty of the FLS input).

A. Motivation

In the previous section, it was highlighted that the prefiltering stage is the critical step in NSFLSs' functionality. This stage is where for each rule, sup-star composition is used to combine the information encoded in the input set(s) with that of the antecedent set(s) to generate the firing level of the rule. Thus, it is clear that any change in the composition method will fundamentally impact the performed inference, and consequently the system's output. In this paper we are particularly interested in establishing an efficient composition method which can perform the combination between input and antecedent sets as comprehensively as possible - minimizing information loss about the distribution of uncertainty in the input fuzzy sets' membership functions.

The traditional approach to NSFLSs provides only a limited capacity for capturing this uncertainty as the degree of firing of a given antecedent FS is the result of a simplified process for extracting the firing strength from the intersection of input and antecedent FSs as clarified in the examples below. Importantly, while the change to this process of finding the appropriate firing strength proposed in this paper may seem small, it is an essential part of the resulting NSFLSs which significantly affects their capacity to model the actually uncertainty present, and thus to provide superior results for appropriately modelled FSs. The latter finally is vital in the more general context of this work: fuzzy systems are credited widely with dealing well with uncertainty in systems. However, exactly *how* to leverage this capacity in the best way possible based on insight on real applications (and uncertainties present) is still not well understood. Providing mechanisms such as in this paper which clearly and comprehensively link the uncertainty captured by input fuzzy sets to the antecedents and thus performance of the fuzzy system are vital as key steps on advancing the research around fuzzy systems as a whole.

In other words, a deviation from the traditional sup-star composition methods that can capture the interaction of input and antecedent FSs as accurately as possible, may provide a pathway for the uncertainty of the input to be captured in the input FSs with high fidelity and translating the effect of these models consistently and efficiently to FLS outputs. In turn, this should enable the separation of input uncertainty and linguistic antecedent uncertainty modelling, both of which are commonly "mixed" in the input and antecedent FSs. We feel this is valuable and important, in particular as it enables the capture and modelling of uncertainties *where they arise*, i.e. separately in the inputs (e.g., a sensor) or in the antecedent (e.g., experts are unsure about the linguistic labels - what is "low"). The latter is vital to pave the way for the development of improved, systematic design methods for real world fuzzy systems.

To illustrate, we provide two examples in Fig. 4. In each, different input FSs in a typical NSFLS context are intersected with an antecedent. As the actual input FSs are different, we expect a different firing level and thus output of the FLS. However, the actual firing levels are the same in both cases, i.e. $\mu_{X1}(x_{max}) = \mu_{X2}(x_{max})$. A different way of looking at this is that the differences in the uncertainty models (captured by the inputs FSs) are not reflected in the firing strengths and thus also not in the NSLS outputs.

The latter is particularly important when input fuzzy sets are modelled precisely to capture the actual levels of uncertainty encountered in the real world, such as is the case when they are designed based for example on prior modelling [8], [11]. Employing fuzzy systems which are designed based on the modelling of the real world uncertainty present in an environment has considerable potential for providing meaningful and efficient pathways for the better design of fuzzy systems. However, in order to enable this, a clear interrelation between the actual uncertainty models and their effect needs to be in place. Establishing this link by improving on the traditional approach to NSFLS is the aim of this paper.

Based on the above, it is arguable that x_{max} may not necessarily be the best choice in terms of capturing the interaction (i.e. intersection) between the input and antecedent FSs with the highest fidelity possible. We will specifically note that certain alternatives to sup-star composition are theoretically valid. Consequently, in NSFLSs, it seems desirable to identify an approach which results in a better capture of the detailed interaction of the input and antecedent set, thus preserving more information captured in the input FS and thus input uncertainty model.

Before formally defining any alternative composition method, we review the theory of the composition of fuzzy relations to identify any formal limitations for developing new alternatives that are not necessarily based on sup-star composition.

B. The Validity of Alternative Composition Methods

To see if the literature provides any specific formal uniqueness for using the supremum (or maximum) operator, we note that in formulating the sup-star composition (2) in [7] and
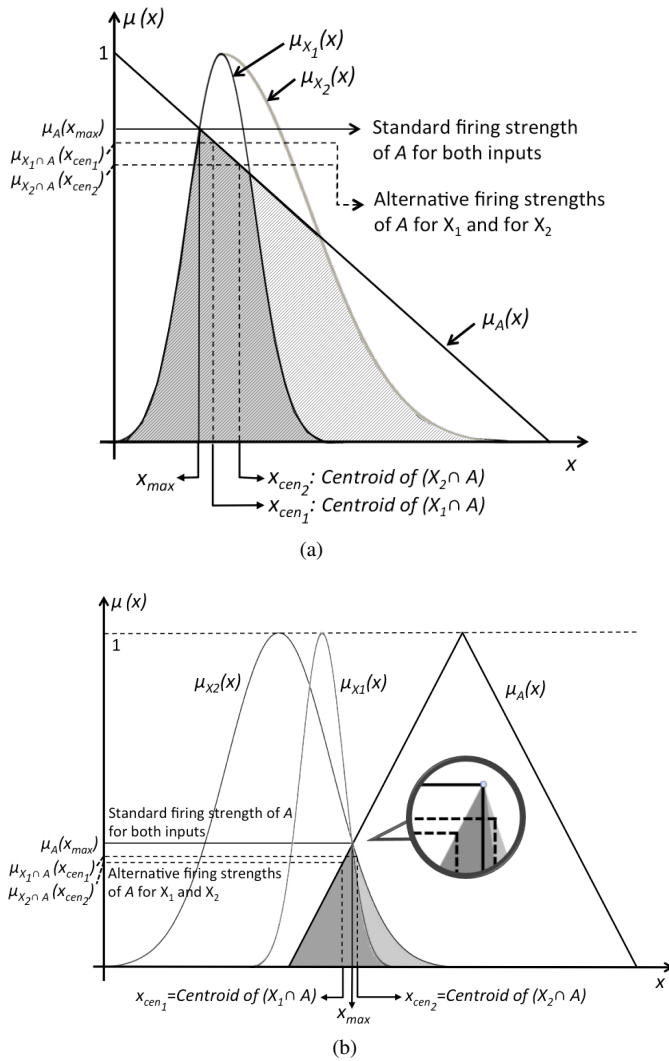
Fig. 4. In two examples (a) and (b), the intersection of two different input fuzzy sets X_1 and X_2 with a single antecedent fuzzy set A are shown, where the firing strengths for the standard NSFLS approach are equal for both inputs. For the proposed approach, the alternative firing strengths arising from using the centroid of each intersection instead of their maximum are also shown.

[13], first a formula for the crisp sets has been proven, then the proof is extended to fuzzy relations: Let $\dot{X}$, $\dot{U}$ and $\dot{Y}$ be crisp sets, and two crisp relations $P(\dot{X}, \dot{U})$ and $Q(\dot{U}, \dot{Y})$ defined over the crisp sets. If $P \circ Q$ is the composition of the two relations (i.e. mapping between $\dot{X}$ and $\dot{Y}$), it was shown in [7] that the crisp MF of such a relation can only be:

$$\mu_{P \circ Q}(\dot{x}, \dot{y}) = max_{\dot{u} \in \dot{U}}[\mu_P(\dot{x}, \dot{u}) \star \mu_Q(\dot{u}, \dot{y})], \qquad (8)$$

where $\star$ is any t-norm operator and $\mu()$ is the MF of crisp sets, i.e. either 0 or 1. The same formula was extended to fuzzy sets which led to the sup-star composition (2). Thus, in the transition from crisp to fuzzy domains, the max operator in the crisp domain was replaced by the same operator in fuzzy domain. In other words, the step of choosing 1's over 0's has been replaced by choosing the maximum of some real numbers in the range of [0, 1]. The justification provided for this replacement in [7] and [13] is that it is *reducible* to the crisp domain. Naturally, a number of different operations equally follow this justification, i.e. any other fuzzy operation

that can be reduced to the max in the crisp domain may be considered.

Based on the motivation and the formal context, we proceed to introduce an alternative composition method which can be employed as part of novel NSFLSs as will later be shown.

C. The Centroid-based Composition

We propose an alternative to the traditional sup-star composition that can be applied in NSFLSs to better capture the interaction of input and antecedent FSs. Since we are interested in taking as much of the information about the *shape* of FSs into account as possible, our choice is computing the *centroid* of the resulting intersection of input and antecedents FSs.

More formally, let cen define an operator for FSs as:

$$cen_{x \in X}(X) = \mu_X(x_{cen}), \qquad (9)$$

where x_{cen} is the centroid of X defined as:

$$x_{cen}(X) = \frac{\int_{x_{min}}^{x_{max}} x\mu_X(x)dx}{\int_{x_{min}}^{x_{max}} \mu_X(x)dx} \qquad (10)$$

For crisp sets, where $\mu_X(x) = 1$ just for a single value $x = x_{max}$, (10) reduces to $x_{cen} = x_{max}$. Thus, note that cen operator is also reducible to max operator for crisp sets.

In discrete domain, the above equation can be rewritten as:

$$x_{cen}(X) = \frac{\sum_{i=1}^{n} x_i\mu_X(x_i)}{\sum_{i=1}^{n} \mu_X(x_i)}, \qquad (11)$$

where n is the number of discretization levels.

A new general alternative to (2) for fuzzy relations composition can now be defined as:

$$\mu_{P \circ Q}(x, y) = cen_{u \in U}[\mu_P(x, u) \star \mu_Q(u, y)] \qquad (12)$$

By analogy to the sup-star composition (2), we refer to the new composition as *cen-star* composition. When applying the cen-star composition in the context of NSFLSs, the input-to-output mapping (5) results:

$$\mu_Y(y) = \mu_C(y) \star \mu_{X \cap A}\big(x_{cen}(X \cap A)\big), \qquad (13)$$

where x_{cen} is the value of x at the centroid of the intersection between the MFs of an input X and an antecedent A. Note that while in max-min composition, $\mu_A(x_{max}) = \mu_C(x_{max})$, in the new formulation this is not necessarily the case, thus resulting in the requirement for specifying the membership degree in the actual intersection, i.e. $\mu_{X \cap A}(x_{cen})$. If the minimum-operator is used as the t-norm, (13) can be written as:

$$\mu_Y(y) = min[\mu_C(y), \mu_{X \cap A}\big(x_{cen}(X \cap A)\big)]. \qquad (14)$$

Briefly, Equation (14) tells us that the firing level of a given antecedent is given by the degree of membership of the centroid of the intersection within the membership function formed by intersecting antecedent and input FSs. Figure 4 illustrates this for the two inputs FSs $X1$ and $X2$, resulting in the firing levels $\mu_{X1}(x_{cen1})$ and $\mu_{X2}(x_{cen2})$ respectively. Note that the firing levels are not equal, as they were for the standard NSFLS approach. This alternative *cen-min* approach

will be used hereafter as an alternative to (7) in mapping between the input and output of NSFLSs. We refer to such a NSFLS as *Cen-NSFLS* hereafter, whereas the standard NSFLS based on max-min composition is referred to as *Standard-NSFLS*. As mentioned before, equations (14) and (7) are developed for a simplified one-rule NSFLS, but they are used iteratively (over all inputs, antecedents and rules) to make a general input-to-output mapping, similar to the iteration explained in [7] for max-min composition.

The following section focuses on the exploration of this new version of composition through a series of experiments and analysis.

IV. EXPERIMENTS AND RESULTS

Following the proposition of the cen-min composition method in (14), its applicability in NSFLSs is examined in this section. Specifically, we aim to shed light on the intuitive hypothesis that the novel composition based on the centroid provides a superior integration and modelling of the uncertainty in system inputs and antecedents, in turn resulting in superior performance when the input uncertainty is captured within the input FSs of NSFLSs. In this context, we design a standard NSFLS and a Cen-NSFLSs for the prediction of arbitrary Mackey-Glass (will be called M-G hereafter) and Lorenz chaotic time series under different Gaussian noise conditions (reflected in the different input FSs), and compare the results. In addition we compare both NSFLSs to a standard SFLS.

In the next subsections, firstly the M-G time series prediction experiment is explained in details. Secondly, the results of the similar experiment on Lorenz time series will briefly be presented. Finally, the results will be discussed in general.

A. Method of M-G Time Series Experiment

The method described in [6], uses standard NSFLSs for the prediction of M-G time series in noisy conditions and compares the results to SFLSs. We follow a similar approach but conduct the same experiment using a standard NSFLS and a Cen-NSFLS, as well as a SFLS. The prediction performance is evaluated based on the MSE (Mean Square Error) indicating the deviation of the prediction from the actual expected value. We then compare the calculated MSEs for the individual experiments. Regarding the rule-base generation, we follow the method given in [41], an established approach to learning rules from existing input-output pairs.

M-G time series is characterized by a differential equation [19] as:

$$\frac{dx(t)}{dx} = \frac{0.2x(t-\tau)}{1+x^{10}(t-\tau)} - 0.1x(t) \qquad (15)$$

For $\tau > 17$, (15) demonstrates a chaotic behaviour. We have selected $\tau = 30$. Using (15), $x(t)$ is calculated for 2000 consecutive time points, i.e.: $t = -999$ to $t = 1000$. The first 1000 points are for the initial transients to die out, then using points $t = 1$ to $t = 700$ the system is trained to develop its rule-base. The last 300 points from $t = 701$ to $t = 1000$ are used for testing the system. Rules are trained according to the one-pass method described in [41]. Nine past points in

the time series are employed as inputs to generate a predicted value. Seven equally-distributed triangular MFs are also used to model the input domains.

The work in [6] used noisy data for both training and testing. While this is the case in most real world scenarios, it is also practically possible that a system is trained in noise-free (lab-) conditions then be used in a noisy real-world setting. To take this into account, we repeat our experiments for the two cases, i.e., where the system is trained in either noise-free or noisy conditions. In noise-free training, the same rule-base generated for the SFLS is used whereas in noisy training the rule-base is re-trained with noisy inputs.

Three FLSs are designed: a SFLS, a standard NSFLS and a Cen-NSFLS. For all the systems, centroid deffuzification and Mamdani inference is used with min and max operators for the t-norm and t-conorm respectively. The same discretization levels (100) is used for all FLSs. The input models used for both NSFLSs are Gaussian MFs centred on the crisp (noise-free) inputs, with a standard-deviation equal to that of the noise added in the given experiment. We note that the Gaussian fuzzifier employed for the NSFLSs is not necessarily the best approach, but we feel it provides a reasonable choice which should allow the NSFLS to "track" increasing uncertainty in the inputs. For both NSFLS, two experiments are conducted for different signal-to-noise ratios (SNRs) of 10dB and 5dB.

The described combinations of the system configuration leads to 8 individual systems/experiments in addition to the singleton FLS experiment. Each experiment generates 300 outputs which are compared to the pre-computed outputs calculated by (15). The MSE over the 300 points is used as a measure of the overall error of a given FLS. Note that the aim of the experiments is the comparison of the individual types of FLS, i.e. the relative best performance. We are not seeking to build a "best" time series prediction FLS for which one could employ more FSs, a different strategy for rule creation, etc. In order to mitigate the effect of randomness, each experiment is repeated 30 times and the generated MSEs are averaged. This means that there are 270 individual experiments in total. The experiment design is illustrated in Fig. 5. The next sub-section provides the experiments' results.

B. Results of M-G Time Series Experiments

In order to illustrate the results, we provide both visual representations of individual experiment outputs as well as tables capturing the MSE results averaged over the 30 repeated experiments.

First, we focus on the prediction of the SFLS. The SFLS is trained using noise-free data for $0 < t \leq 700$ in which 184 rules are generated. Then the SFLS is executed to compute its outputs over $700 < t \leq 1000$. The prediction results (Fig. 6) shows that the system is relatively well trained and closely follows the pre-computed ground truth. This was anticipated based on the results shown in [6]. The averaged MSE over 30 FLS runs is 0.0014. Note that no noise is added at this stage, and the SFLS experiment is merely conducted as a benchmark.

Secondly, we focus on the NSFLSs and show the time series prediction in different noise conditions. We do not compare
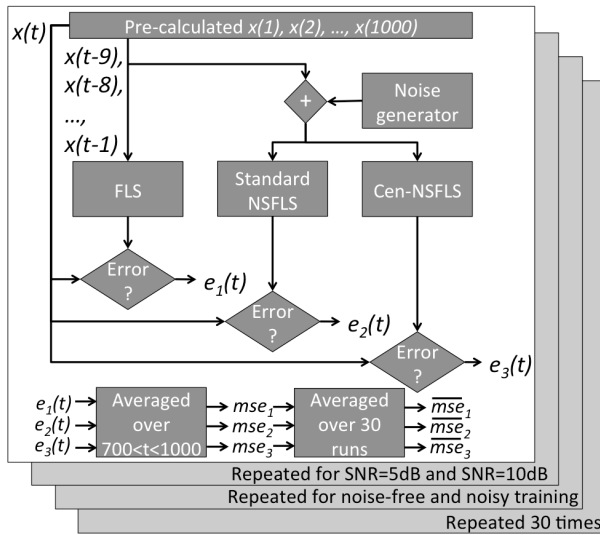
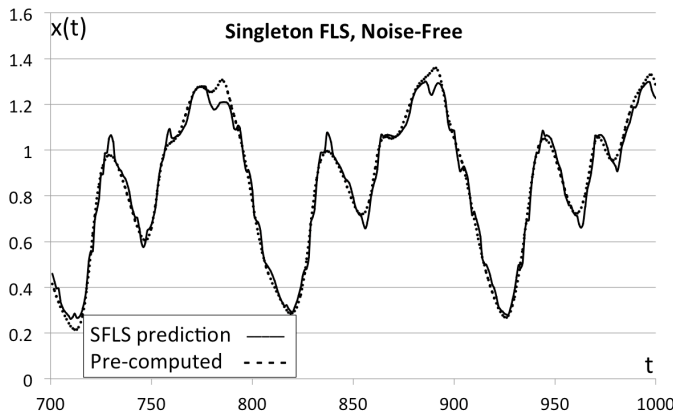Fig. 5. Illustration of the experiment design

Fig. 6. SFLS output when trained with noise-free data compared to the pre-computed dataset.
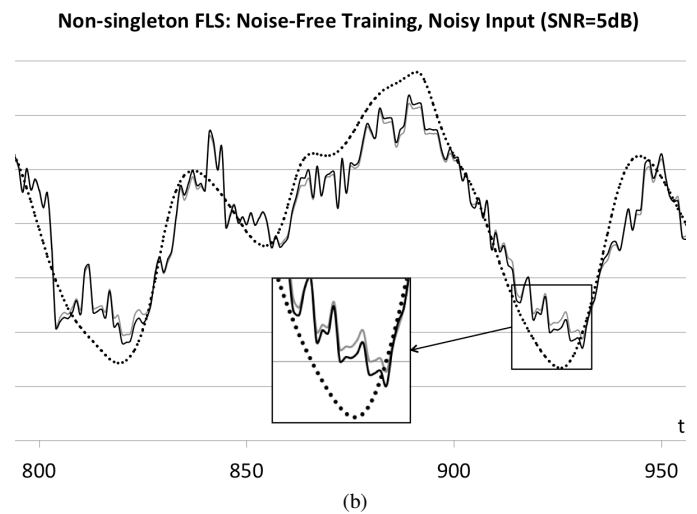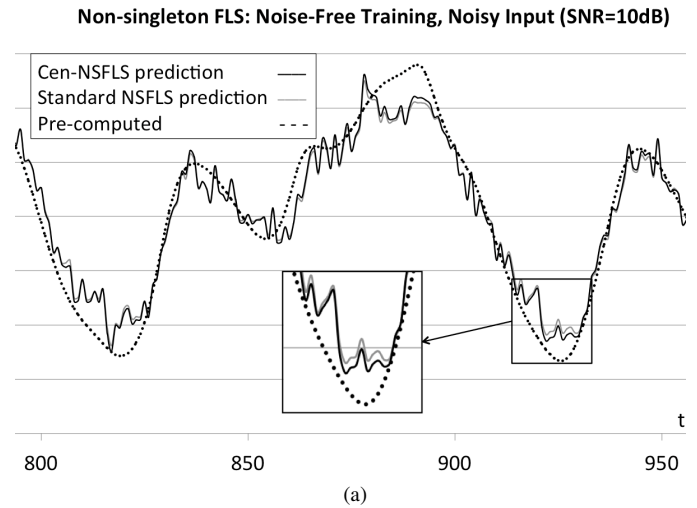

Fig. 7. (a) Comparing the standard and the Cen-NSFLS outputs, when they are trained with noise-free data and tested with noisy data with SNR=10dB. (b) the same for SNR=5dB.

the results of the NSFLS with those of the SFLS (under noise) in this paper, since the same comparison has already been done in [6] where the authors showed that NSFLSs are more capable in handling noisy inputs than SFLS in different noise conditions. Instead, we focus on the comparison of the standard NSFLS and the Cen-NSFLS. While the noise-free training had produced 184 rules, the training process under noise produced 557 rules for SNR=10dB and 664 rules for SNR=5dB. In all the cases, noisy data is used for testing as is intuitive in real-world applications.

Fig. 7 shows the NSFLS predictions when trained with noise-free data and tested with noisy inputs with SNR=10dB and SNR=5dB. This figure and specifically Table I highlight that the Cen-NSFLS shows a reduced deviation from the pre-computed time series and thus better performance in comparison to the standard NSFLS. Specifically, with SNR=10dB, the averaged MSE dropped from 0.0067 to 0.0058 (reduced by 13.19%). Also in SNR=5dB, the averaged MSE has changed from 0.0155 to 0.0135 (reduced by 12.61%).

The same experiments are conducted for the case of training with noisy data. Fig. 8 and Table I show the results in two

noise conditions. The average MSE changed from 0.0124 to 0.0114 (7.71% reduction) for SNR=10dB and from 0.0406 to 0.0332 (17.31% reduction) for SNR=5dB.

TABLE I
THE PREDICTION PERFORMANCE (FOR M-G TIME SERIES) PRODUCED BY THE DIFFERENT FLSS BASED ON AVERAGE MSES (MEAN SQUARE ERRORS).

System	SNR (dB)	Training	MSE (Standard NSFLS)	MSE (Cen-NSFLS)	Change (%)
SFLS	n/a	n/a	0.00138	0.00138	0.00
NSFLS	10	noise-free	0.00673	0.00584	-13.19
NSFLS	5	noise-free	0.01546	0.01351	-12.61
NSFLS	10	noisy	0.01236	0.01141	-7.71
NSFLS	5	noisy	0.04016	0.03320	-17.31

C. Lorenz Time Series Experiment

In addition to the M-G time series, we also compare the NSFLSs' performances using another known time series in hydrodynamics and meteorology, namely the Lorenz time

Non-singleton FLS: Noisy Training, Noisy Input (SNR=10dB)

(a)

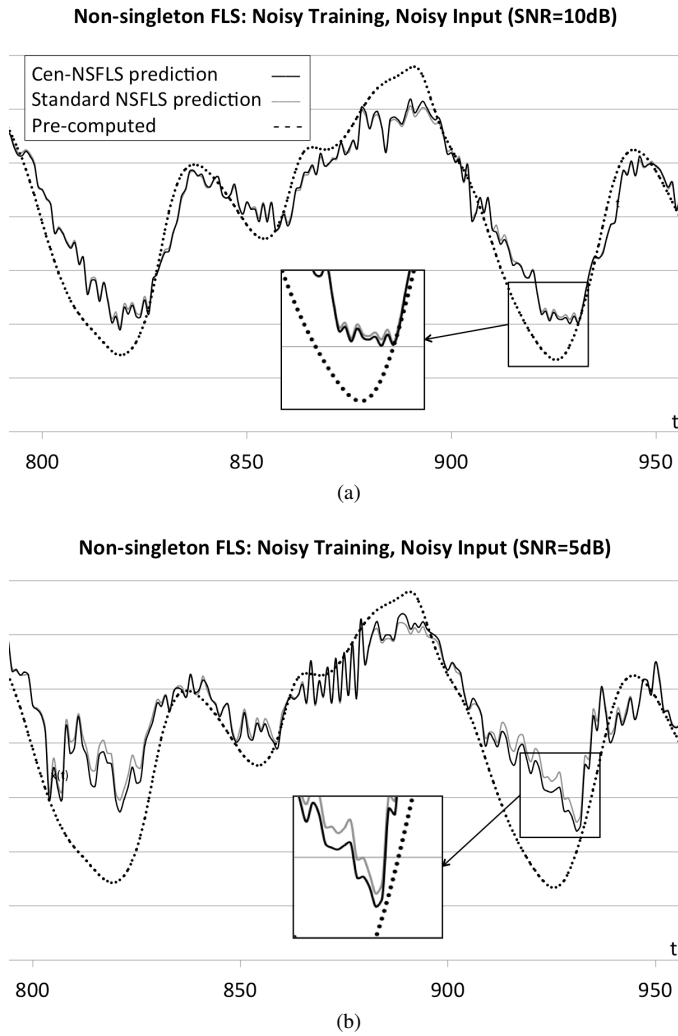**Non-singleton FLS: Noisy Training, Noisy Input (SNR=5dB)**

(b)

Fig. 8. (a) Comparing the standard and the Cen-NSFLS outputs, when they are trained and tested with noisy data with SNR=10dB. (b) the same for SNR=5dB.

series [42]. We consider the time series associated to variable x of the three-dimensional Lorenz differential equations:

$$\dot{x} = \sigma(y - z); \quad \dot{y} = rx - y - xz; \quad \dot{z} = xy - bz \quad (16)$$

where the dots denote the next values to the three variables x, y, z in the time series. To demonstrate a chaotic behaviour, the attributes σ, b and r are respectively set to 10, $\frac{8}{3}$ and 28, as suggested in [42]. We do not repeat the details of the experiment here, as the same method is already detailed for the M-G problem. The summary of the results (Table II) show that similar to the M-G experiment, the prediction errors (MSEs) produced by the Cen-NSFLS is generally lower than the MSEs produced by the standard NSFLS (between 3.11% to 11.22% in different experiment settings).

D. Discussion

The summary of calculated MSEs and their improvements for the M-G time series experiments (Table I and Fig. 9) indicates that by changing from the standard-NSFLS to the Cen-NSFLS, the MSE has improved in all four NSFLS experiments. The highest improvement (17.31%) was found

TABLE II
THE PREDICTION PERFORMANCE (FOR LORENZ TIME SERIES) PRODUCED BY THE DIFFERENT FLSs BASED ON AVERAGE MSEs (MEAN SQUARE ERRORS).

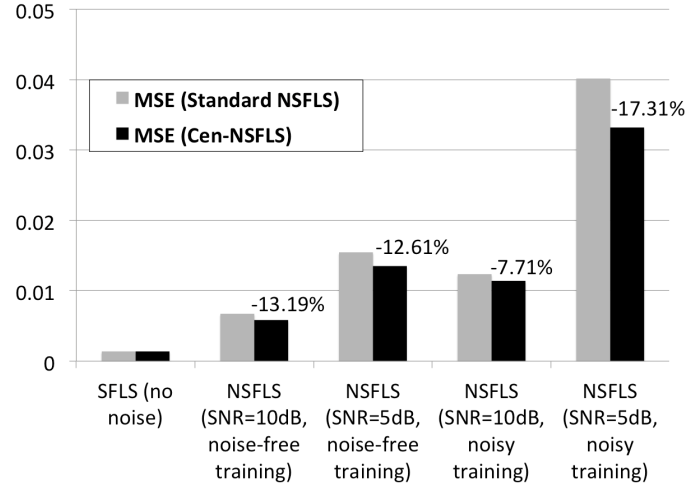| System	SNR (dB)	Training	MSE (Standard NSFLS)	MSE (Cen-NSFLS)	Change (%)
SFLS	n/a	n/a	1.18169	1.18169	0.00
NSFLS	10	noise-free	7.71616	7.47650	-3.11
NSFLS	5	noise-free	12.41810	11.88196	-4.32
NSFLS	10	noisy	11.29301	10.82722	-4.12
NSFLS	5	noisy	23.29384	20.68090	-11.22

Fig. 9. Comparing the improvement of prediction errors (MSEs) produced by the standard NSFLS and the Cen-NSFLS (for M-G time series). The MSE improvements are shown in percentage.

for noisy training with higher levels of noise (SNR=5dB) and the lowest improvement (7.71%) has was achieved in noisy training but under lower noise levels (SNR=10dB). It is also shown that when the system is trained with noise-free data, the MSE improvement is similar in different noise levels compared to when the system is trained with noisy data.

In order to examine the source of the observed MSE improvements in the described time-series prediction FLSs, it is helpful to investigate an individual output calculation and compare the results between both composition methods. We consider one of the NSFLSs (in the explained M-G experiment) where SNR=5dB and where noise-free training is used. Also, we focus on one of the data samples, namely at t=1000. In this case, the outputs of the two systems (the standard and the Cen-NSFLS) are 1.150 and 1.180 respectively, whereas the expected value is 1.280. The two predicted values are results of a centroid defuzzification in an NSFLS based on 9 previous samples and a set of 184 fuzzy rules.

For t=1000, the output FSs of the two NSFLSs are illustrated in Fig. 10. Intuitively, we expect that the MF of the output FS generated by the Cen-NSFLS at each point is less than its counterpart in the standard NSFLS, because for each intersection between an input and an antecedent FS, the firing strength of x_{cen} is equal or less than the firing strength of x_{max} (see Fig. 4). Beyond this, the centroids of the output FSs are calculated based on the overall shape of the output
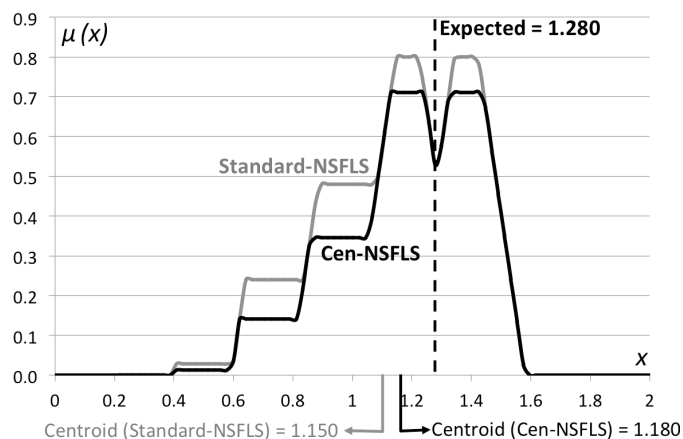
Fig. 10. Output FSs for t=1000 in a sample SNR=5dB and noise-free trained NSFLS. The centroid defuzzification results for both cases, together with the expected value are shown.

TABLE III
THE NUMBER OF SAMPLES (OUT OF 300) WHERE EACH METHOD IS
OUTPERFORMED IN DIFFERENT NSFLS CONFIGURATIONS

NSFLS Configuration	Outperformed cen-min predictions	Outperformed max-min predictions
SNR=10dB noise-free training	216	84
SNR=5dB noise-free training	214	86
SNR=10dB noisy training	204	96
SNR=5dB noisy training	216	84

MF, not only on the individual membership grades. In the example of Fig. 10, the change in the shape of the MF has shifted the graph's centroid to the right and has made it closer to the expected value.

This particular set of results shows a better estimation of the actual value by the Cen-NSFLS compared to the standard NSFLS. However, it is important to note that for other samples, this may not be the case. On a sample-by-sample analysis (from t=701 to t=1000), the Cen-NSFLS "outperformed" in 216 out of 300 samples. This shows that in most of the cases, the change in the shape of the output MF has made the centroid point closer to the expected value. In SNR=10dB and when training method is done using noisy data, the result of the sample-by-sample analysis is almost the same. Table III shows the number of times that each of two NSFLSs outperformed the other over the 300 samples for the different NSFLS configurations. Table III show a similar MSE improvement pattern to Table I. For example, the smallest cen-NSFLS outperformance and the minimum MSE improvement are both in the case of the higher SNR noisy-trained system.

The described outperformance of the Cen-NSFLS means that in the studied time-series prediction systems, applying the new method provides more accurate results especially when the data samples (and thus our inputs) are more noisy. This indicates that the original motivation - to better track uncertainty captured in the input fuzzy sets - was achieved
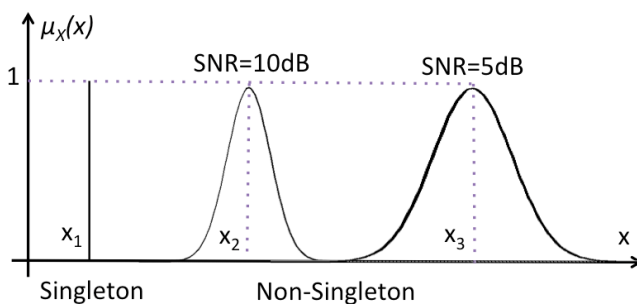

Fig. 11. Three different uncertainty models used in the experiments. x_1, x_2 and x_3 represent three actual inputs to the system where their uncertainties are modelled by three different fuzzification types.

by the new composition method for NSFLS. Clearly, this conclusion is so far based on observing a particular set of NSFLSs in a specific setting for two types of time series. Further work will be required to more generally evaluate the proposed approach. With this in mind, the scalability of the results to a wider range of NSFLS applications (particularly in time-series prediction) is a direction for our future work.

Finally, in order to recapitulate and clarify our aims and approach, it is noticeable that we did not change the method through which the uncertainties are captured or modelled in NSFLSs. Instead, we proposed to change the method in which the uncertainty models are *used* for conducting fuzzy inference in such systems. Fig. 11 shows the different types of input FSs used to model input uncertainties in the experiments (singleton and non-singleton with different input noise levels). The experiments show that changing the *composition method* resulted in better performance, i.e. "better use was made" of the same uncertainty capturing models.

V. COMPUTATIONAL COMPLEXITY

It is intuitive to expect higher computational complexity for the cen-min method over the max-min method. In this Section, first, their respective computational complexity is compared. Secondly, as complexity in both approaches is directly related to the level of discretization performed, we proceed with an analysis of the effects of selecting different discretization levels on the quality of the results of both approaches. This analysis is provided in a wider context than just NSFLSs.

A. Complexity

Arguably, the computational complexity of calculating the centroid may be a drawback for the new composition method. For an initial real world comparison, software (using Juzzy library [43]) was developed to compare the time needed to calculate x_{max} and x_{cen} based on the introduced NSFLSs, with the MF of the input and antecedent FSs being Gaussian. Testing is done based on the standard and the Cen-NSFLS experiment described in the previous section where input is noisy (SNR=10dB) and training is noise-free. Using a regular PC and in average over 300 samples (from t=701 to t=1000), the calculation of a single output out of 9 sample inputs using 184 rules took 37.19ms in the standard NSFLS and 36.84ms

in the Cen-NSFLS. The test shows no significant difference in the necessary time for a controlled iteration between the two methods.

To explain this, it is important to recapitulate the discrete nature of the FSs employed and the underlying computation for either approach. Specifically, while the max-min method relies on identifying the maximum of the intersection of two discrete MFs, the cen-min method seeks to identify its centroid. In fact, both methods rely on traversing the discrete MFs, while performing basic computations as illustrated by the sample Java code listings for both approaches.

The following listing shows the computation of x_{max} in max-min method:

```
peak=0; xmax=1;
for (i=1; i<=disc; i++) {
  if (u(i)>peak) {
    peak=u(i);
    xmax=x(i); } }
return xmax;
```

and the following listing computes x_{cen} in cen-min method:

```
numerator=0; denominator=0;
for (i=1; i>=disc; i++) {
  numerator+=x(i)*u(i);
  denominator+=u(i); }
return numerator/denominator;
```

In the above listings, $disc$ is the discretization level and $u(i)$ is the discrete degree of membership of the intersection between the input and antecedent set. In the second listing, the calculation is based on the discrete version of x_{cen} formulated in (11).

Our explanation to the observed similarity of computation times between the two methods is that firstly as already noted, both listings include a similar loop over $disc$ sampled points. Secondly, while the cen-min approach employs multiplication (i.e. more complex than the Boolean check employed in the max-min approach), both methods are computationally so simple that on modern computer architectures, the effect is negligible for common discretization levels. We discuss the latter further in the following subsection.

B. The effect of Discretization Levels

Since the calculation of the fuzzy set composition is done for discrete MFs, selecting an optimum discretization level is a trade-off between accuracy and computation complexity. Thus, it is interesting to consider comparing the performance of the cen-min and max-min composition methods by changing the discretization levels.

We illustrate this comparison in concept for a simple example, highlighting the difference between the real value (in the continuous case) and the calculated value (in the real world discrete case) of x_{max} and x_{cen} in Fig. 12. The example shows the intersection of two triangular MFs which is captured by 10 discretization steps (for easy visualization). It highlights that in this particular case, the real and actually calculated values for x_{cen} are much closer than those for x_{max}.

For further illustration, using the same example as in Fig. 12, we change the discretization levels varied from 1 to 100, capturing the computed x_{max} and x_{cen}. The results of this are

shown in Fig. 13, indicating how for increasing discretization levels, x_{cen} approaches its steady (and accurate) value much earlier than x_{max}. This is intuitive and explained by considering that when finding x_{max}, the error is directly based on the relative position of the single maximum between the two sampled points around it whereas in x_{cen}, the relative positions of all the sampled points are involved in the calculation (as in (10)).

While the above observations is clearly specific to this particular example, it suggests that in general the computation of the cen-min method is more stable and less error-prone compared to max-min method when the discretization levels are relatively low. A more formal evaluation of this effect is beyond the scope of the current paper and will be addressed in the future.

VI. CONCLUSIONS AND FUTURE WORK

In this paper we proposed a novel method of fuzzy relations composition in order to enable a higher fidelity capture of the input uncertainty embodied by input fuzzy sets in NSFLSs in comparison to traditional NSFLSs. In the novel approach, the *sup* (supremum - commonly maximum) operation is replaced by the *cen* (centroid) operation. The resulting cen-min method thus replaces the maximum of the intersection of input and antecedent FS by the degree of membership of the intersection's centroid. This operation visually better captures the detail of the input FS's MF during inference and thus should provide the potential for the more fine-grained modelling of the input uncertainty using the MF of the input FSs.

In order to support this intuition, we applied the new approach for NSFLS designed to two common time series prediction problem (M-G and Lorenz) and compared the results of the both the novel and traditional NSFLS formulations for (a) different noise levels on SNR=10dB and SNR=5dB, and (b) different training methods with noise-free and noisy data. The results show that the mean-squared error (MSE) between the NSFLS predictions and the actual time series values is
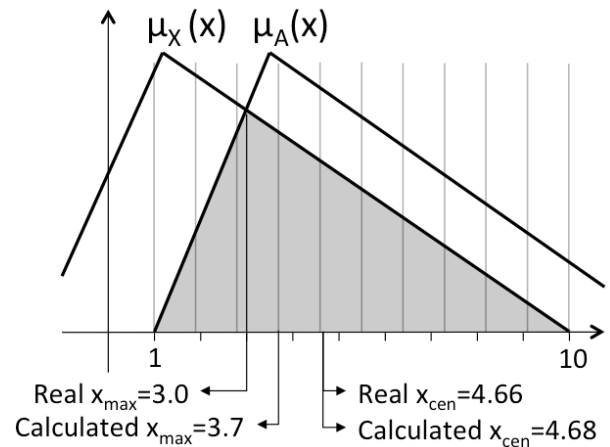

Fig. 12. A sample case where the intersection of the two FSs is a triangular area with $x_{left} = 1$, $x_{max} = 3$ and $x_{right} = 10$. The sample discretization level is 10, so the calculated x_{max} has a considerable difference (0.7) compared to the actual x_{max}, whereas the difference is much lower between the calculated x_{cen} and the actual x_{cen}.
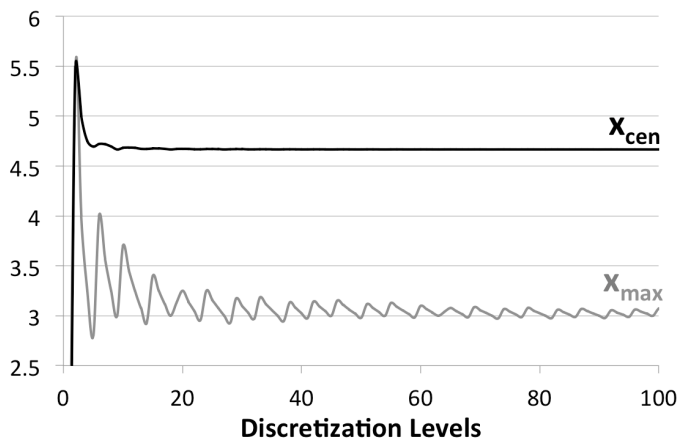
Fig. 13. The effect of selecting the discretization level on the calculated x_{max} and x_{cen} in the example shown in Fig. 12

reduced for the new approach by about 7 to 17 percent in the M-G time series, and by about 3 to 11 percent in the Lorenz time series - for different noise/training conditions. Initial exploration of the computational complexity of the proposed approach has shown that while formally more complex, in real world application there is no significant difference between the novel and traditional NSFLS approaches.

In summary, FLSs excel and are famous for their capability of uncertainty handling. However, the actual capacity of FLSs to capture and model uncertainty in different aspects (e.g., input and linguistic uncertainty) in applications is not well understood and provides significant further scope for research. This paper specifically aims to develop the capacity of NSFLSs to leverage accurate models of input uncertainty to provide better overall uncertainty capture and thus, better FLS performance. In future, the improved uncertainty capture is sought to also support better uncertainty representation at an output stage (i.e. beyond a crisp output, akin to what is currently achieved by the centroid interval for interval type-2 FLSs).

Beyond the work presented here, there are a number of interesting avenues for future work. These include (a) trying other alternative methods (e.g. fuzzy similarity and distance [44]) in NSFLSs for comparing input and antecedent FSs beyond focusing on their intersections; (b) trying other use cases in time-series or other applications when input data is noisy; (c) studying the comparison for other types of MFs including non-Gaussian, non-convex and dynamic/adaptive types of input FSs (as in [17], [36], [37] and [38]); and (d) trying the new method for non-singleton type-2 systems and type-2 non-singleton type-2 systems (i.e. where the input FS is also of type-2).

ACKNOWLEDGMENT

REFERENCES

[1] G. Bojadziev and M. Bojadziev, *Fuzzy logic for business, finance, and management*. World Scientific Publishing Co., Inc., 2007.

[2] L. Duckstein *et al.*, *Fuzzy rule-based modeling with applications to geophysical, biological, and engineering systems*. CRC press, 1995, vol. 8.

[3] J. Yen, R. Langari, and L. A. Zadeh, *Industrial applications of fuzzy logic and intelligent systems*. IEEE Press, 1995.

[4] L. A. Zadeh, "Fuzzy sets," *Information and control*, vol. 8, no. 3, pp. 338–353, 1965.

[5] L. A. Zadeh, "Fuzzy logic = computing with words," *IEEE Trans. Fuzzy Syst.*, vol. 4, no. 2, pp. 103–111, 1996.

[6] G. C. Mouzouris and J. M. Mendel, "Nonsingleton fuzzy logic systems: theory and application," *IEEE Trans. Fuzzy Syst.*, vol. 5, no. 1, pp. 56–71, 1997.

[7] J. M. Mendel, *Uncertain rule-based fuzzy logic system: introduction and new directions*. Prentice-Hall, 2001.

[8] C. Wagner and H. Hagras, "Novel methods for the design of general type-2 fuzzy sets based on device characteristics and linguistic labels surveys," in *Proceedings of the Joint 2009 International Fuzzy Systems Association World Congress and 2009 European Society of Fuzzy Logic and Technology Conference*. European Society for Fuzzy Logic and Technology, 2009, pp. 537–543.

[9] A. B. Cara, I. Rojas, H. Pomares, C. Wagner, and H. Hagras, "On comparing non-singleton type-1 and singleton type-2 fuzzy controllers for a nonlinear servo system," in *IEEE Symposium on Advances in Type-2 Fuzzy Logic Systems*. IEEE, 2011, pp. 126–133.

[10] G. C. Mouzouris and J. M. Mendel, "Non-singleton fuzzy logic systems," in *Fuzzy Systems, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the Third IEEE Conference on*. IEEE, 1994, pp. 456–461.

[11] N. Sahab and H. Hagras, "A type-2 nonsingleton type-2 fuzzy logic system to handle linguistic and numerical uncertainties in real world environments," in *IEEE Symposium on Advances in Type-2 Fuzzy Logic Systems (T2FUZZ)*. IEEE, 2011, pp. 110–117.

[12] A. Pourabdollah, C. Wagner, M. Smith, and K. Wallace, "Real-world utility of non-singleton fuzzy logic systems: A case of environmental management," in *Fuzzy Systems (FUZZ-IEEE), 2015 IEEE International Conference on*. IEEE, 2015.

[13] L. Wang, *A course in fuzzy systems*. Prentice-Hall press, USA, 1999.

[14] E. H. Mamdani and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," *International journal of man-machine studies*, vol. 7, no. 1, pp. 1–13, 1975.

[15] J. M. Mendel, "Uncertainty, fuzzy logic, and signal processing," *Signal Processing*, vol. 80, no. 6, pp. 913–933, 2000.

[16] G. C. Mouzouris and J. M. Mendel, "Nonlinear time-series analysis with non-singleton fuzzy logic systems," in *Proceedings of the IEEE/IAFE Computational Intelligence for Financial Engineering*. IEEE, 1995, pp. 47–56.

[17] G. C. Mouzouris and J. Mendel, "Dynamic non-singleton fuzzy logic systems for nonlinear modeling," *IEEE Trans. Fuzzy Syst.*, vol. 5, no. 2, pp. 199–208, 1997.

[18] D. Kim, S.-H. Huh, and G.-T. Park, "Modeling corrupted time series data via nonsingleton fuzzy logic system," in *Neural Information Processing*. Springer, 2004, pp. 1298–1303.

[19] M. C. Mackey, L. Glass *et al.*, "Oscillation and chaos in physiological control systems," *Science*, vol. 197, no. 4300, pp. 287–289, 1977.

[20] A. Lapedes and R. Farber, "Nonlinear Signal Processing Using Neural Networks," Los Alamos National Laboratory, Los Alamos, NM, Tech. Rep. LA-UR-87-2662, 1987.

[21] R. S. Crowder, "Predicting the mackey-glass time series with cascade-correlation learning," in *Proceedings of Connectionist Models Summer School*. Carnegie Mellon Univ, 1990, pp. 117–123.

[22] J. Jang, "Anfis: adaptive-network-based fuzzy inference system," *IEEE Trans. Syst., Man, Cybern.*, vol. 23, no. 3, pp. 665–685, 1993.

[23] R. D. Jones, Y. Lee, C. Barnes, G. Flake, K. Lee, P. Lewis, and S. Qian, "Function approximation and time series prediction with neural networks," in *IJCNN International Joint Conference on Neural Networks*. IEEE, 1990, pp. 649–665.

[24] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning," *Information sciences*, vol. 8, no. 3, pp. 199–249, 1975.

[25] J. H. Aladi, C. Wagner, and J. M. Garibaldi, "Type-1 or interval type-2 fuzzy logic systemson the relationship of the amount of uncertainty and fou size," in *Fuzzy Systems (FUZZ-IEEE), 2014 IEEE International Conference on*. IEEE, 2014, pp. 2360–2367.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TFUZZ.2016.2540065, IEEE Transactions on Fuzzy Systems

12

[26] J. H. Aladi, C. Wagner, J. M. Garibaldi, and A. Pourabdollah, "On transitioning from type-1 to interval type-2 fuzzy logic systems," in *Fuzzy Systems (FUZZ-IEEE), 2015 IEEE International Conference on*. IEEE, 2015.

[27] P. Melin, C. Gonzalez, J. R. Castro, O. Mendoza, O. Castillo *et al.*, "Edge-detection method for image processing based on generalized type-2 fuzzy logic," *IEEE Trans. Fuzzy Syst.*, vol. 22, no. 6, pp. 1515–1525, 2014.

[28] G. M. Mendez, O. Castillo, R. Colas, and H. Moreno, "Finishing mill strip gage setup and control by interval type-1 non-singleton type-2 fuzzy logic systems," *Applied Soft Computing*, vol. 24, pp. 900–911, 2014.

[29] M. de los Angeles Hernandez, P. Melin, G. M. Mendez, O. Castillo, and I. Lopez-Juarez, "A hybrid learning method composed by the orthogonal least-squares and the back-propagation learning algorithms for interval a2-c1 type-1 non-singleton type-2 tsk fuzzy logic systems," *Soft Computing*, vol. 19, no. 3, pp. 661–678, 2015.

[30] M. A. Sanchez, O. Castillo, and J. R. Castro, "Generalized type-2 fuzzy systems for controlling a mobile robot and a performance comparison with interval type-2 and type-1 fuzzy systems," *Expert Systems with Applications*, vol. 42, no. 14, pp. 5904–5914, 2015.

[31] C. Wagner and H. Hagras, "Toward general type-2 fuzzy logic systems based on zslices," *Fuzzy Systems, IEEE Transactions on*, vol. 18, no. 4, pp. 637–660, Aug 2010.

[32] O. A. Malik and S. A. Senanayake, "An interval type-2 fuzzy logic based classification model for testing single-leg balance performance of athletes after knee surgery," in *Proceedings of the 10th International Symposium on Computer Science in Sports (ISCSS)*. Springer, 2016, pp. 85–92.

[33] H. Hassani and J. Zarei, "Interval type-2 fuzzy logic controller design for the speed control of dc motors," *Systems Science & Control Engineering: An Open Access Journal*, vol. 3, no. 1, pp. 266–273, 2015.

[34] O. Castillo, J. R. Castro, P. Melin, and A. Rodriguez-Diaz, "Application of interval type-2 fuzzy neural networks in non-linear identification and time series prediction," *Soft Computing*, vol. 18, no. 6, pp. 1213–1224, 2014.

[35] R. Martínez-Soto, O. Castillo, and J. R. Castro, "Genetic algorithm optimization for type-2 non-singleton fuzzy logic controllers," in *Recent Advances on Hybrid Approaches for Designing Intelligent Systems*. Springer, 2014, pp. 3–18.

[36] N. Sahab and H. Hagras, "Adaptive non-singleton type-2 fuzzy logic systems: A way forward for handling numerical uncertainties in real world applications." *International Journal of Computers, Communications & Control*, vol. 6, no. 3, 2011.

[37] N. Sahaab and H. Hagras, "An adaptive type-2 input based nonsingleton type-2 fuzzy logic system for real world applications," in *IEEE International Conference on Fuzzy Systems*. IEEE, 2011, pp. 509–516.

[38] N. Sahab and H. Hagras, "Towards comparing adaptive type-2 input based non-singleton type-2 FLS and non-singleton FLSs employing gaussian inputs," in *IEEE International Conference on Fuzzy Systems*. IEEE, 2012, pp. 1–8.

[39] A. B. Cara, C. Wagner, H. Hagras, H. Pomares, and I. Rojas, "Multiobjective optimization and comparison of nonsingleton type-1 and singleton interval type-2 fuzzy logic systems," *IEEE Trans. Fuzzy Syst.*, vol. 21, no. 3, pp. 459–476, 2013.

[40] N. Sahab and H. Hagras, "A hybrid approach to modeling input variables in non-singleton type-2 fuzzy logic systems," in *UK Workshop on Computational Intelligence (UKCI)*. IEEE, 2010, pp. 1–6.

[41] L. Wang and J. M. Mendel, "Generating fuzzy rules by learning from examples," *IEEE Trans. Syst., Man, Cybern.*, vol. 22, no. 6, pp. 1414–1427, 1992.

[42] E. N. Lorenz, "Deterministic nonperiodic flow," *Journal of the atmospheric sciences*, vol. 20, no. 2, pp. 130–141, 1963.

[43] C. Wagner, "Juzzy-a java based toolkit for type-2 fuzzy logic," in *IEEE Symposium on Advances in Type-2 Fuzzy Logic Systems*. IEEE, 2013, pp. 45–52.

[44] J. McCulloch, C. Wagner, and U. Aickelin, "Analysing fuzzy sets through combining measures of similarity and distance," in *2014 IEEE International Conference on Fuzzy Systems*. IEEE, 2014, pp. 155–162.

Amir Pourabdollah is currently a research fellow in computer science with the Lab for Uncertainty in Data and Decision Making at the University of Nottingham, UK. He has received his Ph.D. in Computer Science (2009) and M.Sc. in Information Technology (2004) from the same university. He has also did his M.Eng. in Electrical Engineering from the University of Tehran, Iran (1993). Dr. Pourabdollah has research interests and publications in the theory of fuzzy logic, uncertain data management, linked data modelling, geospatial science and technologies.

Christian Wagner received the Ph.D. degree in computer science from the University of Essex, UK, in 2009. He is currently an Associate Professor of Computer Science and Director of the Lab for Uncertainty in Data and Decision Making at the School of Computer Science, University of Nottingham, UK. Dr. Wagner has published more than 60 peer-reviewed manuscripts and received the Outstanding Paper Award for the IEEE Transactions of Fuzzy Systems in 2013 (for paper published in 2010) and the Best Paper Award at the 2012 IEEE International Conference on Fuzzy Systems. He is co-chair of the IEEE CIS Task Force on Affective Computing and an associate editor of the IEEE Transactions on Fuzzy Systems.

Jabran Aladi received the B.Sc. degree in systems engineering from King Fhad University of Petroleum and Minerals (KFUPM), Dhahran, Saudi Arabia, in 1993, the M.Sc. degree in information and process systems engineering from Surrey University, Guildford, UK, in 2006 and the PhD degree in information technology from University of Nottingham, Nottingham, UK, in 2016. He is currently with the Intelligent Modelling and Analysis Group, School of Computer Science, University of Nottingham, Nottingham, UK. His research interest mainly focused on fuzzy logic theory and applications –type-2 fuzzy logic systems.

Jonathan M. Garibaldi received the B.Sc (Hons) degree in Physics from Bristol University, UK in 1984, and the M.Sc. degree in Intelligent Systems and the Ph.D. degree in Uncertainty Handling in Immediate Neonatal Assessment from the University of Plymouth, UK in 1990 and 1997, respectively. He is Professor of Computer Science, and leads the Intelligent Modelling and Analysis (IMA) Research Group, in the School of Computer Science at the University of Nottingham, UK. The IMA research group undertakes research into intelligent modelling, utilising data analysis and transformation techniques to enable deeper and clearer understanding of complex problems. His main research interests are modelling uncertainty and variation in human reasoning, and in modelling and interpreting complex data to enable better decision making, particularly in medical domains. He has made many theoretical and practical contributions in fuzzy sets and systems, and in a wide range of generic machine learning techniques in real-world applications. Prof. Garibaldi has published over 200 papers on fuzzy systems and intelligent data analysis, is an Associate Editor of Soft Computing, and three other journals. He has served regularly in the organising committees and programme committees of a range of leading international conferences and workshops, such as FUZZ-IEEE, WCCI, EURO and PPSN. He is a member of the IEEE.