



The University of
Nottingham

UNITED KINGDOM · CHINA · MALAYSIA

Pourabdollah, Amir and John, Robert and Garibaldi, Jonathan M. (2017) A new dynamic approach for non-singleton fuzzification in noisy time-series prediction. In: 2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), 9-12 July 2017, Naples, Italy.

Access from the University of Nottingham repository:

<http://eprints.nottingham.ac.uk/45209/1/submission.pdf>

Copyright and reuse:

The Nottingham ePrints service makes this work by researchers of the University of Nottingham available open access under the following conditions.

This article is made available under the University of Nottingham End User licence and may be reused according to the conditions of the licence. For more details see:
http://eprints.nottingham.ac.uk/end_user_agreement.pdf

A note on versions:

The version presented here may differ from the published version or from the version of record. If you wish to cite this item you are advised to consult the publisher's version. Please see the repository url above for details on accessing the published version and note that access may require a subscription.

For more information, please contact eprints@nottingham.ac.uk

A New Dynamic Approach for Non-Singleton Fuzzification in Noisy Time-Series Prediction

Amir Pourabdollah, Robert John and Jonathan M. Garibaldi

Lab for Uncertainty in Data and Decision Making (LUCID), School of Computer Science,
University of Nottingham, Nottingham, NG8 1BB, UK

Email: {amir.pourabdollah, robert.john, jon.garibaldi}@nottingham.ac.uk

Abstract—Non-singleton fuzzification is used to model uncertain (e.g. noisy) inputs within fuzzy logic systems. In the standard approach, assuming the fuzzification type is known, the observed [noisy] input is usually considered to be the core of the input fuzzy set, usually being the centre of its membership function. This paper proposes a new fuzzification method (not type), in which the core of an input fuzzy set is not necessarily located at the observed input, rather it is dynamically adjusted based on statistical methods. Using the weighted moving average, a few past samples are aggregated to roughly estimate where the input fuzzy set should be located. While the added complexity is not huge, applying this method to the well-known Mackey-Glass and Lorenz time-series prediction problems, show significant error reduction when the input is corrupted by different noise levels.

I. INTRODUCTION

Fuzzy logic systems (FLSs) have been widely used to tackle time-series prediction problems, particularly under noise and uncertainty conditions. The applied techniques are ranging from type-1 to type-2 FLSs, and from singleton to non-singleton FLS configurations (as shown, for example, in [1]–[7]). Non-singleton fuzzification is one of the known approaches to provide a capacity for noise/uncertainty handling in FLSs used for such purposes. Although the theory of Non-Singleton Fuzzy Logic Systems (NSFLSs) has been established for many years (e.g., in [5], [8]), and the capacity of NSFLSs to deliver superior performance in comparison to Singleton Fuzzy Logic Systems (SLFSs) has been shown repeatedly (e.g., in [8]–[10]), the research on NSFLSs is still in its early stages since their complexity has not helped to widen their usefulness among the community, compared to the singleton systems.

Briefly, in the singleton FLSs, the inputs to the system are singleton fuzzy sets, meaning that the membership grade of the input fuzzy set is 1 at a single value of input, and 0 otherwise. In other word, there is a full certainty about the value of the given input. In non-singleton systems, the uncertainty of the given input value is modelled in the fuzzifier, by converting the crisp input to a non-singleton fuzzy set, with an arbitrary membership function.

It is widely accepted in NSFLS design, that *conceptually, the non-singleton fuzzifier implies that the given input value is most likely to be the correct one from all of the values in its immediate neighborhood... the neighboring points are also likely to be the correct value, but to a lesser degree.* [8], [11], [12]. This is the basic idea behind the standard approach in

locating the input fuzzy set centred around the observed input. In reality however, a given observed sample may be corrupted by noise so it *may not be the most likely* to be the correct one. The question is when the original noise-free sample is not reachable, is there any better estimation for locating the input fuzzy set?

In this paper, a method is proposed to statistically aggregate the observed sample with a few previous samples (using weighted moving average) to have a better estimation of *the most likely* value to be the correct input, compared to the observed input. This can be an improvement to the standard noise modelling by non-singleton classification, with a potential to improve the performance of NSFLSs. The proposed method has been examined in this paper by a couple of chaotic time-series prediction problems (Mackey-Glass and Lorenz) under noisy conditions. The results are then compared with the standard methods found in the literature.

The rest of this paper is organized as follows: in section 2, the backgrounds on non-singleton fuzzification and its improvement methods for noisy time series prediction are briefly explained. Section 3 introduces the new proposed fuzzification method and in section 4 the time series experiments and their results are presented. The paper will finally be concluded in the last section.

II. BACKGROUND AND MOTIVATION

The fuzzification component in a general FLS architecture (Fig. 1) plays a central role in capturing and modelling the input uncertainty. Associating a fuzzy set (FS) to each crisp input value is what the fuzzification does towards modelling the uncertainty. An input FS around the observed input value x' is represented by $\mu(x)$ in Fig. 1. A wider or narrower input FS represents a more or less uncertainty level on the observed input value, respectively.

Determining the type of the $\mu()$ (e.g., Gaussian) is a system designers' decision based on an estimate of the kind and quantity of noise or uncertainty present [3]. The process of identifying and optimizing the shape of $\mu()$ is a different topic from this paper (see for example [13]–[16]). Since the uncorrupted input is intuitively not known, it is questionable which x might represent the maximum grade in the input FS. In the current literature (e.g. [13]), the common method is to use the observed input (x') such that its membership grade $\mu(x')$ is maximum (namely, 1). In other words, x' is

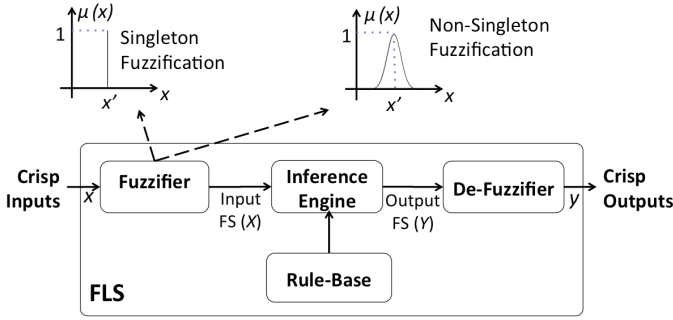


Fig. 1. The illustration of singleton vs. non-singleton FLS architecture

considered by default, to be the *core*¹ of the input FS. This *loosely* means that x' has the highest *chance* to be the real uncorrupted input². However this is not a perfect modelling of noisy input, as this may be misleading for the rest of the FLS, depending on how much a particular observed input has been deviated from its real value. Based on the random characteristic of noise, the fitness of such a model randomly varies among the different inputs.

For example, in Fig. 2, sample noisy and noise-free time series are shown. The input FS based on the observation and based on the real values are also shown, where a Gaussian fuzzification is applied. Since there can be random differences between the observed and real values, the fuzzification based on the observed values can be largely deviated from the ideal (but not observable) FS. The motivation of this research is improving the performance of such NSFLS for noisy time series prediction, by dynamically tuning the fuzzification set centre. This can be achieved by analysing the previously observed samples in addition to the current one.

Looking at the literature, improving the time-series prediction performance of NSFLSs in noisy conditions has been focused in some research works from different perspectives. In two previous articles, it is shown that the performance of such systems can be improved by changing in the inference engine. While the standard method of non-singleton inference in Mamdani's method [18], as described in [13], is the compositional rule of inference i.e. the sup-star (e.g. max-min) operation between the input and the antecedent MFs, it is suggested in [19] to use the centroid operator, and in [20] to use the MF's quantified similarities instead of the standard composition in the inference engine of NSFLSs used for noisy time-series prediction.

Some other research works have focused on the fuzzifier component in order to improve the NSFLS performance in noisy time-series prediction. Particularly there are a few research works on using varieties of moving average method within the development of predictive models by fuzzy logic

¹By definition, the core or the kernel of a fuzzy set is the set of all values where their membership grade is 1, i.e. the 1-cut of the fuzzy set [17].

²We used the term "loosely" together with the word *chance* in order to avoid taking the problem into the probability context. However, the non-singleton fuzzification of a noisy input is linked anyway to the noise properties - including its statistical characteristics - which shows that there is an overlap between the fuzzy set theory and the probability theory when it comes to non-singleton fuzzification.

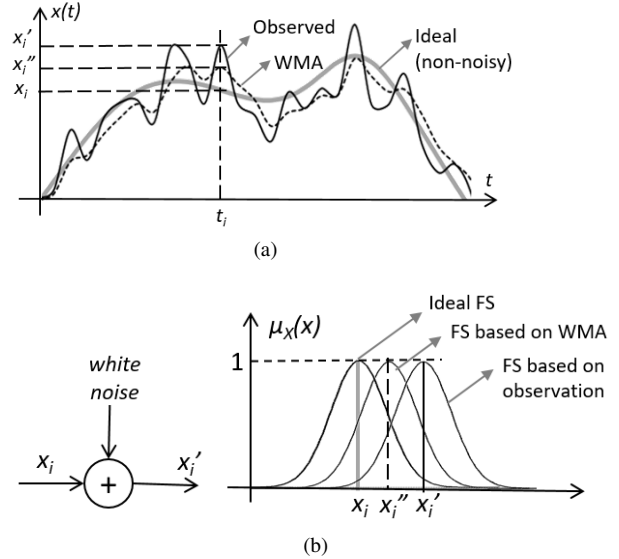


Fig. 2. (a) Noisy time series together with the non-noisy values and an associated WMA curve. In an extreme case, at time t_i , a real input x_i has been observed as x'_i while the WMA shows x''_i . (b) By Gaussian fuzzification, the core of the FS is set to the observed input x'_i (right-side FS). An ideal FS model would be the left-side one in which the core is x_i (left-side FS). As it is shown, in this extreme case, the values more than x'_i are very unlikely to be observed while the right-side FS does not reflect this. In contrast, the values around x_i which should have the highest chance to be observed are graded nearly zero in the right-side FS model. The FS based on WMA is closer to be realistic.

systems, such as in [3] and [21], where the system parameters of NSFLS are dynamically trained for better performance using non-linear autoregressive moving average (NARMA) through feedback loops. In [22] also, the input noise has been handled by tuning both non-singleton fuzzifier and type-2 inference engine in order to improve the performance.

In all of the above attempts, the observed input has been used to determine the core of the input fuzzy set. To the best of our knowledge, no analytical method is used to dynamically determine the core of the non-singleton fuzzification.

III. DYNAMIC FUZZIFICATION BASED ON WMA

The described standard fuzzification approach assumes the observed noisy input to be the core of the input fuzzy set, however, this may not be the only option. Having the FS type already identified (e.g. Gaussian), NSFLSs' performance can be tuned by adjusting the core of the input FS (e.g. its mean), such that the FS can model the input uncertainty as close as possible. The proposed approach is to use a nominated statistical method to estimate the *expected value* of x' according to a set of past inputs. Moving Average (MA) is a simple solution, of which Weighted Moving Average (WMA) provides a more flexible formulation.

WMA has many applications particularly in financial and engineering domains, when it is needed to smooth the curve of a variable change and get a better trend identification. We prefer WMA over simple MA (SMA) for the purpose of this paper, because we would like to assign a higher weight to the last observed input than the past ones. Moreover, we would like to reduce the lag that naturally occurs between

the samples and its moving average curve if SMA is used. Other MA methods e.g. Cumulative MA (CMA), Smoothed MA (SSMA), or more advanced estimation techniques such as Auto-Regression MA (ARMA) and its non-linear version (NARMA) are also possible options, however in this paper we focus on a simple approach (WMA). Notably, the more advanced estimation method is used, the more computational complexity overload is to be considered.

For the i th sample x_i , WMA is defined here as:

$$WMA_d(x_i) = \frac{\sum_{j=i-d+1}^i (x_j w_j)}{\sum_{j=i-d+1}^i (w_j)} ; i = d \dots N; \quad (1)$$

where d is the depth of WMA, w is an array of weights, and N is the number of samples. For $i < d$, WMA is not calculated.

In the same example of Fig. 2(a), the WMA is calculated according to the current and two past input values (here the weight array values are arbitrarily set to 3 for the x'_i , 2 for x'_{i-1} and 1 for x'_{i-2}). For the special case of observed x'_i at time t_i , as seen in Fig. 2(b), the WMA-based FS is closer to the ideal FS than the observation-based FS, leading to a more realistic modelling. Also in general case, Fig. 2(a) shows that the WMA is more closely following the real input over time. This means that the fuzzification based on dynamic WMA values are likely to provide more accurate information about the input uncertainty, than the fuzzification based on the static observations.

The proposed method will be examined in the next section.

IV. EXPERIMENTS AND RESULTS

In this section we will replace the standard NSFLS fuzzification with a simple WMA-based fuzzification and will test it by two known prediction problems, namely Mackey-Glass (M-G) and Lorenz time series under noisy conditions. The results will be compared with the standard NSFLS. Besides the mathematics of the two time series, the rest of the two experiments will follow a same method described below.

A. Methodology

The method described in [5], 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 will compare the results between the two types of NSFLSs. The noise is considered to be Gaussian noise in two signal-to-noise ratios (SNRs) of 10dB and 5dB. Regarding the rule-base generation, we follow the method given in [23], an established approach to learning rules from existing input-output pairs.

Using (5), $x(t)$ 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. This means that the rule-base training stage is conducted in the same noisy condition as the testing. The last 300 points from $t = 701$ to $t = 1000$ are used for testing the system (see Fig. 3). Rules are trained according to the one-pass method described in [23]. 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.

Two NSFLSs are designed that use the same input samples to predict the output. They also use the same fuzzification type, namely a Gaussian membership function. The standard deviation of the Gaussian membership function is derived from the system's SNR value. If the standard deviation of the signal and the standard deviation of the noise in a system are σ_s and σ_n respectively, the system's SNR can be written as:

$$SNR = 10 \log \left(\frac{\sigma_s^2}{\sigma_n^2} \right) \quad (2)$$

Thus the standard deviation of the noise is calculated using:

$$\sigma_n = \frac{SNR}{10(\sigma_s/20)} \quad (3)$$

Both FLSs use Mamdani's inference method, centroid defuzzification, and the standard max-min for the composition of fuzzy relations. The first FLS uses the standard fuzzification, so that the FLS's core is located at the value of the observed input. The second FLS uses the WMA-based fuzzification, which means that the core of the input FS is located at the WMA of the past few inputs, depending on the WMA's depth and weights. We set the WMA's depth to 3 and the WMA's weights to (3, 2, 1) for the last three observed inputs. In other words, the FS's core for the i th input is:

$$x''_i = \frac{3x'_i + 2x'_{i-1} + x'_{i-2}}{6} ; i = 701 \dots 1000 \quad (4)$$

The above selections of depth and weights are not optimised, rather they are simply set for the proof of concept. A more advanced analysis is required to optimize them which is out of the scope of this paper and is considered as a future direction. Fig. 2 shows a part of the testing sample set together with the calculated WMA (for the case of SNR=10dB). It is also noticeable that the NSFLS capable of handling the above calculation needs a memory for keeping the last two observed inputs, which may seem to add to the system's complexity. However, we might notice that an even larger memory has been already required for the rule-training stage.

The prediction performance is evaluated based on the MSE (Mean Square Error) indicating the deviation of the prediction from the expected value. For calculating the performance error (MSE), the squared deviation of each predicted value from the expected value according to (5) is calculated and averaged. The less the averaged MSE, the closer the prediction.

It has been already shown (e.g. in [5]) that the NSFLS outperforms its counterpart FLS for noisy time-series prediction. In this paper, the standard NSFLS has been set as the reference system. As such, four possible settings for the noisy time series prediction are examined here: Standard/WMA-based NSFLS, fed by 5/10dB SNR noisy inputs.

B. Results of Mackey-Glass Time Series Prediction

Among the different time-series prediction problems, predicting Mackey-Glass time series [24] has been a benchmark problem in the literature which has been considered by a number of researchers, e.g., in [25]–[28]. An advantage of using

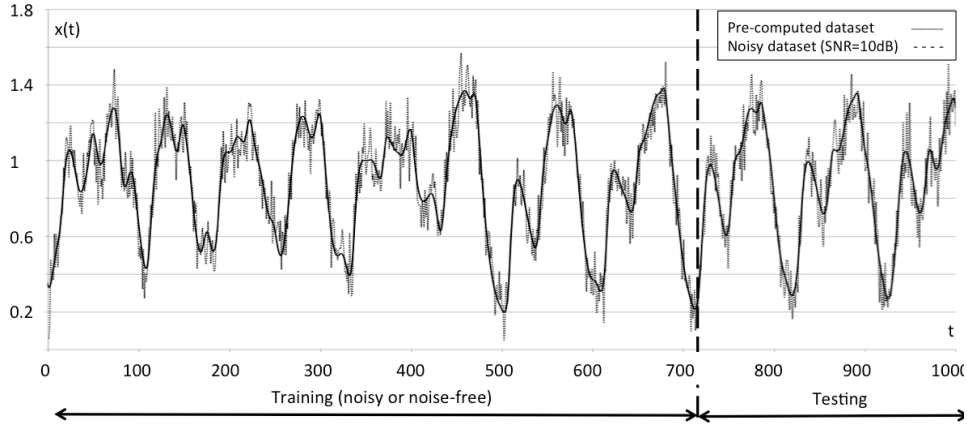


Fig. 3. The illustration of the pre-computed (noise-free) and noisy (Gaussian, SNR=10dB) time series. The system is trained from $t = 1$ to $t = 700$ and tested from $t = 701$ to $t = 1000$. The period $t = -999$ to $t = 0$ (not shown here) is for the initial transients to die out.

M-G time series is that although the time-series behaviour seems highly chaotic and unpredictable, the expected value can be pre-calculated using a closed differential equation, so that the performance of predicting each sample point can be accurately computed. M-G time series is characterised by a differential equation [24] as:

$$\frac{dx(t)}{dx} = \frac{0.2x(t-\tau)}{1+x^{10}(t-\tau)} - 0.1x(t) \quad (5)$$

For $\tau > 17$, (5) demonstrates a chaotic behaviour. We have selected $\tau = 30$. dx is also set to 1.

The prediction results in the four settings are illustrated in Fig. 4. In terms of quantities, table I shows the averaged MSE in the described four experiment settings.

TABLE I
THE PREDICTION PERFORMANCE FACTORS (FOR M-G TIME SERIES)
PRODUCED BY THE TWO NFLSS.

System	SNR (dB)	MSE
Standard	10	0.01099
WMA-based	10	0.00818 (25.53% improvement)
Standard	5	0.02846
WMA-based	5	0.02550 (-10.41% improvement)

As shown in Fig. 4, using WMA has helped a closer prediction in both noise conditions for the M-G time series experiment. Table I shows that in all the four settings, WMA-based fuzzification has improved the examined two prediction performance factors. In the lower noise level (SNR=10dB), the effect of using WMA-based fuzzification is 25.53% less prediction error. In the higher noise level (SNR=5dB) also, the MSE has been still decreased but by a relatively lower percentage (10.41%).

The method of standard NSFLS used in this paper is similar to the few existing similar researches (e.g. [5], [13]) but the absolute results may not be directly compared, due to the significant effect of the experiment settings on the results. However in a similar setting used in [19] and [20], it is shown that the MSE of the standard NSFLS in 10dB noise has dropped by 28.15% and 7.71% when two alternative inference

methods are applied (namely Cen-Min and Similarity-based, respectively). The 10.41% improvement of MSE shown in this paper is in parallel with (and potentially can be added to) the above improvements since it does not touch the inference engine, rather changes the fuzzification method. A future research will be dedicated to explore the results of applying the changes to both fuzzification and inference methods.

C. Results of Lorenz Time Series Prediction

In addition to the M-G time series, we also analyse the NSFLSs' performances using another known time series common in hydrodynamics and meteorology, namely Lorenz time series [29]. We consider the time series associated to variable x of the three-dimensional Lorenz differential equations:

$$\dot{x} = \sigma(y - z); \quad \dot{y} = rx - y - xz; \quad \dot{z} = xy - bz \quad (6)$$

where the dots denote the next values to the three variables x , y , z in the time series. In fact, this is a three-dimensional time series, but just one of the dimensions (x) is used in this experiment. To demonstrate a chaotic behaviour, the attributes σ , b and r are respectively set to 10, 8/3 and 28, as suggested in [29].

Similar to the what was observed in the last sub-section, applying the new fuzzification method for Lorenz time series shows the outperformance of the new method, as shown in Fig. 5 and Table II. The new method has produced less prediction error for the both noise levels. It shows 7.24% less error in SNR=10dB and 13.19% less error in SNR=5dB compared to the standard method.

TABLE II
THE PREDICTION PERFORMANCE FACTORS (FOR LORENZ TIME SERIES)
PRODUCED BY THE TWO NFLSS.

System	SNR (dB)	MSE
Standard	10	11.05557
WMA-based	10	10.25474 (7.24% improvement)
Standard	5	24.55937
WMA-based	5	21.32056 (-13.19% improvement)

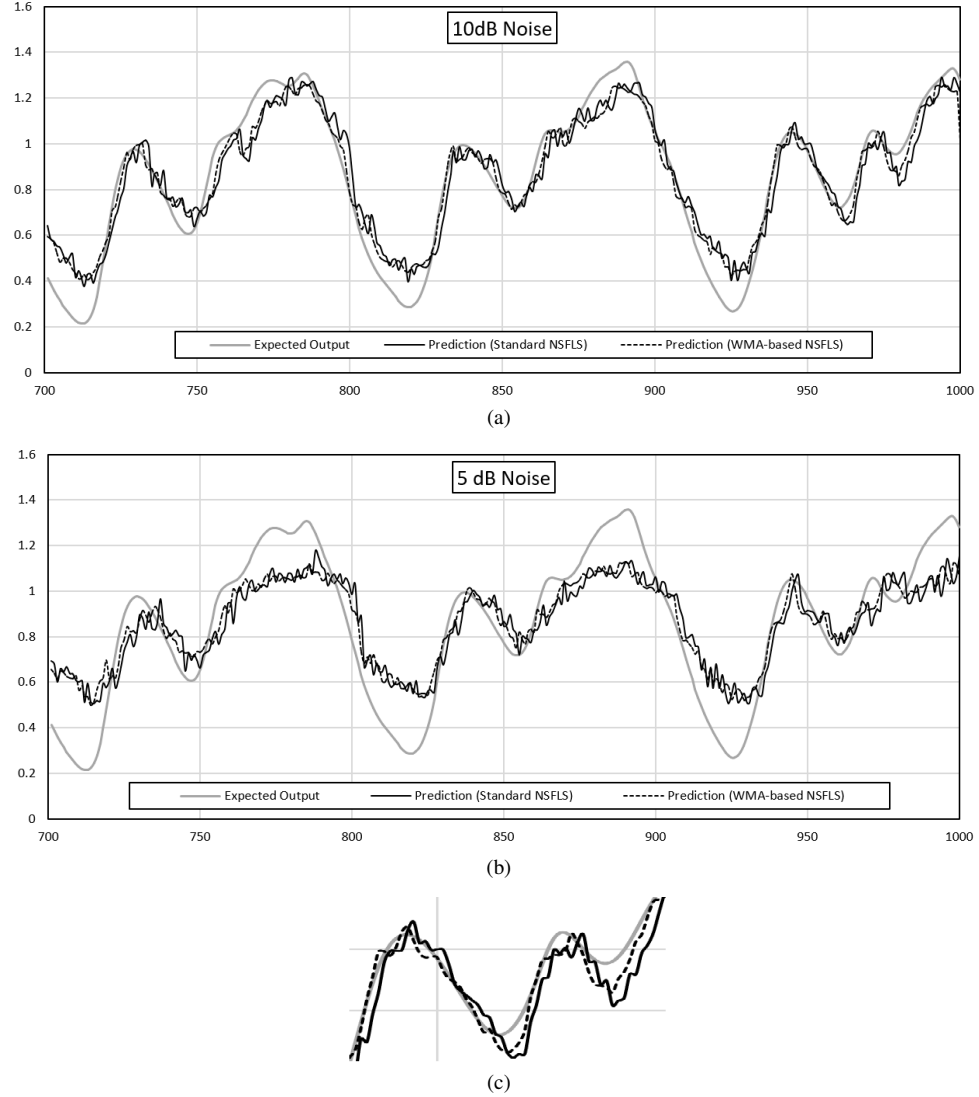


Fig. 4. (a) The prediction results of the two NSFLs together with the expected values in SNR=10dB (b) the same for SNR=5dB (c) In a zoomed part of graph (a), where the dashed line shows a lower deviation from the expected values compared to the solid line.

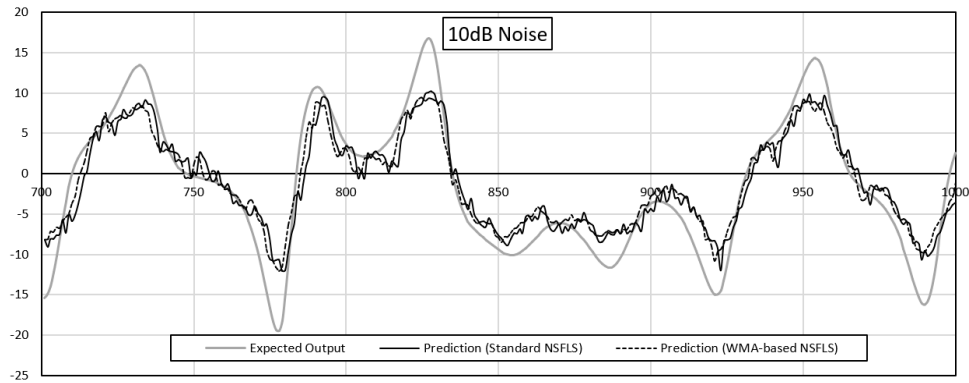


Fig. 5. The prediction results of the two NSFLs together with the expected values in SNR=10dB for the sample Lorenz time series.

V. CONCLUSIONS AND FUTURE WORK

In this paper, we attempted to improve the NSFLSs' performance by challenged an idea that states "the given [noisy] input value is most likely to be the correct one". The core of a fuzzified input (the fuzzy variable where the membership function is maximum), is not necessarily the observed input in a given time. Instead, an aggregation of the current input *and* a number of past inputs can be chosen as the core, in order to increase the chance of having a closer estimation of to the real input. It is intuitively sensible to assign a higher weight to the observed input, than its immediate neighbours, than the farther ones. Thus the suggested aggregation method in this paper is, but not limited to, the weighted moving average (WMA). This idea is followed by designing a fuzzifier that sets the core of its fuzzy set dynamically on the calculated WMA, regardless of the arbitrarily choice of the membership function type.

Two experiments set for applying the new method to a couple of chaotic and noisy time series prediction problems Mackey-Glass and Lorenz. The performances of NSFLSs in predicting the noisy time-series are examined for the standard fuzzification and the new methods, by measuring the produced error (MSE). The results in all settings are significantly improved. In general, the both experiments collectively provide evidences that applying the new method can potentially optimize the NSFLSs' design. However, not all of the parameters in both experiments are carefully selected or optimized. Thus more analysis is necessary to find the exact effect of those parameters on the final results. Moreover, it will be useful to know which statistical aggregation (not limited to WMA) may fit to which SNR or other input parameters/patterns.

There are different directions for the future works. Many of the methods, settings and parameters are needed to be more carefully selected. For example, the statistical aggregation can be chosen from a wide range of available methods. For the WMA parameters, the depth and weights could be tuned to fit a specific time series under a specific noise condition. Particularly, the choice of WMA weights in this paper was simplistic, in order to proof the concept only. More statistical analysis of the results can also help determining the selection criteria for those parameters, which consequently help optimizing the fuzzifier design for a given purpose.

ACKNOWLEDGMENT

This research is funded by the UK MRC MR/N005953/1 grant.

REFERENCES

- [1] D. Kim, S.-H. Huh, and G.-T. Park, "Modeling corrupted time series data via non-singleton fuzzy logic system," in *Neural Information Processing*, Springer, 2004, pp. 1298–1303.
- [2] 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.
- [3] 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.
- [4] 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.
- [5] 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.
- [6] J. M. Mendel, "Uncertainty, fuzzy logic, and signal processing," *Signal Processing*, vol. 80, no. 6, pp. 913–933, 2000.
- [7] A. Muhammad and G. King, "Foreign exchange market forecasting using evolutionary fuzzy networks," in *Computational Intelligence for Financial Engineering (CIFER)*. IEEE, 1997, pp. 213–219.
- [8] G. C. Mouzouris and J. M. Mendel, "Non-singleton fuzzy logic systems," in *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. IEEE, 1994, pp. 456–461.
- [9] N. Sahab and H. Hagrass, "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.
- [10] T. W. Chua and W. W. Tan, "Ga optimisation of non-singleton fuzzy logic system for classification," in *IEEE Congress on Evolutionary Computation*, 2007, pp. 1677–1684.
- [11] D. Wu and J. M. Mendel, "Designing practical interval type-2 fuzzy logic systems made simple," in *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. IEEE, 2014, pp. 800–807.
- [12] J. Zeng and Z.-Q. Liu, *Type-2 fuzzy graphical models for pattern recognition*. Springer, 2015.
- [13] J. M. Mendel, *Uncertain rule-based fuzzy logic system: introduction and new directions*. Prentice-Hall, 2001.
- [14] A. B. Cara, I. Rojas, H. Pomares, C. Wagner, and H. Hagrass, "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.
- [15] A. B. Cara, C. Wagner, H. Hagrass, 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.
- [16] N. Sahab and H. Hagrass, "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.
- [17] L. A. Zadeh, "Fuzzy sets," *Information and control*, vol. 8, no. 3, pp. 338–353, 1965.
- [18] 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.
- [19] A. Pourabdollah, C. Wagner, J. Aladi, and J. M. Garibaldi, "Improved uncertainty capture for non-singleton fuzzy systems," *IEEE Trans. Fuzzy Syst.*, vol. 24, no. 6, pp. 1513–1524, 2016.
- [20] C. Wagner, A. Pourabdollah, J. McCulloch, R. John, and J. M. Garibaldi, "A similarity-based inference engine for non-singleton fuzzy logic systems," in *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 2016.
- [21] G. C. Mouzouris and J. M. Mendel, "Nonlinear predictive modeling using dynamic non-singleton fuzzy logic systems," in *the Fifth IEEE International Conference on Fuzzy Systems*, vol. 2. IEEE, 1996, pp. 1217–1223.
- [22] J. H. Aladi, C. Wagner, A. Pourabdollah, and J. M. Garibaldi, "Contrasting singleton type-1 and interval type-2 non-singleton type-1 fuzzy logic systems," in *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. IEEE, 2016, pp. 2043–2050.
- [23] 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.
- [24] M. C. Mackey, L. Glass *et al.*, "Oscillation and chaos in physiological control systems," *Science*, vol. 197, no. 4300, pp. 287–289, 1977.
- [25] 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.
- [26] 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.
- [27] J. Jang, "Anfis: adaptive-network-based fuzzy inference system," *IEEE Trans. Syst., Man, Cybern.*, vol. 23, no. 3, pp. 665–685, 1993.
- [28] 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.
- [29] E. N. Lorenz, "Deterministic nonperiodic flow," *Journal of the atmospheric sciences*, vol. 20, no. 2, pp. 130–141, 1963.