

Article

Constructing a Precise Fuzzy Feedforward Neural Network Using an Independent Fuzzification Approach

Hsin-Chieh Wu ¹, Tin-Chih Toly Chen ^{2,*} and Min-Chi Chiu ³ ¹ Department of Industrial Engineering and Management, Chaoyang University of Science and Technology, Taichung 413310, Taiwan; wusteve11@gmail.com² Department of Industrial Engineering and Management, National Yang Ming Chiao Tung University, 1001, University Road, Hsinchu 30010, Taiwan³ Department of Industrial Engineering and Management, National Chin-Yi University of Technology, Taichung 411030, Taiwan; mcchiu@ncut.edu.tw

* Correspondence: tolychen@ms37.hinet.net

Abstract: This study discusses how to fuzzify a feedforward neural network (FNN) to generate a fuzzy forecast that contains the actual value, while minimizing the average range of fuzzy forecasts. This topic has rarely been investigated in past studies, but is an essential step to constructing a precise fuzzy FNN (FFNN). Existing methods fuzzify all parameters at the same time, which results in a nonlinear programming (NLP) problem that is not easy to solve. In contrast, in this study, the parameters of a FNN are fuzzified independently. In this way, the optimal values of fuzzy parameters can be derived theoretically. An illustrative example is used to illustrate the applicability of the proposed methodology. According to the experimental results, fuzzifying the thresholds on hidden-layer nodes or the connection weights between input and hidden layers may not guarantee that all fuzzy forecasts contain the corresponding actual values. In contrast, fuzzifying the threshold on the output node and the connection weights between the hidden and output layers is more likely to achieve a 100% hit rate. The results lay a foundation for establishing a precise deep FFNN in the future.

Keywords: feedforward neural network; fuzzy neural network; forecasting; precision



Citation: Wu, H.-C.; Chen, T.-C.T.; Chiu, M.-C. Constructing a Precise Fuzzy Feedforward Neural Network Using an Independent Fuzzification Approach. *Axioms* **2021**, *10*, 282. <https://doi.org/10.3390/axioms10040282>

Academic Editor: Ming-Fu Hsu

Received: 8 October 2021

Accepted: 26 October 2021

Published: 28 October 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Fuzzy feedforward neural networks (FFNNs) combines the advantages of fuzzy logic (in uncertainty modelling) and feedforward neural networks (FNNs) (in nonlinear approximation) [1], and have been widely applied to forecasting in many fields [2–5]. There are various types of FFNNs with fuzzy or crisp inputs, parameters, and outputs. The numbers of layers and activation (or transformation) functions in these FFNNs are also different [6]. A recent review on FFNNs refers to de Campos Souza [7]. At present, the most commonly applied FFNNs are the variants of adaptive network-based fuzzy inference system (ANFIS) [8–13]. Past studies have shown that FFNNs can improve the forecasting accuracy, that is, each forecast is close to the actual value [14–16]. However, the present study aims to construct an FFNN to improve the forecasting precision, that is, every actual value is included in the narrowest possible fuzzy forecast. This topic has rarely been discussed in the past, which constitutes the motivation of this research.

However, even if a sophisticated FFNN is applied, the network output is rarely equal to the actual value, especially when the FFNN is applied to unlearned data. To address this issue, an alternative is to estimate the range of the actual value [17]. In other words, a fuzzy forecast that contains the actual value needs to be generated by an FFNN, at least for the training data. However, it is not easy since there are no actual values of the lower and upper bounds of the range. In addition, a fuzzy forecast needs to be as narrow as possible to have a reference value [18]. This is also a challenging task because a narrower

range is less likely to contain the actual value. Some of the relevant literature are reviewed as follows.

In an ANFIS, the network output before defuzzification cannot guarantee the inclusion of the actual value [19]. The problem is even more complicated for an FFNN in which all parameters are fuzzy and nonlinear transformation functions (such as sigmoid or tansig functions) are applied. For example, Chen and Wang [20] showed that the problem of deriving the values of fuzzy parameters in an FFNN was a nonlinear programming (NLP) problem that was difficult to solve. A branch-and-bound algorithm can be applied to find a solution to the NLP problem [21], but the solution may be far from (global) optimal. Instead, Chen and Wang established goals for the lower and upper bounds of the actual value to simplify the NLP problem to a goal programming (GP) problem. However, a number of goals needed to be tried to improve the solution, which was time-consuming. A similar method was proposed in Chen and Lin [18], in which the membership of an actual value in the fuzzy forecast had to be greater than a specified level. If only the threshold on the output node was fuzzy, the optimal value of the fuzzy threshold can be derived by solving two linear equations [22]. Similar treatments have been taken by Chen and Wu [17] and Chen [23]. Wang et al. [24] randomized the values of fuzzy thresholds on hidden-layer nodes and then optimized the fuzzy threshold on the output node. After a few replications, these fuzzy thresholds could be optimized. However, connection weights in the FFNN were still crisp.

This study considers an FFNN with a single hidden layer in which all network parameters can be fuzzified and nonlinear transformation functions (i.e., sigmoid functions) are adopted. We aim to optimize the values of fuzzy parameters theoretically without solving an NLP problem, while guaranteeing that all actual values are contained in the corresponding fuzzy forecasts. However, instead of fuzzifying all parameters at the same time, this study follows an independent fuzzification approach in which parameters are fuzzified independently. This study is important because it is a fundamental step towards the construction of a precise FFNN, which lays a foundation for establishing a precise deep FFNN with multiple or recurrent hidden layers.

The contribution of this research is to derive the formula for optimizing the value of each fuzzy parameter in an FFNN, so as to minimize the average range of fuzzy forecasts while ensuring a 100% hit rate. In contrast, existing methods need to solve an NLP problem to achieve the same goal.

The remainder of this study is organized as follows. The independent fuzzification approach is detailed in Section 2. A numerical example is given in Section 3 to illustrate the applicability of the proposed methodology. The effects of fuzzifying various parameters on the average range of fuzzy forecasts are also compared. This study is concluded in Section 4. Some directions for future investigation are also provided.

2. Independent Fuzzification Approach

All parameters and variables in the proposed methodology are given in or approximated by triangular fuzzy numbers (TFNs).

2.1. FFNN Configuration

The FFNN considered in this study is an FFNN that has three layers: the input layer, a single hidden layer, and the output layer. Inputs to the FFNN are indicated with $\{z_{jp} \mid p = 1 \sim P; j = 1 \sim n\}$. z_{jp} is the normalized value of decision variable x_{jp} :

$$z_{jp} = \frac{x_{jp} - \min_v x_{vp}}{\max_v x_{vp} - \min_v x_{vp}} \quad (1)$$

To convert back to the original value,

$$\begin{aligned} x_{jp} &= U(z_{jp}) \\ &= z_{jp}(\max_v x_{vp} - \min_v x_{vp}) + \min_v x_{vp} \end{aligned} \quad (2)$$

These inputs are propagated through the FFNN as follows. First, from the input layer to the hidden layer, the following operations are performed:

$$\tilde{I}_{jl}^h = \sum_{p=1}^P (\tilde{w}_{pl}^h z_{jp}) \quad (3)$$

$$\begin{aligned} \tilde{n}_{jl}^h &= \tilde{I}_{jl}^h (-) \tilde{\theta}_l^h \\ &= (\tilde{I}_{jl1}^h - \theta_{l3}^h, \tilde{I}_{jl2}^h - \theta_{l2}^h, \tilde{I}_{jl3}^h - \theta_{l1}^h) \end{aligned} \quad (4)$$

$$\tilde{h}_{jl} = \frac{1}{1 + e^{-\tilde{n}_{jl}^h}} \quad (5)$$

where $\tilde{w}_{pl}^h$ is the connection weight between input node p and hidden-layer node l ; $l = 1 \sim L$. $\tilde{\theta}_l^h$ is the threshold on hidden-layer node l . $\tilde{h}_{jl}$ is the output from hidden-layer node l . $(-)$ denotes fuzzy subtraction. In Equation (5), the activation (or transformation) function is the logistic sigmoid activation function that returns a value within $[0, 1]$.

Outputs from the hidden layer are aggregated on the output node,

$$\tilde{I}_j^o = \sum_{l=1}^L (\tilde{w}_l^o (\times) \tilde{h}_{jl}) \quad (6)$$

and then the network output $\tilde{o}_j = (o_{j1}, o_{j2}, o_{j3})$ is generated as

$$\tilde{o}_j = \frac{1}{1 + e^{-\tilde{n}_j^o}} \quad (7)$$

where

$$\tilde{n}_j^o = \tilde{I}_j^o (-) \tilde{\theta}^o \quad (8)$$

$\tilde{w}_l^o$ is the connection weight between hidden-layer node l and the output node. $\tilde{\theta}^o$ is the threshold on the output node. $\tilde{o}_j$ is unnormalized according to Equation (2) and then compared with the actual value y_j .

2.2. Deriving the Cores of Fuzzy Parameters

The training of the FFNN is composed of two stages. First, the cores of fuzzy parameters are derived by training the FFNN as a crisp FNN using the Levenberg–Marquardt (LM) algorithm [25], so as to minimize the mean squared error (MSE). The optimal solution is indicated with $\{w_{pl2}^{h*}, \theta_{l2}^{h*}, w_{l2}^{o*}, \theta_2^{o*} \mid p = 1 \sim P; l = 1 \sim L\}$.

Subsequently, the lower and upper bounds of fuzzy parameters are to be determined, so as to minimize the average range (AR) of fuzzy forecasts:

$$\text{Min AR} = \frac{1}{n} \sum_{j=1}^n (o_{j3} - o_{j1}) \quad (9)$$

However, deriving the optimal values of all fuzzy parameters at the same time is a computationally intensive task [20]. As an alternative, the optimal values of fuzzy parameters are derived independently as follows.

2.3. Deriving the Optimal Value of $\tilde{\theta}^o$

The optimal value of $\tilde{\theta}^o$ is to be derived. First, substituting Equation (8) into Equation (7) gives

$$\tilde{o}_j = \frac{1}{1 + e^{(\tilde{\theta}^o(-)I_j^o)}} \quad (10)$$

which can be decomposed into

$$o_{j1} = \frac{1}{1 + e^{(\theta_3^o - I_{j1}^o)}} \quad (11)$$

$$o_{j3} = \frac{1}{1 + e^{(\theta_1^o - I_{j3}^o)}} \quad (12)$$

The other parameters are not fuzzified, so I_{j1}^o and I_{j3}^o are equal to I_{j2}^{o*} that is a fixed value:

$$o_{j1} = \frac{1}{1 + e^{(\theta_3^o - I_{j2}^{o*})}} \quad (13)$$

$$o_{j3} = \frac{1}{1 + e^{(\theta_1^o - I_{j2}^{o*})}} \quad (14)$$

To minimize AR , o_{j1} and o_{j3} should be maximized and minimized, respectively, which correspond to the minimization of θ_3^o and the maximization of θ_1^o . However, $\tilde{o}_j$ should include a_j (the actual value), therefore

$$o_{j1} \leq a_j; j = 1 \sim n \quad (15)$$

$$o_{j3} \geq a_j; j = 1 \sim n \quad (16)$$

In addition, $o_{j1} \leq o_{j2}^* \leq o_{j3}$, so

$$o_{j1} \leq \min(o_{j2}^*, a_j); j = 1 \sim n \quad (17)$$

$$o_{j3} \geq \max(o_{j2}^*, a_j); j = 1 \sim n \quad (18)$$

Substituting Equation (13) into Constraint (17) gives

$$\theta_3^o \geq I_{j2}^{o*} + \ln\left(\frac{1}{\min(o_{j2}^*, a_j)} - 1\right); j = 1 \sim n \quad (19)$$

Therefore,

$$\theta_3^o \geq \max_j \left(I_{j2}^{o*} + \ln\left(\frac{1}{\min(o_{j2}^*, a_j)} - 1\right) \right) \quad (20)$$

To minimize θ_3^o ,

$$\theta_3^{o*} = \max_j \left(I_{j2}^{o*} + \ln\left(\frac{1}{\min(o_{j2}^*, a_j)} - 1\right) \right) \quad (21)$$

Similarly, by substituting Equation (14) into Constraint (18), the following result can be obtained:

$$\theta_1^{o*} = \min_j \left(I_{j2}^{o*} + \ln\left(\frac{1}{\max(o_{j2}^*, a_j)} - 1\right) \right) \quad (22)$$

Most past studies [17,22–24] stop at this step. The following discussion is new to the body of knowledge.

2.4. Deriving the Optimal Value of $\tilde{w}_{l_f}^o$

The optimal value of $\tilde{w}_l^o$ is to be derived when $l = l_f$. Equation (6) can be decomposed into

$$I_{j1}^o = \min \left(\sum_{l=1}^L (w_{l1}^o h_{jl1}), \sum_{l=1}^L (w_{l1}^o h_{jl3}) \right) \quad (23)$$

$$I_{j3}^o = \max \left(\sum_{l=1}^L (w_{l3}^o h_{jl1}), \sum_{l=1}^L (w_{l3}^o h_{jl3}) \right) \quad (24)$$

because $\tilde{h}_{jl}$ is positive, while $\tilde{w}_l^o$ may be negative. Substituting Equations (23) and (24) into Equations (13) and (14) gives

$$o_{j1} = \frac{1}{1 + e^{\left(\theta_3^o - \min \left(\sum_{l=1}^L (w_{l1}^o h_{jl1}), \sum_{l=1}^L (w_{l1}^o h_{jl3}) \right) \right)}} \quad (25)$$

$$o_{j3} = \frac{1}{1 + e^{\left(\theta_1^o - \max \left(\sum_{l=1}^L (w_{l3}^o h_{jl1}), \sum_{l=1}^L (w_{l3}^o h_{jl3}) \right) \right)}} \quad (26)$$

Only connection weights are fuzzified. The other fuzzy parameters are set to their optimized cores as

$$\begin{aligned} o_{j1} &= \frac{1}{1 + e^{\left(\theta_2^{o*} - \min \left(\sum_{l=1}^L (w_{l1}^o h_{jl2}^*), \sum_{l=1}^L (w_{l1}^o h_{jl2}^{*}) \right) \right)}} \\ &= \frac{1}{1 + e^{\left(\theta_2^{o*} - \sum_{l=1}^L (w_{l1}^o h_{jl2}^{*}) \right)}} \end{aligned} \quad (27)$$

$$\begin{aligned} o_{j3} &= \frac{1}{1 + e^{\left(\theta_2^{o*} - \max \left(\sum_{l=1}^L (w_{l3}^o h_{jl2}^*), \sum_{l=1}^L (w_{l3}^o h_{jl2}^{*}) \right) \right)}} \\ &= \frac{1}{1 + e^{\left(\theta_2^{o*} - \sum_{l=1}^L (w_{l3}^o h_{jl2}^{*}) \right)}} \end{aligned} \quad (28)$$

Substituting Equations (27) and (28) respectively into Constraints (17) and (18), we obtain

$$\sum_{l=1}^L (w_{l1}^o h_{jl2}^*) \leq \theta_2^{o*} - \ln \left(\frac{1}{\min(o_{j2}^*, a_j)} - 1 \right); j = 1 \sim n \quad (29)$$

$$\sum_{l=1}^L (w_{l3}^o h_{jl2}^*) \geq \theta_2^{o*} - \ln \left(\frac{1}{\max(o_{j2}^*, a_j)} - 1 \right); j = 1 \sim n \quad (30)$$

Only $w_{l_f}^o$ is fuzzified, the other connection weights are equal to their optimized cores:

$$w_{l_f1}^o h_{jl_f2}^* + \sum_{l \neq l_f} (w_{l2}^{o*} h_{jl2}^*) \leq \theta_2^{o*} - \ln \left(\frac{1}{\min(o_{j2}^*, a_j)} - 1 \right); j = 1 \sim n \quad (31)$$

$$w_{l_f3}^o h_{jl_f2}^* + \sum_{l \neq l_f} (w_{l2}^{o*} h_{jl2}^*) \geq \theta_2^{o*} - \ln \left(\frac{1}{\max(o_{j2}^*, a_j)} - 1 \right); j = 1 \sim n \quad (32)$$

As a result,

$$w_{l_f1}^o \leq \frac{\theta_2^{o*} - \ln \left(\frac{1}{\min(o_{j2}^*, a_j)} - 1 \right) - \sum_{l \neq l_f} (w_{l2}^{o*} h_{jl2}^*)}{h_{jl_f2}^*}; j = 1 \sim n \quad (33)$$

$$w_{l_f 3}^o \geq \frac{\theta_2^{o*} - \ln\left(\frac{1}{\max(o_{j2}^*, a_j)} - 1\right) - \sum_{l \neq l_f} (w_{l2}^{o*} h_{jl2}^*)}{h_{jl_f 2}^*}; j = 1 \sim n \quad (34)$$

Therefore,

$$w_{l_f 1}^o \leq \min_j \left(\frac{\theta_2^{o*} - \ln\left(\frac{1}{\min(o_{j2}^*, a_j)} - 1\right) - \sum_{l \neq l_f} (w_{l2}^{o*} h_{jl2}^*)}{h_{jl_f 2}^*} \right) \quad (35)$$

$$w_{l_f 3}^o \geq \max_j \left(\frac{\theta_2^{o*} - \ln\left(\frac{1}{\max(o_{j2}^*, a_j)} - 1\right) - \sum_{l \neq l_f} (w_{l2}^{o*} h_{jl2}^*)}{h_{jl_f 2}^*} \right) \quad (36)$$

Increasing the fuzziness (i.e., width) of $\tilde{w}_{l_f}^o$ will make $\tilde{o}_j$ wider. Therefore, it is reasonable to set

$$w_{l_f 1}^{o*} = \min_j \left(\frac{\theta_2^{o*} - \ln\left(\frac{1}{\min(o_{j2}^*, a_j)} - 1\right) - \sum_{l \neq l_f} (w_{l2}^{o*} h_{jl2}^*)}{h_{jl_f 2}^*} \right) \quad (37)$$

$$w_{l_f 3}^{o*} = \max_j \left(\frac{\theta_2^{o*} - \ln\left(\frac{1}{\max(o_{j2}^*, a_j)} - 1\right) - \sum_{l \neq l_f} (w_{l2}^{o*} h_{jl2}^*)}{h_{jl_f 2}^*} \right) \quad (38)$$

2.5. Deriving the Optimal Value of $\tilde{\theta}_l^h$

Substituting Equation (4) into Equation (5) gives

$$\tilde{h}_{jl} = \frac{1}{1 + e^{(\tilde{\theta}_l^h(-) \tilde{I}_{jl}^h)}} \quad (39)$$

which can be decomposed into

$$h_{jl1} = \frac{1}{1 + e^{(\theta_{l3}^h - I_{jl1}^h)}} \quad (40)$$

$$h_{jl3} = \frac{1}{1 + e^{(\theta_{l1}^h - I_{jl3}^h)}} \quad (41)$$

Substituting Equations (40) and (41) into Equations (23) and (24) leads to

$$I_{j1}^o = \min \left(\sum_{l=1}^L \frac{w_{l1}^o}{1 + e^{(\theta_{l3}^h - I_{jl1}^h)}}, \sum_{l=1}^L \frac{w_{l1}^o}{1 + e^{(\theta_{l1}^h - I_{jl3}^h)}} \right) \quad (42)$$

$$I_{j3}^o = \max \left(\sum_{l=1}^L \frac{w_{l3}^o}{1 + e^{(\theta_{l3}^h - I_{jl1}^h)}}, \sum_{l=1}^L \frac{w_{l3}^o}{1 + e^{(\theta_{l1}^h - I_{jl3}^h)}} \right) \quad (43)$$

that are substituted into Equations (13) and (14):

$$o_{j1} = \frac{1}{1 + e^{\left(\theta_3^o - \min \left(\sum_{l=1}^L \frac{w_{l1}^o}{1 + e^{(\theta_{l3}^h - I_{jl1}^h)}}, \sum_{l=1}^L \frac{w_{l1}^o}{1 + e^{(\theta_{l1}^h - I_{jl3}^h)}} \right) \right)}} \quad (44)$$

$$o_{j3} = \frac{1}{1 + e^{\left(\theta_1^0 - \max \left(\sum_{l=1}^L \frac{w_{l3}^0}{1 + e^{(\theta_{l3}^h - I_{jl1}^h)}}, \sum_{l=1}^L \frac{w_{l3}^0}{1 + e^{(\theta_{l1}^h - I_{jl3}^h)}} \right) \right)}} \quad (45)$$

The following requirements should be met:

$$\frac{1}{1 + e^{\left(\theta_3^0 - \min \left(\sum_{l=1}^L \frac{w_{l1}^0}{1 + e^{(\theta_{l3}^h - I_{jl1}^h)}}, \sum_{l=1}^L \frac{w_{l1}^0}{1 + e^{(\theta_{l1}^h - I_{jl3}^h)}} \right) \right)}} \leq \min(o_{j2}^*, a_j); j = 1 \sim n \quad (46)$$

$$\frac{1}{1 + e^{\left(\theta_1^0 - \max \left(\sum_{l=1}^L \frac{w_{l3}^0}{1 + e^{(\theta_{l3}^h - I_{jl1}^h)}}, \sum_{l=1}^L \frac{w_{l3}^0}{1 + e^{(\theta_{l1}^h - I_{jl3}^h)}} \right) \right)}} \geq \max(o_{j2}^*, a_j); j = 1 \sim n \quad (47)$$

that are equivalent to

$$\min \left(\sum_{l=1}^L \frac{w_{l1}^0}{1 + e^{(\theta_{l3}^h - I_{jl1}^h)}}, \sum_{l=1}^L \frac{w_{l1}^0}{1 + e^{(\theta_{l1}^h - I_{jl3}^h)}} \right) \leq \theta_3^0 - \ln \left(\frac{1}{\min(o_{j2}^*, a_j)} - 1 \right); j = 1 \sim n \quad (48)$$

$$\max \left(\sum_{l=1}^L \frac{w_{l3}^0}{1 + e^{(\theta_{l3}^h - I_{jl1}^h)}}, \sum_{l=1}^L \frac{w_{l3}^0}{1 + e^{(\theta_{l1}^h - I_{jl3}^h)}} \right) \geq \theta_1^0 - \ln \left(\frac{1}{\max(o_{j2}^*, a_j)} - 1 \right); j = 1 \sim n \quad (49)$$

Only $\theta_{l_f}^h$ is fuzzified, the other fuzzy parameters are set to their optimized cores:

$$\min \left(\frac{w_{l_f2}^{0*}}{1 + e^{(\theta_{l_f3}^h - I_{jl_f2}^{h*})}} + \sum_{l \neq l_f} \frac{w_{l2}^{0*}}{1 + e^{(\theta_{l2}^{h*} - I_{jl2}^{h*})}}, \frac{w_{l_f2}^{0*}}{1 + e^{(\theta_{l_f1}^h - I_{jl_f2}^{h*})}} + \sum_{l \neq l_f} \frac{w_{l2}^{0*}}{1 + e^{(\theta_{l2}^{h*} - I_{jl2}^{h*})}} \right) \leq \theta_2^{0*} - \ln \left(\frac{1}{\min(o_{j2}^*, a_j)} - 1 \right); j = 1 \sim n \quad (50)$$

$$\max \left(\frac{w_{l_f2}^{0*}}{1 + e^{(\theta_{l_f3}^h - I_{jl_f2}^{h*})}} + \sum_{l \neq l_f} \frac{w_{l2}^{0*}}{1 + e^{(\theta_{l2}^{h*} - I_{jl2}^{h*})}}, \frac{w_{l_f2}^{0*}}{1 + e^{(\theta_{l_f1}^h - I_{jl_f2}^{h*})}} + \sum_{l \neq l_f} \frac{w_{l2}^{0*}}{1 + e^{(\theta_{l2}^{h*} - I_{jl2}^{h*})}} \right) \geq \theta_2^{0*} - \ln \left(\frac{1}{\max(o_{j2}^*, a_j)} - 1 \right); j = 1 \sim n \quad (51)$$

If $w_{l_f2}^{0*} \geq 0$,

$$\frac{w_{l_f2}^{0*}}{1 + e^{(\theta_{l_f3}^h - I_{jl_f2}^{h*})}} + \sum_{l \neq l_f} \frac{w_{l2}^{0*}}{1 + e^{(\theta_{l2}^{h*} - I_{jl2}^{h*})}} \leq \theta_2^{0*} - \ln \left(\frac{1}{\min(o_{j2}^*, a_j)} - 1 \right); j = 1 \sim n \quad (52)$$

$$\frac{w_{l_f2}^{0*}}{1 + e^{(\theta_{l_f1}^h - I_{jl_f2}^{h*})}} + \sum_{l \neq l_f} \frac{w_{l2}^{0*}}{1 + e^{(\theta_{l2}^{h*} - I_{jl2}^{h*})}} \geq \theta_2^{0*} - \ln \left(\frac{1}{\max(o_{j2}^*, a_j)} - 1 \right); j = 1 \sim n \quad (53)$$

As a result,

$$\theta_{l_f3}^h \geq \ln \left(\frac{w_{l_f2}^{0*}}{\theta_2^{0*} - \ln \left(\frac{1}{\min(o_{j2}^*, a_j)} - 1 \right) - \sum_{l \neq l_f} \frac{w_{l2}^{0*}}{1 + e^{(\theta_{l2}^{h*} - I_{jl2}^{h*})}}} - 1 \right) + I_{jl_f2}^{h*}; j = 1 \sim n \quad (54)$$

$$\theta_{l_f1}^h \leq \ln \left(\frac{w_{l_f2}^{0*}}{\theta_2^{0*} - \ln \left(\frac{1}{\max(o_{j2}^*, a_j)} - 1 \right) - \sum_{l \neq l_f} \frac{w_{l2}^{0*}}{1 + e^{(\theta_{l2}^{h*} - I_{jl2}^{h*})}}} - 1 \right) + I_{jl_f2}^{h*}; j = 1 \sim n \quad (55)$$

Therefore,

$$\theta_{l_f1}^h \leq \min_j \left(\ln \left(\frac{w_{l_f2}^{0*}}{\theta_2^{0*} - \ln \left(\frac{1}{\max(o_{j2}^*, a_j)} - 1 \right) - \sum_{l \neq l_f} \frac{w_{l2}^{0*}}{1 + e^{(\theta_{l2}^{h*} - I_{jl2}^{h*})}}} - 1 \right) + I_{jl_f2}^{h*} \right) \quad (56)$$

$$\theta_{l_f3}^h \geq \max_j \left(\ln \left(\frac{w_{l_f2}^{o*}}{\theta_2^{o*} - \ln \left(\frac{1}{\min(o_{j2}^*, a_j)} \right) - 1} - 1 \right) + I_{j l_f2}^{h*} \right) \quad (57)$$

To minimize the fuzziness of $\hat{\theta}_{l_f}^h$,

$$\theta_{l_f1}^{h*} = \min_j \left(\ln \left(\frac{w_{l_f2}^{o*}}{\theta_2^{o*} - \ln \left(\frac{1}{\max(o_{j2}^*, a_j)} \right) - 1} - 1 \right) + I_{j l_f2}^{h*} \right) \quad (58)$$

$$\theta_{l_f3}^{h*} = \max_j \left(\ln \left(\frac{w_{l_f2}^{o*}}{\theta_2^{o*} - \ln \left(\frac{1}{\min(o_{j2}^*, a_j)} \right) - 1} - 1 \right) + I_{j l_f2}^{h*} \right) \quad (59)$$

Otherwise,

$$\frac{w_{l_f2}^{o*}}{1 + e^{(\theta_{l_f1}^h - I_{j l_f2}^{h*})}} + \sum_{l \neq l_f} \frac{w_{l2}^{o*}}{1 + e^{(\theta_{l2}^{h*} - I_{j l2}^{h*})}} \leq \theta_2^{o*} - \ln \left(\frac{1}{\min(o_{j2}^*, a_j)} \right) - 1; j = 1 \sim n \quad (60)$$

$$\frac{w_{l_f2}^{o*}}{1 + e^{(\theta_{l_f3}^h - I_{j l_f2}^{h*})}} + \sum_{l \neq l_f} \frac{w_{l2}^{o*}}{1 + e^{(\theta_{l2}^{h*} - I_{j l2}^{h*})}} \geq \theta_2^{o*} - \ln \left(\frac{1}{\max(o_{j2}^*, a_j)} \right) - 1; j = 1 \sim n \quad (61)$$

As a result,

$$\theta_{l_f1}^h \geq \ln \left(\frac{w_{l_f2}^{o*}}{\theta_2^{o*} - \ln \left(\frac{1}{\min(o_{j2}^*, a_j)} \right) - 1} - 1 \right) + I_{j l_f2}^{h*}; j = 1 \sim n \quad (62)$$

$$\theta_{l_f3}^h \leq \ln \left(\frac{w_{l_f2}^{o*}}{\theta_2^{o*} - \ln \left(\frac{1}{\max(o_{j2}^*, a_j)} \right) - 1} - 1 \right) + I_{j l_f2}^{h*}; j = 1 \sim n \quad (63)$$

Therefore,

$$\theta_{l_f1}^h \geq \max_j \left(\ln \left(\frac{w_{l_f2}^{o*}}{\theta_2^{o*} - \ln \left(\frac{1}{\min(o_{j2}^*, a_j)} \right) - 1} - 1 \right) + I_{j l_f2}^{h*} \right) \quad (64)$$

$$\theta_{l_f3}^h \leq \min_j \left(\ln \left(\frac{w_{l_f2}^{o*}}{\theta_2^{o*} - \ln \left(\frac{1}{\max(o_{j2}^*, a_j)} \right) - 1} - 1 \right) + I_{j l_f2}^{h*} \right) \quad (65)$$

To minimize the fuzziness of $\hat{\theta}_{l_f}^h$, $\theta_{l_f1}^{h*}$ and $\theta_{l_f3}^{h*}$ should be maximized and minimized, respectively, but they are still bounded by $\theta_{l_f2}^{h*}$. Therefore,

$$\theta_{l_f1}^{h*} = \theta_{l_f2}^{h*} \quad (66)$$

$$\theta_{l_f3}^{h*} = \theta_{l_f2}^{h*} \quad (67)$$

2.6. Deriving the Optimal Value of $\tilde{w}_{pl}^h$

Equation (3) can be decomposed into

$$I_{jl1}^h = \sum_{p=1}^P (w_{pl1}^h z_{jp}) \quad (68)$$

$$I_{jl3}^h = \sum_{p=1}^P (w_{pl3}^h z_{jp}) \quad (69)$$

Substituting Equations (68) and (69) into Equations (46) and (47) gives

$$o_{j1} = \frac{1}{1+e^{\left(\theta_3^o - \min \left(\sum_{l=1}^L \frac{w_{l1}^o}{(\theta_{l3}^h - \sum_{p=1}^P (w_{pl1}^h z_{jp}))}, \sum_{l=1}^L \frac{w_{l1}^o}{(\theta_{l1}^h - \sum_{p=1}^P (w_{pl3}^h z_{jp}))} \right) \right)}} \quad (70)$$

$$o_{j3} = \frac{1}{1+e^{\left(\theta_1^o - \max \left(\sum_{l=1}^L \frac{w_{l3}^o}{(\theta_{l3}^h - \sum_{p=1}^P (w_{pl1}^h z_{jp}))}, \sum_{l=1}^L \frac{w_{l3}^o}{(\theta_{l1}^h - \sum_{p=1}^P (w_{pl3}^h z_{jp}))} \right) \right)}} \quad (71)$$

The following requirements need to be met:

$$\frac{1}{1+e^{\left(\theta_3^o - \min \left(\sum_{l=1}^L \frac{w_{l1}^o}{(\theta_{l3}^h - \sum_{p=1}^P (w_{pl1}^h z_{jp}))}, \sum_{l=1}^L \frac{w_{l1}^o}{(\theta_{l1}^h - \sum_{p=1}^P (w_{pl3}^h z_{jp}))} \right) \right)}} \leq \min(o_{j2}^*, a_j); j = 1 \sim n \quad (72)$$

$$\frac{1}{1+e^{\left(\theta_1^o - \max \left(\sum_{l=1}^L \frac{w_{l3}^o}{(\theta_{l3}^h - \sum_{p=1}^P (w_{pl1}^h z_{jp}))}, \sum_{l=1}^L \frac{w_{l3}^o}{(\theta_{l1}^h - \sum_{p=1}^P (w_{pl3}^h z_{jp}))} \right) \right)}} \geq \max(o_{j2}^*, a_j); j = 1 \sim n \quad (73)$$

that are equivalent to

$$\min \left(\sum_{l=1}^L \frac{w_{l1}^o}{1+e^{\left(\theta_{l3}^h - \sum_{p=1}^P (w_{pl1}^h x_{jp}) \right)}}, \sum_{l=1}^L \frac{w_{l1}^o}{1+e^{\left(\theta_{l1}^h - \sum_{p=1}^P (w_{pl3}^h x_{jp}) \right)}} \right) \leq \theta_3^o - \ln \left(\frac{1}{\min(o_{j2}^*, a_j)} - 1 \right); j = 1 \sim n \quad (74)$$

$$\max \left(\sum_{l=1}^L \frac{w_{l3}^o}{1+e^{\left(\theta_{l3}^h - \sum_{p=1}^P (w_{pl1}^h x_{jp}) \right)}}, \sum_{l=1}^L \frac{w_{l3}^o}{1+e^{\left(\theta_{l1}^h - \sum_{p=1}^P (w_{pl3}^h x_{jp}) \right)}} \right) \geq \theta_1^o - \ln \left(\frac{1}{\max(o_{j2}^*, a_j)} - 1 \right); j = 1 \sim n \quad (75)$$

Only w_{plf}^h is fuzzified, the other fuzzy parameters are set to their optimized cores:

$$\min \left(\frac{w_{l1f}^{o*}}{1+e^{\left(\theta_{l3}^{h*} - w_{plf}^{h*} x_{jp} - \sum_{p \neq p_f} (w_{plf}^{h*} x_{jp}) \right)}} + \sum_{l \neq l_f} \frac{w_{l2}^{o*}}{1+e^{\left(\theta_{l2}^{h*} - \sum_{p=1}^P (w_{pl2}^{h*} x_{jp}) \right)}}, \frac{w_{l3f}^{o*}}{1+e^{\left(\theta_{l3}^{h*} - w_{plf}^{h*} x_{jp} - \sum_{p \neq p_f} (w_{plf}^{h*} x_{jp}) \right)}} + \sum_{l \neq l_f} \frac{w_{l2}^{o*}}{1+e^{\left(\theta_{l2}^{h*} - \sum_{p=1}^P (w_{pl2}^{h*} x_{jp}) \right)}} \right) \leq \theta_2^{o*} - \ln \left(\frac{1}{\min(o_{j2}^*, a_j)} - 1 \right); j = 1 \sim n \quad (76)$$

$$\max \left(\frac{\frac{w_{lf2}^{o*}}{1+e} + \sum_{l \neq lf} \frac{w_{lf2}^{o*}}{1+e}}{\frac{(\theta_{lf2}^{h*} - w_{pflf1}^h x_{jp} - \sum_{p \neq p_f} (w_{plf2}^{h*} x_{jp}))}{1+e}}}, \frac{\frac{w_{lf2}^{o*}}{1+e} + \sum_{l \neq lf} \frac{w_{lf2}^{o*}}{1+e}}{\frac{(\theta_{lf2}^{h*} - w_{pflf3}^h x_{jp} - \sum_{p \neq p_f} (w_{plf2}^{h*} x_{jp}))}{1+e}}} \right) \geq \theta_2^{o*} - \ln\left(\frac{1}{\max(o_{j2}^*, a_j)} - 1\right); j = 1 \sim n \quad (77)$$

If $w_{lf2}^{o*} \geq 0$,

$$\frac{\frac{w_{lf2}^{o*}}{1+e} + \sum_{l \neq lf} \frac{w_{lf2}^{o*}}{1+e}}{\frac{(\theta_{lf2}^{h*} - w_{pflf1}^h x_{jp} - \sum_{p \neq p_f} (w_{plf2}^{h*} x_{jp}))}{1+e}} \geq \theta_2^{o*} - \ln\left(\frac{1}{\max(o_{j2}^*, a_j)} - 1\right); j = 1 \sim n \quad (78)$$

As a result,

$$w_{pflf1}^h \leq \frac{\theta_{lf2}^{h*} - \ln\left(\frac{w_{lf2}^{o*}}{\theta_2^{o*} - \ln\left(\frac{1}{\min(o_{j2}^*, a_j)} - 1\right) - \sum_{l \neq lf} \frac{w_{lf2}^{o*}}{1+e}} - 1\right) - \sum_{p \neq p_f} (w_{plf2}^{h*} x_{jp})}{x_{jp}}; j = 1 \sim n \quad (79)$$

$$w_{pflf3}^h \geq \frac{\theta_{lf2}^{h*} - \ln\left(\frac{w_{lf2}^{o*}}{\theta_2^{o*} - \ln\left(\frac{1}{\max(o_{j2}^*, a_j)} - 1\right) - \sum_{l \neq lf} \frac{w_{lf2}^{o*}}{1+e}} - 1\right) - \sum_{p \neq p_f} (w_{plf2}^{h*} x_{jp})}{x_{jp}}; j = 1 \sim n \quad (80)$$

Therefore,

$$w_{pflf1}^h \leq \min_j \left(\frac{\theta_{lf2}^{h*} - \ln\left(\frac{w_{lf2}^{o*}}{\theta_2^{o*} - \ln\left(\frac{1}{\min(o_{j2}^*, a_j)} - 1\right) - \sum_{l \neq lf} \frac{w_{lf2}^{o*}}{1+e}} - 1\right) - \sum_{p \neq p_f} (w_{plf2}^{h*} x_{jp})}{x_{jp}} \right) \quad (81)$$

$$w_{pflf3}^h \geq \max_j \left(\frac{\theta_{lf2}^{h*} - \ln\left(\frac{w_{lf2}^{o*}}{\theta_2^{o*} - \ln\left(\frac{1}{\max(o_{j2}^*, a_j)} - 1\right) - \sum_{l \neq lf} \frac{w_{lf2}^{o*}}{1+e}} - 1\right) - \sum_{p \neq p_f} (w_{plf2}^{h*} x_{jp})}{x_{jp}} \right) \quad (82)$$

To minimize the fuzziness of $\tilde{w}_{pflf}^h$,

$$w_{pflf1}^{h*} = \min_j \left(\frac{\theta_{lf2}^{h*} - \ln\left(\frac{w_{lf2}^{o*}}{\theta_2^{o*} - \ln\left(\frac{1}{\min(o_{j2}^*, a_j)} - 1\right) - \sum_{l \neq lf} \frac{w_{lf2}^{o*}}{1+e}} - 1\right) - \sum_{p \neq p_f} (w_{plf2}^{h*} x_{jp})}{x_{jp}} \right) \quad (83)$$

$$w_{p_f l_f 3}^{h*} = \max_j \left(\frac{\theta_{l_f 2}^{h*} - \ln \left(\frac{w_{l_f 2}^{0*}}{\theta_2^{0*} - \ln \left(\frac{1}{\max(o_{j2}^*, a_j)} - 1 \right) - \sum_{l \neq l_f} \frac{w_{l2}^{0*}}{(\theta_{l2}^{h*} - \sum_{p=1}^P (w_{pl2}^{h*} x_{jp}))}} - 1 \right) - \sum_{p \neq p_f} (w_{pl_f 2}^{h*} x_{jp})}{x_{jp_f}} \right) \quad (84)$$

Otherwise,

$$\frac{w_{l_f 2}^{0*}}{1 + e^{(\theta_{l_f 2}^{h*} - w_{p_f l_f 3}^h x_{jp_f} - \sum_{p \neq p_f} (w_{pl_f 2}^{h*} x_{jp}))}} + \sum_{l \neq l_f} \frac{w_{l2}^{0*}}{1 + e^{(\theta_{l2}^{h*} - \sum_{p=1}^P (w_{pl2}^{h*} x_{jp}))}} \leq \theta_2^{0*} - \ln \left(\frac{1}{\min(o_{j2}^*, a_j)} - 1 \right); j = 1 \sim n \quad (85)$$

$$\frac{w_{l_f 2}^{0*}}{1 + e^{(\theta_{l_f 2}^{h*} - w_{p_f l_f 1}^h x_{jp_f} - \sum_{p \neq p_f} (w_{pl_f 2}^{h*} x_{jp}))}} + \sum_{l \neq l_f} \frac{w_{l2}^{0*}}{1 + e^{(\theta_{l2}^{h*} - \sum_{p=1}^P (w_{pl2}^{h*} x_{jp}))}} \geq \theta_2^{0*} - \ln \left(\frac{1}{\max(o_{j2}^*, a_j)} - 1 \right); j = 1 \sim n \quad (86)$$

As a result,

$$\frac{w_{l_f 2}^{0*}}{1 + e^{(\theta_{l_f 2}^{h*} - w_{p_f l_f 3}^h x_{jp_f} - \sum_{p \neq p_f} (w_{pl_f 2}^{h*} x_{jp}))}} + \sum_{l \neq l_f} \frac{w_{l2}^{0*}}{1 + e^{(\theta_{l2}^{h*} - \sum_{p=1}^P (w_{pl2}^{h*} x_{jp}))}} \leq \theta_2^{0*} - \ln \left(\frac{1}{\min(o_{j2}^*, a_j)} - 1 \right); j = 1 \sim n \quad (87)$$

$$\frac{w_{l_f 2}^{0*}}{1 + e^{(\theta_{l_f 2}^{h*} - w_{p_f l_f 1}^h x_{jp_f} - \sum_{p \neq p_f} (w_{pl_f 2}^{h*} x_{jp}))}} + \sum_{l \neq l_f} \frac{w_{l2}^{0*}}{1 + e^{(\theta_{l2}^{h*} - \sum_{p=1}^P (w_{pl2}^{h*} x_{jp}))}} \geq \theta_2^{0*} - \ln \left(\frac{1}{\max(o_{j2}^*, a_j)} - 1 \right); j = 1 \sim n \quad (88)$$

Therefore,

$$w_{p_f l_f 3}^h \geq \max_j \left(\frac{\theta_{l_f 2}^{h*} - \ln \left(\frac{w_{l_f 2}^{0*}}{\theta_2^{0*} - \ln \left(\frac{1}{\min(o_{j2}^*, a_j)} - 1 \right) - \sum_{l \neq l_f} \frac{w_{l2}^{0*}}{(\theta_{l2}^{h*} - \sum_{p=1}^P (w_{pl2}^{h*} x_{jp}))}} - 1 \right) - \sum_{p \neq p_f} (w_{pl_f 2}^{h*} x_{jp})}{x_{jp_f}} \right) \quad (89)$$

$$w_{p_f l_f 1}^h \leq \min_j \left(\frac{\theta_{l_f 2}^{h*} - \ln \left(\frac{w_{l_f 2}^{0*}}{\theta_2^{0*} - \ln \left(\frac{1}{\max(o_{j2}^*, a_j)} - 1 \right) - \sum_{l \neq l_f} \frac{w_{l2}^{0*}}{(\theta_{l2}^{h*} - \sum_{p=1}^P (w_{pl2}^{h*} x_{jp}))}} - 1 \right) - \sum_{p \neq p_f} (w_{pl_f 2}^{h*} x_{jp})}{x_{jp_f}} \right) \quad (90)$$

To minimize the fuzziness of $\tilde{w}_{p_f l_f}^h$,

$$w_{p_f l_f 3}^{h*} = \max_j \left(\frac{\theta_{l_f 2}^{h*} - \ln \left(\frac{w_{l_f 2}^{0*}}{\theta_2^{0*} - \ln \left(\frac{1}{\min(o_{j2}^*, a_j)} - 1 \right) - \sum_{l \neq l_f} \frac{w_{l2}^{0*}}{(\theta_{l2}^{h*} - \sum_{p=1}^P (w_{pl2}^{h*} x_{jp}))}} - 1 \right) - \sum_{p \neq p_f} (w_{pl_f 2}^{h*} x_{jp})}{x_{jp_f}} \right) \quad (91)$$

$$w_{p_f l_f 1}^{l_1*} = \min_j \left(\frac{\theta_{l_f 2}^{l_1*} - \ln \left(\frac{w_{l_f 2}^{l_1*}}{\theta_2^{l_1*} - \ln \left(\frac{1}{\max(a_j^*)} - 1 \right) - \sum_{l \neq l_f} \frac{w_{l_2}^{l_1*}}{(e_{l_2}^{l_1*} - \sum_{p=1}^p (w_{p l_2}^{l_1*} x_{jp}))} - 1 \right) - \sum_{p \neq p_f} (w_{p l_f 2}^{l_1*} x_{jp})}{x_{jp_f}} \right) \quad (92)$$

3. An Illustrative Case Using FFNN(3, 6, 1)

The problem of predicting the time to replace a computer numeric control (CNC) tool based on the monitoring results of three sensors is adopted to illustrate the applicability of the proposed methodology. Therefore, $P = 3$. A small-scale problem is used, so that the constructed FFNN will not be too large, and the effect of fuzzifying each parameter can be more obvious. The collected data includes ninety records, as shown in Table 1. The collected data are first normalized (see Table 2).

Table 1. Collected data.

j	x_{j1}	x_{j2}	x_{j3}	y_j
1	265	30	2028	468
2	224	40	2018	507
3	173	52	2641	811
4	151	36	1837	468
5	322	55	2274	776
6	167	56	2508	926
90	311	39	2170	468
Min	125	29	1173	463
Max	364	57	3269	967

Table 2. Normalized data.

j	z_{j1}	z_{j2}	z_{j3}	a_j
1	0.585	0.036	0.408	0.010
2	0.415	0.392	0.403	0.087
3	0.199	0.835	0.700	0.691
4	0.111	0.267	0.317	0.010
5	0.825	0.942	0.525	0.621
6	0.174	0.962	0.637	0.919
90	0.780	0.340	0.476	0.010

The FFNN has a hidden layer with six nodes. Therefore, it is indicated with FFNN(3, 6, 1) afterwards. There is no absolute rule for determining the optimal number of nodes in the hidden layer. In many studies, it has been shown that a hidden layer with twice the number of inputs is sufficient to fit a complex nonlinear relationship [26–28]. The first sixty records are used to train the FFNN, while the rest records are left for evaluating the forecasting performance.

First, the FFNN(3, 6, 1) is regarded as a crisp FNN(3, 6, 1) and trained using the LM algorithm to derive the cores of fuzzy parameters. Other training algorithms, such as the gradient descent (GD) algorithm, the Broyden–Fletcher–Goldfarb–Shanno (BFGS) quasi-Newton algorithm, the GD algorithm with momentum and adaptive learning rate

(GDX), and the resilient backpropagation (RP) algorithm [25], are also applicable. However, this research aims to improve the forecasting precision, rather than the forecasting accuracy. The choice of the training algorithm does not affect the application of the proposed methodology.

The optimal values of cores are summarized in Table 3. The crisp forecasts for the training data based on the cores of fuzzy parameters are shown in Figure 1. The forecasting accuracy, measured in terms of root mean squared error (RMSE), is 0.084 (normalized value). Although the forecasting accuracy is satisfactory, there are many records with considerable deviations between actual values and crisp forecasts, showing the necessity of estimating the range of the actual value. To this end, the effects of fuzzifying four parameters, θ^o , w_1^o , θ_1^h , and w_{11}^h , are compared. There are four types of parameters in the FNN(3, 6, 1): the connection weights between the input layer and the hidden layer, the thresholds on the nodes of the hidden layer, the connection weights between the hidden layer and the output layer, and the threshold on the node of the output layer. In this way, the effects of fuzzifying all types of parameters can be observed and compared.

Table 3. Optimized cores of fuzzy parameters.

w_{112}^{h*}	w_{122}^{h*}	w_{132}^{h*}	w_{142}^{h*}	w_{152}^{h*}	w_{162}^{h*}	w_{212}^{h*}	w_{222}^{h*}	w_{232}^{h*}	w_{242}^{h*}
2.034	3.410	−0.349	−2.615	−2.557	1.213	3.912	−1.746	1.813	3.187
w_{252}^{h*}	w_{262}^{h*}	w_{312}^{h*}	w_{322}^{h*}	w_{332}^{h*}	w_{342}^{h*}	w_{352}^{h*}	w_{362}^{h*}	θ_{12}^{h*}	θ_{22}^{h*}
−2.090	3.082	3.733	−0.424	3.964	3.831	−4.754	4.911	3.222	5.251
θ_{32}^{h*}	θ_{42}^{h*}	θ_{52}^{h*}	θ_{62}^{h*}	w_{12}^{o*}	w_{22}^{o*}	w_{32}^{o*}	w_{42}^{o*}	w_{52}^{o*}	w_{62}^{o*}
9.183	2.293	2.569	4.785	4.844	4.740	−2.641	3.727	−4.905	3.668
θ_2^{o*}									
9.475									

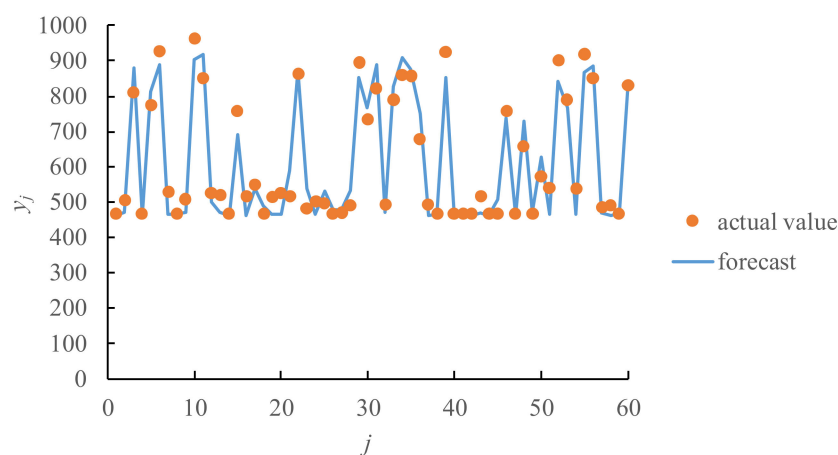


Figure 1. Crisp forecasts based on the cores of fuzzy parameters.

First, θ^o is fuzzified to minimize the average range of fuzzy forecasts. To this end, the lower and upper bounds of $\tilde{\theta}^o$ are to be derived. For this purpose, Equations (21) and (22) are applied. As a result, $\theta_1^{o*} = 4.646$ and $\theta_3^{o*} = 11.690$. Therefore, $\tilde{\theta}^o = (4.646, 9.475, 11.690)$. The estimated ranges by fuzzifying this parameter are shown in Figure 2. In this case, all ranges contain the corresponding actual values for the training data by fuzzifying θ^o . The average range of fuzzy forecasts is 303.9 (unnormlized value).

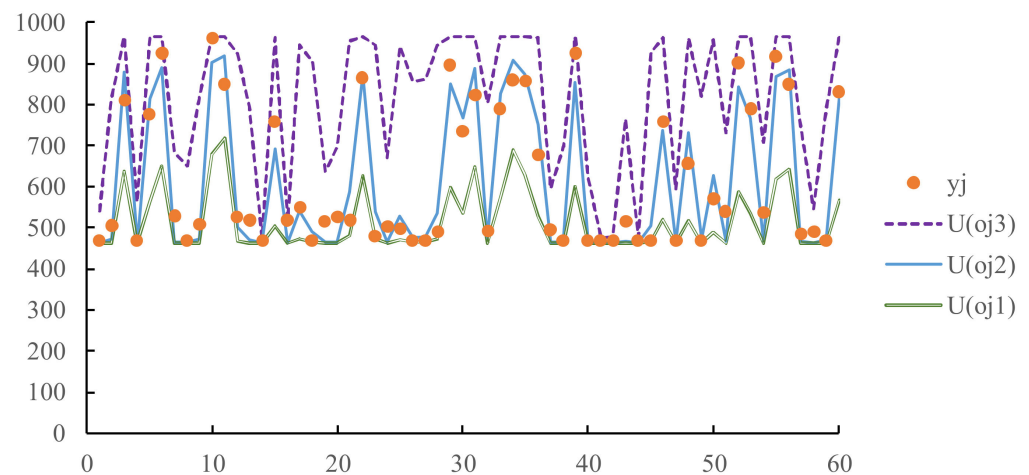


Figure 2. Estimated ranges by fuzzifying θ^o .

The second fuzzified network parameter is w_1^o . To this end, Equations (37) and (38) are applied to derive the lower and upper bounds of $\tilde{w}_1^o$. The result is $\tilde{w}_1^o = (1.892, 4.844, 66.88)$. The ranges estimated by fuzzifying w_1^o are shown in Figure 3. All actual values in the training data are contained in the corresponding fuzzy forecasts. However, the average range widens to 455.4.

Subsequently, θ_1^h is fuzzified by deriving its lower and upper bounds according to Equations (58) and (59), since w_{12}^{o*} is positive. The optimal solution is $\tilde{\theta}_1^h = (-6.477, 3.222, 5.798)$. The estimated ranges of actual values are shown in Figure 4. However, it is not possible for all fuzzy forecasts to contain the corresponding actual values solely by fuzzifying θ_1^h . The hit rate is only 67%. With such a low hit rate, the average range is narrowed to only 93.5.

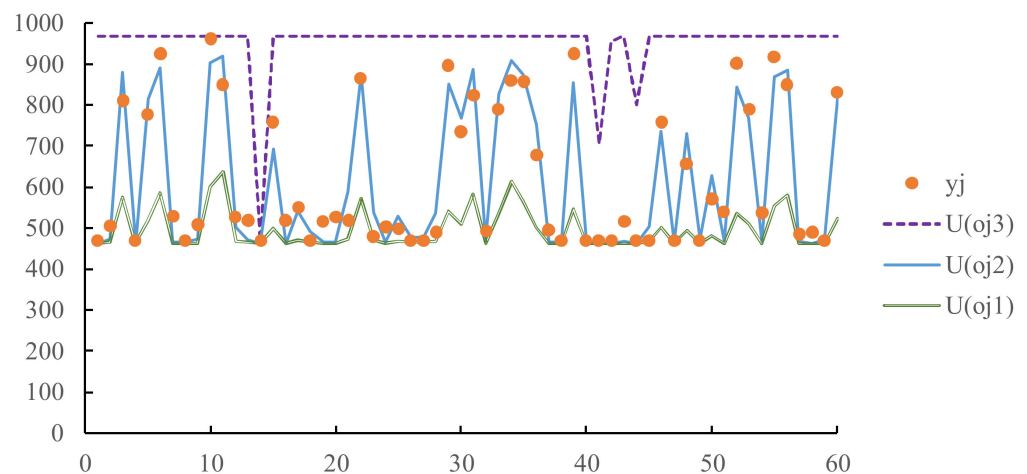


Figure 3. Ranges estimated by fuzzifying w_1^o .

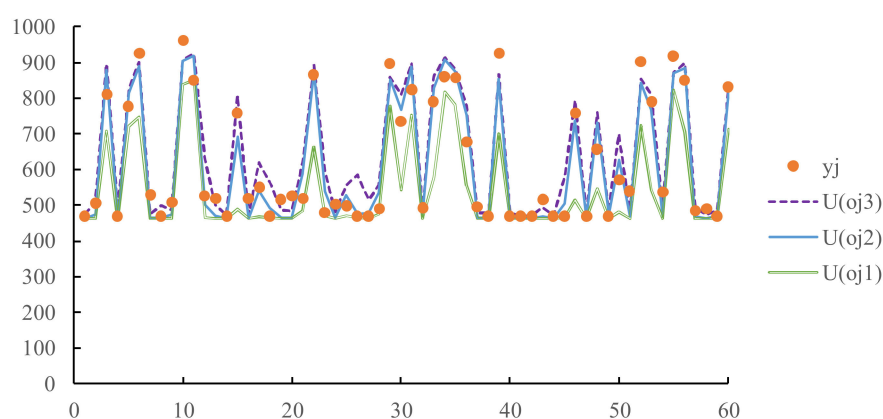


Figure 4. Estimated ranges after fuzzifying θ_1^h .

The last parameter fuzzified in the experiment is w_{11}^h . Since $w_{12}^{o*} \geq 0$, Equations (84) and (85) are applied to derive the lower and upper bounds of $\tilde{w}_{11}^h$. The result is $\tilde{w}_{11}^h = (-84.42, 2.034, 247.4)$. The ranges of actual values estimated by fuzzifying this parameter are summarized in Figure 5. The hit rate is only 65%, accompanied by an average range of 160.4.

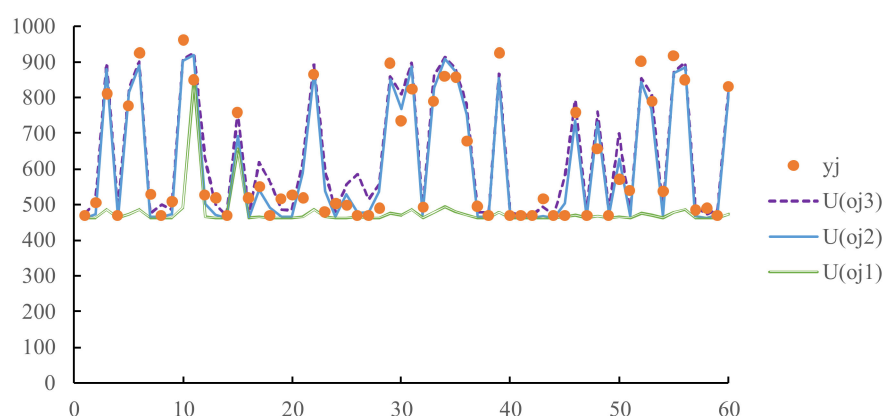


Figure 5. Ranges of actual values estimated by fuzzifying w_{11}^h .

From the experimental results, the following discussion is made:

1. Fuzzifying some network parameters may not guarantee that all actual values are contained in the estimated ranges.
2. In contrast, fuzzifying a network parameter closer to the output node is more like to ensure a 100% hit rate.
3. Both the ranges estimated by fuzzifying θ^o and w_1^o contain the actual value. Therefore, the fuzzy intersection (FI) of the ranges also contain the actual value, which further narrows the range of the actual value.
4. After applying the trained FFNN(3, 6, 1) to the test/unlearned data, the fore-casting precision levels achieved by fuzzifying various network parameters are evaluated and compared in Table 4. As expected, the hit rate has decreased compared to the results when applied to the training data, but is still acceptable. Fuzzifying w_1^o achieves the highest hit rate, while fuzzifying θ_1^h minimizes the average range of fuzzy forecasts.
5. The effectiveness (i.e., forecasting precision) and efficiency of the proposed methodology is compared with those of some existing methods in Table 5. All methods are implemented using MATLAB 2017a on a PC with an i7-7700 CPU of 3.6 GHz and 8 GB of RAM. Obviously, the proposed methodology maximized the hit rate for the test data without considerably widening the average range. In addition, the proposed methodology is also the most efficient method.

Table 4. Forecasting precision levels achieved by fuzzifying various network parameters.

	Fuzzifying θ^o	Fuzzifying w_1^o	Fuzzifying θ_1^h	Fuzzifying w_{11}^h
Hit Rate	97%	100%	63%	63%
Average Range	367.3	457.9	122.7	228.5

Table 5. Comparing the effectiveness and efficiency of the proposed methodology with those of existing methods.

Method	Hit Rate	Average Range	Execution Time (sec)
FFNN-NLP [18]	82%	407	168
FFNN-GP [20]	76%	482	46
FFNN (only $\tilde{\theta}^o$ fuzzified) [22]	97%	367	<1
The proposed methodology w_1^o fuzzified)	100%	458	<1

4. Conclusions and Future Research Directions

Many complex FFNNs have been constructed to improve the forecasting accuracy. Even so, a forecast is rarely equal to the actual value. In addition, the range of a fuzzy forecast generated by prevalent FFNNs does not necessarily include the actual value. In order to solve these problems, this research explores how to fuzzify the parameters of a FNN so that every fuzzy forecast generated by the FFNN contains the actual value. To achieve this goal, most previous studies have solved an NLP problem, which was computationally challenging. In contrast, this research proposes an independent fuzzification approach to fuzzify the parameters of a FNN independently. In this way, the optimal value of each fuzzy parameter can be derived theoretically, thereby enabling the construction of a precise FFNN.

After applying the proposed methodology to an illustrative case FFNN(3, 6, 1), the following conclusions are drawn:

1. Fuzzifying θ_1^h and w_{11}^h alone cannot guarantee that all fuzzy forecasts contain corresponding actual values.
2. Fuzzifying θ^o and w_1^o has a higher chance of achieving a 100% hit rate.
3. Parameters closer to the output node have a greater impact on the forecast-ing precision, and should be fuzzified earlier.
4. Fuzzifying parameters far away from the output node cannot guarantee a 100% hit rate. Therefore, multiple such parameters should be fuzzified at the same time.

The FFNN discussed in this study is an FFNN with a single hidden layer. The proposed methodology can be extended to deal with a deep FFNN with multiple hidden layers [29–31] or recurrent layers [32]. In this case, the parameters of the output layer will be fuzzified first, then the parameters of the hidden layer closest to the output layer, and so on. In addition, FI can also be applied to aggregate the estimated ranges by fuzzifying various parameters (with 100% hit rates) to further enhance the fore-casting precision. These constitute some direction for future research.

Author Contributions: All authors equally contributed to the writing of this paper. Data curation, methodology and writing original draft: T.-C.T.C., H.-C.W. and M.-C.C.; writing—review and editing: T.-C.T.C. and H.-C.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: This work was supported by Ministry of Science of Technology of Taiwan.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ishibuchi, H.; Tanaka, H.; Okada, H. Fuzzy neural networks with fuzzy weights and fuzzy biases. In Proceedings of the IEEE International Conference on Neural Networks, San Francisco, CA, USA, 28 March–1 April 1993; pp. 1650–1655.
2. Chen, S.X.; Gooi, H.B.; Wang, M.Q. Solar radiation forecast based on fuzzy logic and neural networks. *Renew. Energy* **2013**, *60*, 195–201. [\[CrossRef\]](#)
3. Hong, Y.Y.; Chang, H.L.; Chiu, C.S. Hour-ahead wind power and speed forecasting using simultaneous perturbation stochastic approximation (SPSA) algorithm and neural network with fuzzy inputs. *Energy* **2010**, *35*, 3870–3876. [\[CrossRef\]](#)
4. Kaynar, O.; Yilmaz, I.; Demirkoparan, F. Forecasting of natural gas consumption with neural network and neuro fuzzy system. *Energy Educ. Sci. Technol. Part A Energy Sci. Res.* **2011**, *26*, 221–238.
5. De Campos Souza, P.V.; Torres, L.C.B. Regularized fuzzy neural network based on or neuron for time series forecasting. In Proceedings of the North American Fuzzy Information Processing Society Annual Conference, Fortaleza, Brazil, 4–6 July 2018; pp. 13–23.
6. Jiang, Y.; Yang, C.; Ma, H. A review of fuzzy logic and neural network based intelligent control design for discrete-time systems. *Discret. Dyn. Nat. Soc.* **2016**, *2016*, 7217364. [\[CrossRef\]](#)
7. De Campos Souza, P.V. Fuzzy neural networks and neuro-fuzzy networks: A review the main techniques and applications used in the literature. *Appl. Soft Comput.* **2020**, *92*, 106275. [\[CrossRef\]](#)
8. Deng, Y.; Ren, Z.; Kong, Y.; Bao, F.; Dai, Q. A hierarchical fused fuzzy deep neural network for data classification. *IEEE Trans. Fuzzy Syst.* **2016**, *25*, 1006–1012. [\[CrossRef\]](#)
9. Rajurkar, S.; Verma, N.K. Developing deep fuzzy network with Takagi Sugeno fuzzy inference system. In Proceedings of the 2017 IEEE International Conference on Fuzzy Systems, Naples, Italy, 9–12 July 2017; pp. 1–6.
10. Mudiyanse, T.K.B.; Xiao, X.; Zhang, Y.; Pan, Y. Deep fuzzy neural networks for biomarker selection for accurate cancer detection. *IEEE Trans. Fuzzy Syst.* **2019**, *28*, 3219–3228. [\[CrossRef\]](#)
11. Liang, X.; Wang, G.; Min, M.R.; Qi, Y.; Han, Z. A deep spatio-temporal fuzzy neural network for passenger demand prediction. In Proceedings of the 2019 SIAM International Conference on Data Mining, Calgary, AB, Canada, 2–4 May 2019; pp. 100–108.
12. Qasem, S.N.; Mohammadzadeh, A. A deep learned type-2 fuzzy neural network: Singular value decomposition approach. *Appl. Soft Comput.* **2021**, *105*, 107244. [\[CrossRef\]](#)
13. Radulović, J.; Ranković, V. Feedforward neural network and adaptive network-based fuzzy inference system in study of power lines. *Expert Syst. Appl.* **2010**, *37*, 165–170. [\[CrossRef\]](#)
14. Hou, Y.; Zhao, L.; Lu, H. Fuzzy neural network optimization and network traffic forecasting based on improved differential evolution. *Future Gener. Comput. Syst.* **2018**, *81*, 425–432.
15. Sharifian, A.; Ghadi, M.J.; Ghavidel, S.; Li, L.; Zhang, J. A new method based on Type-2 fuzzy neural network for accurate wind power forecasting under uncertain data. *Renew. Energy* **2018**, *120*, 220–230.
16. Wen, Z.; Xie, L.; Fan, Q.; Feng, H. Long term electric load forecasting based on TS-type recurrent fuzzy neural network model. *Electr. Power Syst. Res.* **2020**, *179*, 106106.
17. Chen, T.; Wu, H.C. A new cloud computing method for establishing asymmetric cycle time intervals in a wafer fabrication factory. *J. Intell. Manuf.* **2017**, *28*, 1095–1107. [\[CrossRef\]](#)
18. Chen, T.C.; Lin, Y.C. A collaborative fuzzy-neural approach for internal due date assignment in a wafer fabrication plant. *Int. J. Innov. Comput. Inf. Control.* **2011**, *7*, 5193–5210.
19. Chen, T.C.T.; Honda, K. *Fuzzy Collaborative Forecasting and Clustering: Methodology, System Architecture, and Applications*; Springer International Publishing: Cham, Switzerland, 2020.
20. Chen, T.; Wang, Y.C. Incorporating the FCM-BPN approach with nonlinear programming for internal due date assignment in a wafer fabrication plant. *Robot. Comput. Integr. Manuf.* **2010**, *26*, 83–91. [\[CrossRef\]](#)
21. Tarray, T.A.; Bhat, M.R. A nonlinear programming problem using branch and bound method. *Investig. Oper.* **2018**, *38*, 291–298.
22. Chen, T. An effective fuzzy collaborative forecasting approach for predicting the job cycle time in wafer fabrication. *Comput. Ind. Eng.* **2013**, *66*, 834–848. [\[CrossRef\]](#)
23. Chen, T. An efficient and effective fuzzy collaborative intelligence approach for cycle time estimation in wafer fabrication. *Int. J. Intell. Syst.* **2012**, *30*, 620–650. [\[CrossRef\]](#)
24. Wang, Y.C.; Tsai, H.R.; Chen, T. A selectively fuzzified back propagation network approach for precisely estimating the cycle time range in wafer fabrication. *Mathematics* **2021**, *9*, 1430. [\[CrossRef\]](#)
25. Nocedal, J.; Wright, S. *Numerical Optimization*, 2nd ed.; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2006.
26. Ramchoun, H.; Idrissi, M.A.J.; Ghanou, Y.; Ettaoui, M. Multilayer perceptron: Architecture optimization and training. *Int. J. Interact. Multim. Artif. Intell.* **2016**, *4*, 26–30.
27. Cilimkovic, M. *Neural networks and back propagation algorithm*; Institute of Technology Blanchardstown: Dublin, Ireland, 2015; Volume 15, pp. 1–12.
28. Lin, Y.C.; Chen, T. An advanced fuzzy collaborative intelligence approach for fitting the uncertain unit cost learning process. *Complex Intell. Syst.* **2019**, *5*, 303–313.

-
29. Karsoliya, S. Approximating number of hidden layer neurons in multiple hidden layer BPNN architecture. *Int. J. Eng. Trends Technol.* **2012**, *3*, 714–717.
 30. Moghaddam, A.H.; Moghaddam, M.H.; Esfandyari, M. Stock market index prediction using artificial neural network. *J. Econ. Financ. Adm. Sci.* **2016**, *21*, 89–93.
 31. Jana, G.C.; Swetapadma, A.; Pattnaik, P.K. Enhancing the performance of motor imagery classification to design a robust brain computer interface using feed forward back-propagation neural network. *Ain Shams Eng. J.* **2018**, *9*, 2871–2878.
 32. Šestanović, T.; Arnerić, J. Can recurrent neural networks predict inflation in Euro Zone as good as professional forecasters? *Mathematics* **2021**, *9*, 2486.