fractal and fractional

Article

Fixed-Time Sliding Mode Synchronization of Uncertain Fractional-Order Hyperchaotic Systems by Using a Novel Non-Singleton-Interval Type-2 Probabilistic Fuzzy Neural Network

Ke-Yong Shao, Ao Feng * and Ting-Ting Wang

School of Electrical Information Engineering, Northeast Petroleum University, Daqing 163318, China
* Correspondence: 208002060315@stu.nepu.edu.cn; Tel.: +86-15957119575

Abstract: In this study, we proposed a sliding mode control method based on fixed-time sliding mode surface for the synchronization of uncertain fractional-order hyperchaotic systems. In addition, we proposed a novel self-evolving non-singleton-interval type-2 probabilistic fuzzy neural network (SENSIT2PFNN) to estimate the uncertain dynamics of the system. Moreover, an adaptive compensator was designed to eliminate the influences of random uncertainty and fuzzy uncertainty, thereby yielding an asymptotically stable controlled system. Furthermore, an adaptive law was introduced to optimize the consequence parameters of SENSIT2PFNN. The membership layer and rule base of SENSIT2PFNN were optimized using the self-evolving algorithm and whale optimization algorithm, respectively. The simulation results verified the effectiveness of the proposed methods for the synchronization of uncertain fractional-order hyperchaotic systems.

Keywords: uncertain fractional-order hyperchaotic system; non-singleton interval type-2 probabilistic fuzzy neural network; fixed-time sliding mode controller; whale optimization algorithm; self-evolving

check for updates

Citation: Shao, K.-Y.; Feng, A.; Wang, T.-T. Fixed-Time Sliding Mode Synchronization of Uncertain Fractional-Order Hyperchaotic Systems by Using a Novel Non-Singleton-Interval Type-2 Probabilistic Fuzzy Neural Network. *Fractal Fract.* **2023**, *7*, 247. https://doi.org/10.3390/fractalfract7030247

Academic Editors: Song Zheng, Emad E. Mahmoud, Yangquan Chen and Riccardo Caponetto

Received: 2 February 2023
Revised: 24 February 2023
Accepted: 2 March 2023
Published: 9 March 2023

Copyright: © 2023 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

Fractional calculus was born more than three centuries ago. It is a theory that extends integral calculus to arbitrary order [1,2]. In the two to three centuries following its birth, it was studied as a pure theoretical science with almost no practical application. However, in recent years, increased attention has been paid to the synchronization of fractional-order hyperchaotic systems because of their potential applications in many aspects of science and engineering, such as information processing, biological systems, and chemical science [3,4]. In addition, because fractional-order hyperchaotic systems exhibit uncertainty, unpredictability, and high sensitivity to initial conditions, they are extensively used in engineering fields such as secure communication and encryption. Different synchronization types and control schemes are available, such as projection synchronization [5,6], anti-synchronization [7,8], robust synchronization [9,10], generalized synchronization [11,12], the adaptive control scheme [13,14], and the sliding mode control scheme [15]. The sliding mode control scheme includes the finite-time sliding mode and fixed-time sliding mode. In the finite-time sliding mode, the system state becomes stable within a period depending on the initial values of the system after reaching the sliding surface, whereas the system state becomes stable within a fixed period that depends only on the system parameters after reaching the sliding surface in the fixed-time sliding mode [16,17].

The sliding mode control scheme exhibits good robustness and can effectively suppress uncertain external interferences; however, it is only applicable to deterministic systems with known dynamics. To solve this problem, a general controller can be designed for uncertain systems by combining the fuzzy neural network (FNN) with the approximation characteristics. Type-1 and type-2 FNNs can effectively approximate nonlinear systems, and the synchronization of fractional-order hyperchaotic systems can be realized using fuzzy controllers [18]. A generalized type-2 FNN has been proposed to approximate unknown nonlinear systems, solving the multi-switch synchronization problem encountered in fractional-order hyperchaotic systems [19]. A fuzzy sliding mode control scheme has been proposed to improve the robustness of the unknown time-varying disturbance of fractional-order hyperchaotic systems [20–22]. A type-2 fuzzy disturbance observer has been developed to describe variable-order hyperchaotic systems, and a robust controller has been designed to solve the synchronization problem encountered in fractional-order hyperchaotic systems [23,24]. The aforementioned FNN controller ignores two problems. The first problem is the curse of dimensionality, i.e., the number of rules of the FNN increases exponentially with the increase in input dimensions, which, in turn, greatly increases the system load. To solve this problem, many self-evolving fuzzy systems have been proposed [25–27]. The second problem is random uncertainty. In the synchronization process, due to the interference of the environment, unknown and missing data dimensions, and randomness of the fractional-order hyperchaotic system, various fuzzy and random uncertainties are generated, which greatly affect the performance of the control system [28–30].

Based on the above discussion, to solve the synchronization problem encountered in uncertain fractional-order hyperchaotic systems and avoid the curse of dimension and various uncertainty problems that may be encountered in the synchronization process, in this paper, we proposed a novel self-evolving non-singleton type-2 probabilistic FNN (SENSIT2PFNN) with fixed-time sliding mode control scheme based on the improved whale optimization algorithm (WOA) [31]. The effectiveness of WOA has been verified in fields such as COVID-19 disease detection [32], water demand prediction [33], neural network hyperparameter optimization [34], and multicell production planning [35]. The main contributions of this study are as follows:

1. A novel SENSIT2PFNN was proposed to solve the problems of fuzzy uncertainty and random uncertainty encountered in fuzzy systems. The network structure was modified using the self-evolution algorithm, the rule base was optimized using the improved WOA, and the network adaptive law was used to optimize the network parameters.
2. By using the proposed SENSIT2PFNN to approximate the linear and uncertain nonlinear parts of uncertain fractional-order hyperchaotic systems, a universal fractional-order hyperchaotic synchronization controller was developed.
3. The developed fixed-time sliding mode controller eliminates the influence of approximation error and external interference and realizes the fixed-time synchronization of various uncertain fractional-order hyperchaotic systems.

By using the aforementioned control scheme, the synchronization of the uncertain fractional-order hyperchaotic system was realized. The rest of this article is organized as follows. In Section 2, the problem formulation and preliminary are discussed. In Section 3, the proposed SENSIT2PFNN structure is presented. In Section 4, the proposed fixed-time sliding mode controller and its stability analysis are presented. The synchronization results of uncertain fractional-order hyperchaotic systems are discussed in Section 5. Finally, the conclusion is presented in Section 6.

2. Problem Formulation and Preliminary

Definition 1 ([36]). *The Riemann–Liouville fractional differential with order α of function $f(t)$ can be expressed as follows:*

$$ {}_{t_0}D_t^{\alpha} f(t) = \frac{1}{\Gamma(m-\alpha)} \frac{d^m}{dt} \int_{t_0}^{t} \frac{f(\tau)}{(t-\tau)^{\alpha-m+1}} d\tau, \tag{1} $$

where $m-1 < \alpha \le m$, $m \in N$, and $\Gamma(\cdot)$ is the Gamma function.

Property 1 ([37]). *The following equations are established using the definitions by Riemann–Liouville and Caputo:*

$$ {}_{t_0}^{RL,C}D_t^{\alpha}\left({}_{t_0}^{RL,C}D_t^{-\beta}f(t)\right) = {}_{t_0}^{RL,C}D_t^{\alpha-\beta}f(t), \tag{2}$$

where $\alpha \geq \beta \geq 0$, RL represents the derivative defined by Riemann–Liouville, and C represents the derivative defined by Caputo.

Consider the following system:

$$ \dot{x}(t) = f(x(t)), x(0) = x_0, \tag{3}$$

where $x \in R^n$, and $f : R^n \to R^n$ is a nonlinear function. The system defined using Equation (3) is stable at the origin.

Definition 2 ([38]). *When the system defined using Equation (3) is globally finite-time stable and the convergence time $T(x_0)$ is bounded, the system's origin is called a fixed-time stable equilibrium point.*

Lemma 1 ([38]). *Consider the following system:*

$$ \dot{y} = -\alpha y^{m/n} - \beta y^{p/q}, y(0) = y_0, \tag{4a}$$

where $\alpha, \beta > 0$; m, n, p, and q are positive odd integers and satisfy the conditions $m > n$ and $p < q$. The system's equilibrium point is a fixed-time stable equilibrium point, and the upper bound of the convergence time can be expressed as follows:

$$ T < \frac{1}{\alpha}\frac{n}{m-n} + \frac{1}{\beta}\frac{q}{q-p}. \tag{4b}$$

Lemma 2 ([39]). *For any non-negative real number $\xi_1, \xi_2, \ldots, \xi_n$, the following inequalities exist:*

$$ \begin{cases} \sum\limits_{i=1}^{n} \xi_i^p \geq \left(\sum\limits_{i=1}^{n} \xi_i\right)^p, 0 < p \leq 1 \\ \sum\limits_{i=1}^{n} \xi_i^p \geq n^{1-p}\left(\sum\limits_{i=1}^{n} \xi_i\right)^p, p > 1 \end{cases}. \tag{5}$$

The following n-dimensional uncertain fractional-order hyperchaotic systems are considered:

$$ \text{Drive system}: \begin{cases} D^{\alpha}x_1 = f_1(x,t) + \Delta f_1(x,t) \\ \vdots \\ D^{\alpha}x_i = f_i(x,t) + \Delta f_i(x,t) \\ \vdots \\ D^{\alpha}x_n = f_n(x,t) + \Delta f_n(x,t) \end{cases}, \tag{6}$$

$$ \text{Response system}: \begin{cases} D^{\alpha}y_1 = g_1(y,t) + \Delta g_1(y,t) + d_1^g(t) + u_1(t) \\ \vdots \\ D^{\alpha}y_i = g_i(y,t) + \Delta g_i(y,t) + d_i^g(t) + u_i(t) \\ \vdots \\ D^{\alpha}y_n = g_n(y,t) + \Delta g_n(y,t) + d_n^g(t) + u_n(t) \end{cases}, \tag{7}$$

where Δf_i and Δg_i are uncertain bounded functions; d_i^g is bounded external interference; $d_i^g \leq |\varepsilon_d|$, f_i, and g_i are nonlinear bounded functions; u_i is the control signal; $\boldsymbol{y} = [y_1, y_2, \ldots, y_n]^T$ and $\boldsymbol{x} = [x_1, x_2, \ldots, x_n]^T$ are the state vectors of the response system and drive system, respectively; $0 < \alpha < 1$ is the fractional-order derivative; and $i = 1, \ldots, n$.

The synchronization error is defined as $e_i = y_i - x_i$. The control objective is to design the controller u_i such that $\lim_{t\to\infty} \|e_i\| = 0$.

The i-th subsystem is defined as follows:

$$D^{\alpha} y_i = F_i(y,t) + d_i^g(t) + u_i(t), \tag{8}$$

where $F_i(y,t) = g_i(y,t) + \Delta g_i(y,t)$ as shown in Figure 1, and $(\hat{g}_i)$ is estimated using the proposed NST2PFNN.

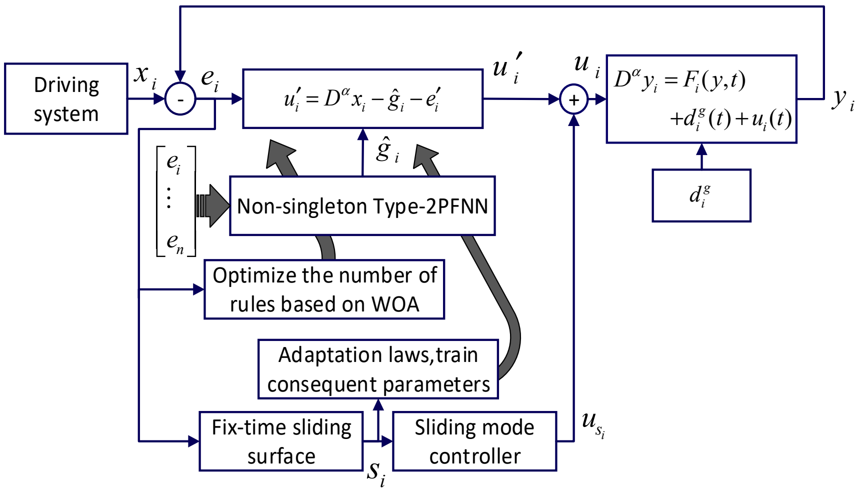

Figure 1. Control block diagram of the i-th subsystem.

Set

$$F_i(y,t) = g_i^* + \varepsilon_i, \tag{9}$$

where ε_i is the estimation error ($|\varepsilon_i| \leq \varepsilon^*$), and g_i^* is the best estimated value ($|g_i^*| \leq \varepsilon_{g^*}$).

The control law is defined as follows:

$$u_i = u'_i + u_{s_i}, \tag{10}$$

$$u'_i = D^{\alpha} x_i - \hat{g}_i - e'_i, \tag{11}$$

where $e'_i = D^{\alpha-1}(\beta_1(1/2)^{m_1/n_1} sig(e_i)^{2m_1/n_1 - 1} + \lambda_1(1/2)^{p_1/q_1} sig(e_i)^{2p_1/q_1 - 1})$; D^{α} is the fractional differential operator with order α, $\beta_1 > 0$, and $\lambda_1 > 0$; m_1, n_1, p_1, and q_1 are positive odd integers with $m_1 > n_1$ and $p_1 < q_1$; and $\hat{g}_i$ is the bounded output of SENSIT2PFNN with $|\hat{g}_i| \leq \varepsilon_{\hat{g}}$.

The i-th error subsystem can be obtained by using Equations (9)–(11):

$$D^{\alpha} e_i = g_i^* + \varepsilon_i - \hat{g}_i - e'_i + u_{s_i} + d_i^g. \tag{12}$$

We aimed to design a fixed-time sliding mode control law (u_{s_i}) such that the error system given by Equation (12) can become stable within a fixed period independent of the initial value.

3. Self-Evolving Non-Singleton Type-2 Probabilistic Fuzzy Neural Network

3.1. Network Structure

To facilitate the self-evolution of the membership layer structure presented, type-2 asymmetric Gaussian functions are employed as the activation functions of all the neural network layers because they offer good flexibility. As shown in Figure 2, the NSIT2PFNN model has six layers:

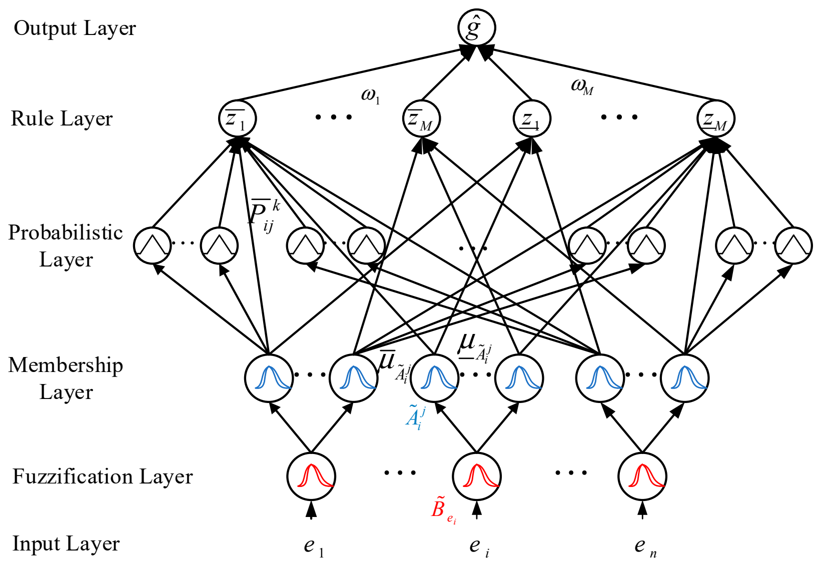

Figure 2. Structure of NSIT2PFNN.

Input layer: The input of this layer is the output of the error system.

Fuzzification layer: In this layer, the single point input of the neural network is fuzzified into a type-2 membership function (MF), representing the uncertainty of the input. Let the type-2 MF generated by the i-th input be $\widetilde{B}_{e_i}$. $\widetilde{B}_{e_i}$ utilizes an asymmetric Gaussian MF with a mean value of e_i and a standard deviation width of $\sigma_{e_i}^L \in [\underline{\sigma}_{e_i}^L, \overline{\sigma}_{e_i}^L]$–$\sigma_{e_i}^R \in [\underline{\sigma}_{e_i}^R, \overline{\sigma}_{e_i}^R]$:

$$
\begin{array}{l}
\overline{\mu}_{\widetilde{B}_{e_i}}(x) = \begin{cases} \exp(-\frac{(x-e_i)^2}{\overline{\sigma}_{e_i}^{L2}}), -\infty < x \le e_i \\ \exp(-\frac{(x-e_i)^2}{\overline{\sigma}_{e_i}^{R2}}), e_i < x \le \infty \end{cases} \\
\underline{\mu}_{\widetilde{B}_{e_i}}(x) = \begin{cases} \exp(-\frac{(x-e_i)^2}{\underline{\sigma}_{e_i}^{L2}}), -\infty < x \le e_i \\ \exp(-\frac{(x-e_i)^2}{\underline{\sigma}_{e_i}^{R2}}), e_i < x \le \infty \end{cases}
\end{array}. \tag{13}
$$

As shown in Figure 3, to obtain the upper and lower membership degrees of the i-th input of the network under the j-th MF, the input of the network must be non-singleton blurred [40]:

$$
\begin{array}{l}
\overline{e}_i^j = \begin{cases} \frac{[(\overline{\sigma}_i^{Lj})^2 e_i + (\overline{\sigma}_{e_i}^R)^2 m_i^j]}{[(\overline{\sigma}_i^{Lj})^2 + (\overline{\sigma}_{e_i}^R)^2]}, e_i \le m_i^j \\ \frac{[(\overline{\sigma}_i^{Rj})^2 e_i + (\overline{\sigma}_{e_i}^L)^2 m_i^j]}{[(\overline{\sigma}_i^{Rj})^2 + (\overline{\sigma}_{e_i}^L)^2]}, e_i > m_i^j \end{cases} \\
\underline{e}_i^j = \begin{cases} \frac{[(\underline{\sigma}_i^{Lj})^2 e_i + (\underline{\sigma}_{e_i}^R)^2 m_i^j]}{[(\underline{\sigma}_i^{Lj})^2 + (\underline{\sigma}_{e_i}^R)^2]}, e_i \le m_i^j \\ \frac{[(\underline{\sigma}_i^{Rj})^2 e_i + (\underline{\sigma}_{e_i}^L)^2 m_i^j]}{[(\underline{\sigma}_i^{Rj})^2 + (\underline{\sigma}_{e_i}^L)^2]}, e_i > m_i^j \end{cases}
\end{array}, \tag{14}
$$

where $j = 1, \ldots, K$; m_i^j σ_i^{Lj}, and σ_i^{Rj} are, respectively, the mean and standard deviations of the j-th type-2 MF$(\widetilde{A}_i^j)$ of the i-th input; and e_i^j is the fuzzy value of the i-th input under the j-th MF.

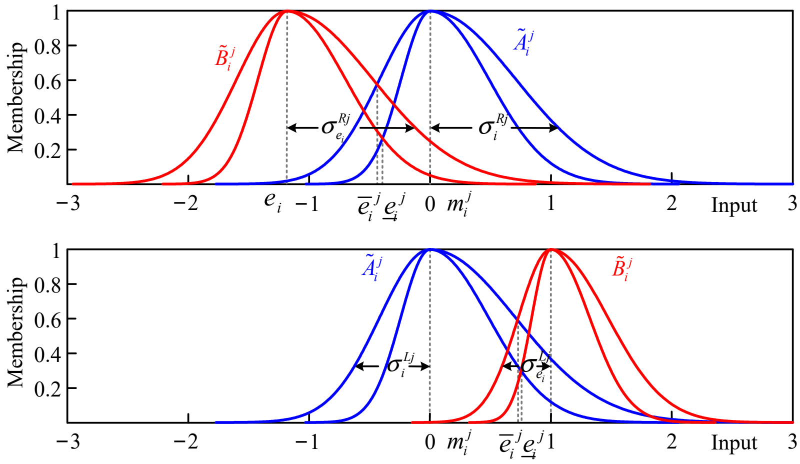

Figure 3. Non-singleton fuzzification.

Membership layer: The membership degree of the i-th input under the j-th MF is calculated in this layer:

$$\begin{aligned}
\overline{\mu}_{\widetilde{A}_i^j}(\overline{e}_i^j) &= \begin{cases} \exp[-\frac{(\overline{e}_i^j - m_i^j)^2}{\sigma_i^{Lj2}}], \overline{e}_i^j \le m_i^j \\ \exp[-\frac{(\overline{e}_i^j - m_i^j)^2}{\sigma_i^{Rj2}}], \overline{e}_i^j > m_i^j \end{cases} \\
\underline{\mu}_{\widetilde{A}_i^j}(\underline{e}_i^j) &= \begin{cases} \exp[-\frac{(\underline{e}_i^j - m_i^j)^2}{\sigma_i^{Lj2}}], \underline{e}_i^j \le m_i^j \\ \exp[-\frac{(\underline{e}_i^j - m_i^j)^2}{\sigma_i^{Rj2}}], \underline{e}_i^j > m_i^j \end{cases}
\end{aligned}, \tag{15}$$

where $\widetilde{A}_i^j$ is the j-th MF of the i-th input; $\overline{\mu}_{\widetilde{A}_i^j}$ and $\underline{\mu}_{\widetilde{A}_i^j}$ are, respectively, the upper and lower membership degrees of the i-th input under the j-th MF; and σ_i^{Lj} and σ_i^{Rj} are, respectively the left and right width of the MF.

Probabilistic layer: In this layer, the Gaussian function is usually employed as the probability function (PF). To improve calculation efficiency, as shown in Figure 4, the triangle function was used in this study:

$$\begin{aligned}
\overline{\mu}_{P_{ij}^k}(\overline{\mu}_{\widetilde{A}_i^j}) &= \begin{cases} 0, \overline{\mu}_{\widetilde{A}_i^j} \ge m_p^k + \sigma_p^k, \overline{\mu}_{\widetilde{A}_i^j} \le m_p^k - \sigma_p^k \\ \frac{\overline{\mu}_{\widetilde{A}_i^j} - m_p^k + \sigma_p^k}{\sigma_p^k}, m_p^k - \sigma_p^k < \overline{\mu}_{\widetilde{A}_i^j} \le m_p^k \\ \frac{-\overline{\mu}_{\widetilde{A}_i^j} + m_p^k + \sigma_p^k}{\sigma_p^k}, m_p^k < \overline{\mu}_{\widetilde{A}_i^j} \le m_p^k + \sigma_p^k \end{cases} \\
\underline{\mu}_{P_{ij}^k}(\overline{\mu}_{\widetilde{A}_i^j}) &= \begin{cases} 0, \underline{\mu}_{\widetilde{A}_i^j} \ge m_p^k + \sigma_p^k, \underline{\mu}_{\widetilde{A}_i^j} \le m_p^k - \sigma_p^k \\ \frac{\underline{\mu}_{\widetilde{A}_i^j} - m_p^k + \sigma_p^k}{\sigma_p^k}, m_p^k - \sigma_p^k < \underline{\mu}_{\widetilde{A}_i^j} \le m_p^k \\ \frac{-\underline{\mu}_{\widetilde{A}_i^j} + m_p^k + \sigma_p^k}{\sigma_p^k}, m_p^k < \underline{\mu}_{\widetilde{A}_i^j} \le m_p^k + \sigma_p^k \end{cases}
\end{aligned}, \tag{16}$$

where P_{ij}^k is the k-th PF of the j-th MF, $\overline{\mu}_{P_{ij}^k}$ and $\underline{\mu}_{P_{ij}^k}$ are the output of the k-th node of the j-th input variable, m_p^k is the center of the triangle, and σ_p^k is the center width of the triangle.

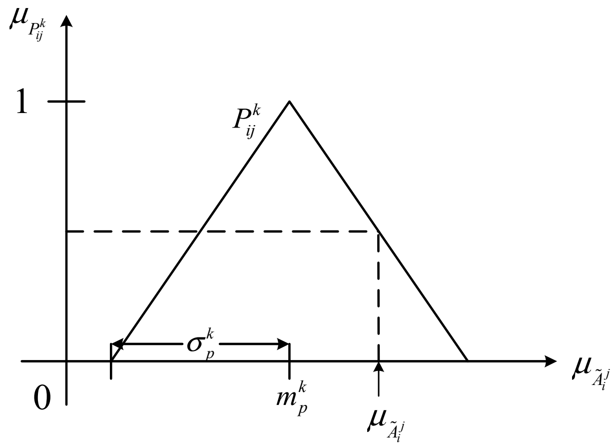

Figure 4. Probability function of NSIT2PFNN.

Rule layer: Each node in this layer represents a fuzzy rule, which calculates the upper and lower firing degrees based on product reasoning. Each rule is defined as follows:

$$Rule^l\text{: IF } e_1 \text{ is } \widetilde{A}_1^{P_1} \text{ and } \dots \text{ and } e_n \text{ is } \widetilde{A}_n^{P_n}\text{, then } \hat{g} \text{ is } \omega^l, \tag{17}$$

where $\widetilde{A}_i^{P_j}$ is the p_j-th type-2 MF of the i-th input, and ω^l is the consequent parameter of the l-th rule. Let the total number of rules be M, and the upper and lower firing degrees of the l-th rule are

$$\begin{array}{c} \overline{z}^l = (\overline{\mu}_{\tilde{A}_i^j} \times \prod\limits_k \overline{\mu}_{P_{ij}^k}) \times \cdots \times (\overline{\mu}_{\tilde{A}_n^K} \times \prod\limits_k \overline{\mu}_{P_{nK}^k}) \\ \underline{z}^l = (\underline{\mu}_{\tilde{A}_i^j} \times \prod\limits_k \underline{\mu}_{P_{ij}^k}) \times \cdots \times (\underline{\mu}_{\tilde{A}_n^K} \times \prod\limits_k \underline{\mu}_{P_{nK}^k}) \end{array}, \tag{18}$$

where $\overline{z}^l$ and $\underline{z}^l$ are, respectively, the upper and lower firing degrees of the l-th rule, and K is the number of MFs per network input.

Output layer: The output of deblurring can be expressed as follows:

$$\hat{g} = \frac{\sum\limits_{i=1}^{M} (\underline{z}^i + \overline{z}^i)\omega^i}{\sum\limits_{i=1}^{M} (\underline{z}^i + \overline{z}^i)}. \tag{19}$$

To facilitate the subsequent derivation process, $\hat{g}$ can be written as follows:

$$\begin{array}{c} H = \frac{1}{\sum\limits_{i=1}^{M} (\underline{z}^i + \overline{z}^i)} \left[\left(\underline{z}^1 + \overline{z}^1 \right) \cdots \left(\underline{z}^M + \overline{z}^M \right) \right]^T \\ W_i = \left[\omega_i{}^1 \cdots \omega_i{}^M \right]^T, \widetilde{W}_i = W_i^* - \hat{W}_i \\ \hat{g}_i = \hat{W}_i{}^T H, g_i^* = W_i^{*T} H \end{array}, \tag{20}$$

where W_i is the weight vector of the i-th neural network, W_i^* is the optimal estimate of the weight, and $\hat{W}_i$ is the weight estimate.

3.2. Self-Evolution Algorithm

(1) Optimizing the rule base of neural networks with the improved WOA.

Due to the curse of dimensionality, an FNN with n inputs and K membership functions for each input generates up to K^n rules, which greatly increases unnecessary calculations. WOA offers the advantages of simple mechanisms, few parameters, and strong optimization ability. To make WOA suitable for the optimization of the NSIT2PFNN rule base, it was modified appropriately in this study; the steps are shown in Figure 5.

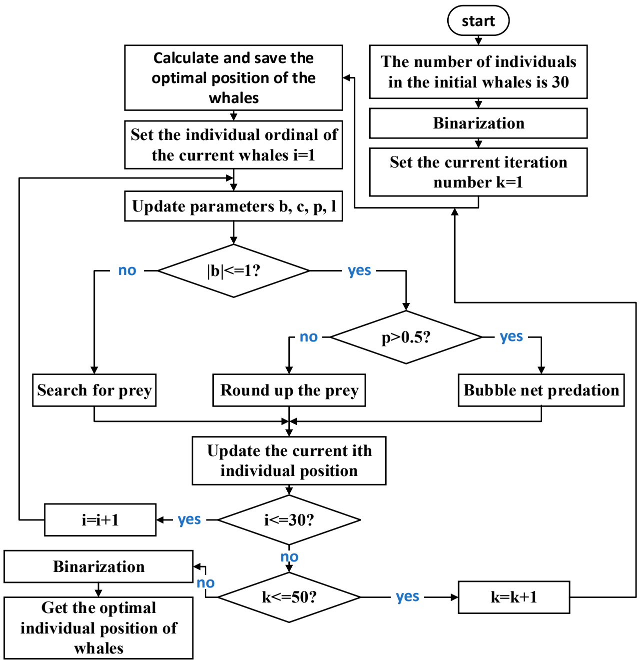

Figure 5. Block diagram of the improved WOA.

First, the total number of rules M of the network was obtained, and the individual position vector of the population with dimension M was randomly generated. Next, the population size was set as 30 and binarized (set as 0 if it was less than 0.5 and as 1 for the rest). Each position was multiplied with the neural network rules to calculate the optimal fitness value for the current optimal individual position. The basic parameters, such as b and p, were initialized such that when $|b| \leq 1$ and $p \leq 0.5$, the individual population updates its position by following the "round up the prey" strategy; when $|b| \leq 1$ and $p > 0.5$, the individual population updates its position by following the "bubble net predation" strategy; and when $|b| > 1$, the individual population updates its position by following the "search for prey" strategy. After a round of position updates was completed, all position information was brought into the neural network rule base, the fitness value was calculated, the optimal individual position was retained, and the cycle was terminated upon reaching the maximum number of iterations. In this paper, the maximum number of iterations was set as 50, and the following fitness function was used:

$$J = \int_0^\infty \sum_{i=1}^{n} |e_i(t)| dt. \tag{21}$$

Remark 1. *In this paper, only two numbers were used in the individual coordinate values of the population to be optimized: 0 and 1. In such cases, the traditional WOA is no longer applicable, and all population individuals and the search space must be binarized after each iteration to generate a new population. The binarization algorithm can be expressed as follows:*

$$a_i(t) = \begin{cases} 1 & round(a_i(t))\%2 = 1 \\ 0 & round(a_i(t))\%2 = 0' \end{cases} \tag{22}$$

where $a_i(t)$ is the i-th coordinate value of an individual in the population.

(2) Self-evolution of the membership layer of NSIT2PFNN

The structure of the membership layer of NSIT2PFNN was optimized by considering the following three type-2 MFs of the i-th input: $\widetilde{A}_i^1(\sigma_i^{L1}, m_i^1, \sigma_i^{R1})$, $\widetilde{A}_i^2(\sigma_i^{L2}, m_i^2, \sigma_i^{R2})$, and

$\widetilde{A}_i^3(\sigma_i^{L3}, m_i^3, \sigma_i^{R3})$, where $\sigma_i^{Lj} \in [\overline{\sigma}_i^{Lj}, \underline{\sigma}_i^{Lj}], j = 1,2,3$ is the left width of the standard deviation of the MF, $m_i^j, j = 1,2,3$ is the mean point of the MF, $\sigma_i^{Rj} \in [\overline{\sigma}_i^{Rj}, \underline{\sigma}_i^{Rj}], j = 1,2,3$ is the right width of the standard deviation of the MF, and the standard deviation width of each input of lower MF is half of that of the upper MF. As shown in Figure 6, for each input e_i, if the maximum upper membership degree is lower than the set value min_Degree (0.2), a new type-2 MF is generated. Its mean point is e_i, and the width of the left (right) standard deviation is the distance from its mean point to the mean point of the adjacent MF:

$$\begin{cases} \overline{\sigma}_{e_i}^{Rj} = m_i^1 - e_i \\ \overline{\sigma}_{e_i}^{Lj} = e_i - m_i^3 \end{cases}. \tag{23}$$

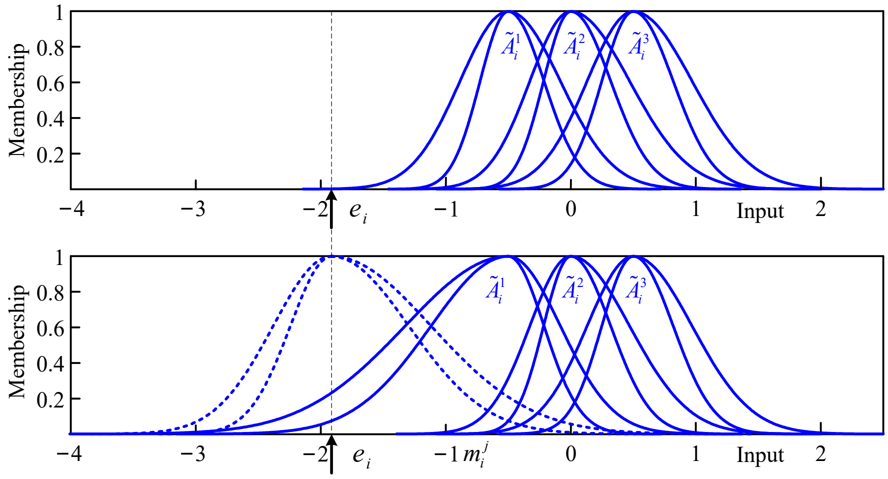

Figure 6. Adding a new MF.

The right (left) standard deviation width of its adjacent MF was changed in the same manner.

If the number of MFs exceeds the set value max_Mf (value 3), the farthest MF is deleted, as shown in Figure 7.

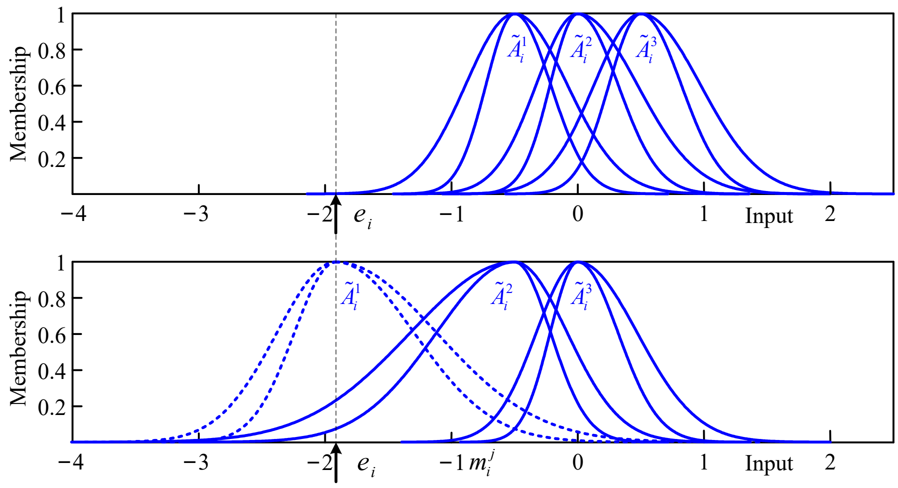

Figure 7. Deleting a redundant MF.

4. Controller Design and Stability Analysis

We designed a fixed-time sliding mode controller u_{s_i} to stabilize the error subsystem given by Equation (12). The sliding surface can be selected as follows:

$$\begin{aligned} s_i &= D^{\alpha-1}e_i + D^{\alpha-2}(\beta_1(1/2)^{m_1/n_1}sig(e_i)^{2m_1/n_1-1} \\ &+\lambda_1(1/2)^{p_1/q_1}sig(e_i)^{2p_1/q_1-1}) \end{aligned}, \tag{24}$$

where s_i is the sliding surface of the i-th subsystem; $\beta_1 > 0$; $\lambda_1 > 0$; m_1, n_1, p_1, and q_1 are positive odd integers such that $m_1 > n_1$ and $p_1 < q_1$; and $sig(\cdot)^x = |\cdot|^x sig(\cdot)$.

When the state variable of the error system reaches the sliding surface, it satisfies the following condition:

$$\begin{aligned} D^{\alpha-1}e_i = -D^{\alpha-2}(\beta_1(1/2)^{m_1/n_1}sig(e_i)^{2m_1/n_1-1} \\ +\lambda_1(1/2)^{p_1/q_1}sig(e_i)^{2p_1/q_1-1}) \end{aligned}. \tag{25}$$

Theorem 1. *The state variable of the error subsystem given by Equation (12) converges to zero within a fixed time T_1 after reaching the sliding surface:*

$$T_1 < \frac{1}{n^{1-m_1/n_1}\beta_1}\frac{n_1}{m_1-n_1} + \frac{1}{\lambda_1}\frac{q_1}{q_1-p_1}. \tag{26}$$

Proof. According to the Lyapunov function:

$$V_1(t) = \frac{1}{2}\sum_{i=1}^{n} e_i^2. \tag{27}$$

According to Property 1, the derivative of the Lyapunov function with respect to time t can be obtained as follows:

$$\begin{aligned} \dot{V}_1 &= \sum_{i=1}^{n} e_i\dot{e}_i \\ &= \sum_{i=1}^{n} e_i(D^{2-\alpha}(D^{\alpha-1}e_i)) \\ &= -\sum_{i=1}^{n} (\beta_1(\tfrac{1}{2}e_i^2)^{m_1/n_1} + \lambda_1(\tfrac{1}{2}e_i^2)^{p_1/q_1}) \end{aligned}. \tag{28}$$

According to Lemma 2:

$$\begin{aligned} \dot{V}_1 &\le -n^{1-m_1/n_1}\beta_1\Big(\tfrac{1}{2}\sum_{i=1}^{n} e_i^2\Big)^{m_1/n_1} - \lambda_1\Big(\tfrac{1}{2}\sum_{i=1}^{n} e_i^2\Big)^{p_1/q_1} \\ &= -n^{1-m_1/n_1}\beta_1 V_1^{m_1/n_1} - \lambda_1 V_1^{p_1/q_1} \end{aligned}. \tag{29}$$

According to Lemma 1, the state variable of the error subsystem converges to zero within a fixed time T_1 after reaching the sliding surface. Thus, Theorem 1 is proved.

The sliding mode control law u_{s_i} can be designed as follows:

$$\begin{aligned} u_{s_i} &= -\beta_2(1/2)^{m_2/n_2}sig(s_i)^{2m_2/n_2-1} \\ &-\lambda_2(1/2)^{p_2/q_2}sig(s_i)^{2p_2/q_2-1} - \eta_i sig(s_i) \end{aligned}. \tag{30}$$

Then:

$$\begin{aligned} u_i &= u'_i + u_{s_i} \\ &= D^{\alpha}x_i - \hat{g}_i - e' - \beta_2(1/2)^{m_2/n_2}sig(s_i)^{2m_2/n_2-1} \\ &-\lambda_2(1/2)^{p_2/q_2}sig(s_i)^{2p_2/q_2-1} - \eta_i sig(s_i) \end{aligned}. \tag{31}$$

□

Theorem 2. *The error subsystem given by Equation (12) converges to the sliding mode surface s_i under the control law (Equation (31)).*

Proof. According to the Lyapunov function:

$$V_2 = \sum_{i=1}^{n} (\frac{1}{2}s_i^2 + \frac{1}{2}\widetilde{W}_i^T \widetilde{W}_i). \tag{32}$$

According to Equation (20), the derivative of the Lyapunov function with respect to time t is obtained as follows:

$$\dot{V}_2 = \sum_{i=1}^{n} (s_i \dot{s}_i - \widetilde{W}_i^T \dot{\hat{W}}_i). \tag{33}$$

Substituting Equations (12), (20), and (24) into Equation (33), we obtain:

$$\begin{aligned} \dot{V}_2 &= \sum_{i=1}^{n} (s_i(D^\alpha e_i + e'_i) - \widetilde{W}_i^T \dot{\hat{W}}_i) \\ &= \sum_{i=1}^{n} (s_i(g_i^* + \varepsilon_i - \hat{g}_i - e'_i + u_{s_i} + d_i^g + e'_i) - \widetilde{W}_i^T \dot{\hat{W}}_i) \\ &= \sum_{i=1}^{n} (s_i(W_i^{*T} H + \varepsilon_i - \hat{W}_i^T H + u_{s_i} + d_i^g) - \widetilde{W}_i^T \dot{\hat{W}}) \end{aligned} . \tag{34}$$

Setting $\dot{\hat{W}} = s_i H$, we obtain:

$$\begin{aligned} \dot{V}_2 &= \sum_{i=1}^{n} s_i(\varepsilon_i - \beta_2 (1/2)^{m_2/n_2} sig(s_i)^{2m_2/n_2 - 1} \\ &\quad -\lambda_2 (1/2)^{p_2/q_2} sig(s_i)^{2p_2/q_2 - 1} - \eta_i sig(s_i) + d_i^g) \\ &= \sum_{i=1}^{n} (s_i \varepsilon_i - \beta_2 (\tfrac{1}{2}s_i^2)^{m_2/n_2} \\ &\quad -\lambda_2 (\tfrac{1}{2}s_i^2)^{p_2/q_2} - \eta |s_i| + s_i d_i^g) \end{aligned} . \tag{35}$$

Setting $\eta_i > \varepsilon^* + \varepsilon_d + \varepsilon_{\hat{g}} + \varepsilon_{g^*}$, we obtain:

$$\dot{V}_2 \le \sum_{i=1}^{n} (-\beta_2 (\frac{1}{2}s_i^2)^{m_2/n_2} - \lambda_2 (\frac{1}{2}s_i^2)^{p_2/q_2}) \le 0. \tag{36}$$

Thus, according to the Lyapunov stability theorem, Theorem 2 is proved. □

Theorem 3. *The error subsystem given by Equation (12) converges to the sliding mode surface s_i within a fixed time t under the action of the control law (Equation (31)):*

$$T_2 < \frac{1}{n^{1-m_2/n_2}\beta_2} \frac{n_2}{m_2 - n_2} + \frac{1}{\lambda_2} \frac{q_2}{q_2 - p_2}. \tag{37}$$

Proof. According to the Lyapunov function:

$$V_3(t) = \frac{1}{2}\sum_{i=1}^{n} s_i^2. \tag{38}$$

Substituting Equations (12) and (24) into Equation (38) and differentiating it, we obtain:

$$\begin{aligned} V_3(t) &= \sum_{i=1}^{n} s_i\dot{s}_i = \sum_{i=1}^{n} s_i(D^\alpha e_i + e'_i) \\ &= \sum_{i=1}^{n} s_i(g_i^* + \varepsilon_i - \hat{g}_i - e'_i + u_{s_i} + d_i^g + e'_i) \\ &= \sum_{i=1}^{n} s_i(g_i^* + \varepsilon_i - \hat{g}_i - \beta_2(1/2)^{m_2/n_2} sig(s_i)^{2m_2/n_2-1} \\ &\quad -\lambda_2(1/2)^{p_2/q_2} sig(s_i)^{2p_2/q_2-1} - \eta_i sig(s_i) + d_i^g) \end{aligned}. \tag{39}$$

Setting $\eta_i > \varepsilon^* + \varepsilon_d + \varepsilon_{\hat{g}} + \varepsilon_{g^*}$, we obtain:

$$V_3(t) \le \sum_{i=1}^{n} (-\beta_2(\frac{1}{2}s_i^2)^{m_2/n_2} - \lambda_2(\frac{1}{2}s_i^2)^{p_2/q_2}). \tag{40}$$

According to Lemma 2:

$$\begin{aligned} \dot{V}_3 &\le -n^{1-m_2/n_2}\beta_2(\tfrac{1}{2}\sum_{i=1}^{n} s_i^2)^{m_2/n_2} - \lambda_2(\tfrac{1}{2}\sum_{i=1}^{n} s_i^2)^{p_2/q_2} \\ &= -n^{1-m_2/n_2}\beta_2 V_3^{m_2/n_2} - \lambda_2 V_3^{p_2/q_2} \end{aligned}. \tag{41}$$

According to Lemma 1, the error subsystem converges to the sliding mode surface s_i within a fixed time T_2 under the action of control law (Equation (31)). Thus, Theorem 3 is proved. □

5. Simulation

We verified the effectiveness of the proposed fractional-order hyperchaotic system synchronization scheme by performing simulation experiments.

As performed in a previous study [41], we used the fractional-order hyperchaotic Chen system as the drive system and the fractional-order hyperchaotic Lorenz system as the response system. The respective formulas are as follows:

$$\text{Drive system}: \begin{cases} D^\alpha x_1 = 35(x_2 - x_1) + x_4 \\ D^\alpha x_2 = 7x_1 + 12x_2 - x_1x_3 \\ D^\alpha x_3 = x_1x_2 - 8x_3 \\ D^\alpha x_4 = x_2x_3 + 0.3x_4 \end{cases}, \tag{42}$$

$$\text{Response system}: \begin{cases} D^\alpha y_1 = 10(y_2 - y_1) + y_4 + \Delta g_1 + d_1^y + u_1 \\ D^\alpha y_2 = 28y_1 - y_2 - y_1y_3 + \Delta g_2 + d_2^y + u_2 \\ D^\alpha y_3 = y_1y_2 - 8/3y_3 + \Delta g_3 + d_3^y + u_3 \\ D^\alpha y_4 = -y_2y_3 - y_4 + \Delta g_4 + d_4^y + u_4 \end{cases}, \tag{43}$$

where the nonlinear unknown function Δg_i and external interference d_i^y are selected as follows:

$$\begin{aligned} \Delta g_1 + d_1^y &= 0.25\cos(6t)y_1 - 0.15\sin(t) \\ \Delta g_2 + d_2^y &= -0.2\cos(2t)y_2 + 0.1\sin(3t) \\ \Delta g_3 + d_3^y &= 0.15\sin(3t)y_3 + 0.2\cos(5t) \\ \Delta g_4 + d_4^y &= -0.2\cos(t)y_4 - 0.15\cos(t) \end{aligned}. \tag{44}$$

The proposed method was used for synchronization control. The parameters of NSIT2PFNN are presented in Table 1. The initial values were set as follows:

$[x_1(0), x_2(0), x_3(0), x_4(0)]$=$[3, 1, 4, -1]$, and $[y_1(0), y_2(0), y_3(0), y_4(0)]$=$[1, 2, 3, 4]$, with the fractional order $\alpha = 0.98$. The control parameters were set as $\eta_i = 8$, $\beta_1 = \beta_2 = 12$, $\lambda_1 = \lambda_2 = 12$, $p_1 = p_2 = 5$, $q_1 = q_2 = 9$, $n_1 = n_2 = 5$, and $m_1 = m_2 = 9$.

Table 1. Initial parameters of NSIT2PFNN.

Initial Parameters of MF									**Initial Parameters of PF**					
A_i^1			A_i^2			A_i^3			P_{ij}^1		P_{ij}^2		P_{ij}^3	
σ_i^{L1}	m_i^1	σ_i^{R1}	σ_i^{L2}	m_i^2	σ_i^{R2}	σ_i^{L3}	m_i^3	σ_i^{R3}	σ_p^1	m_p^1	σ_p^2	m_p^2	σ_p^3	m_p^3
−20	−10	0	−10	0	10	0	10	20	1	0.4	1	0.5	1	0.6

The synchronization results are shown in Figures 8 and 9, and the root mean square error (RMSE) values are presented in Table 2 for comparison. The proposed method was employed for the synchronization of fractional-order hyperchaotic financial systems with different initial values [42], novel fractional-order hyperchaotic systems, and fractional-order hyperchaotic Chen systems [43]. The control and network parameters were the same for all. The RMSE values are presented in Table 2 for comparison.

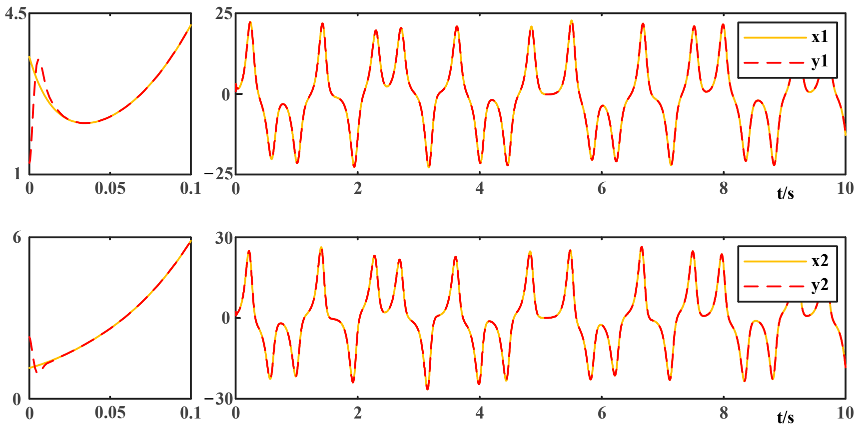

Figure 8. Synchronization results of x1y1 and x2y2.

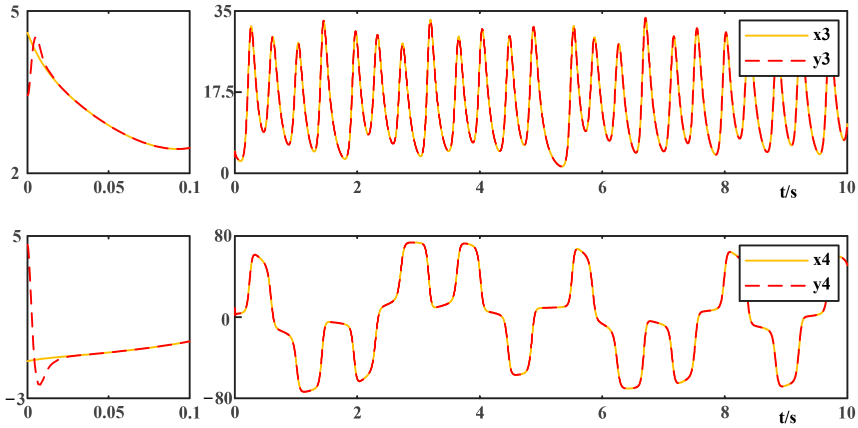

Figure 9. Synchronization results of x3y3 and x4y4.

Table 2. RMSE values of the proposed and other methods proposed in the literature [41–43].

	RMSE			
	e_1	e_2	e_3	e_4
Proposed method	0.1261	0.0625	0.0629	0.3173
The method of [41]	1.2365	2.1173	1.8924	3.6215
Proposed method	0.0247	0.0615	0.0460	0.0305
The method of [42]	2.5281	2.3682	1.9546	1.5640
Proposed method	0.0471	0.0480	0.0294	0.0705
The method of [43]	0.0457	0.0907	0.0895	0.0902

As shown in Figures 8 and 9 and Table 2, the proposed method achieved full synchronization of the uncertain fractional-order hyperchaotic system and yielded lower RMSE, higher control accuracy, and better control effect than the three synchronization schemes.

6. Conclusions

In this study, to address the synchronization problem encountered in uncertain fractional-order hyperchaotic systems, a novel self-evolving NSIT2PFNN model was proposed to estimate nonlinear functions in system dynamics. In the proposed model, the structure of the neural network is not fixed, and the structure of the membership layer and the number of fuzzy rules can be adjusted adaptively.

In addition, a fixed-time sliding mode controller based on the FNN was designed to eliminate the approximation error and external interference, and three synchronization simulation experiments using different fractional-order hyperchaotic systems were performed.

The simulation results demonstrated that the proposed controller could achieve a good control effect when there are various uncertainties in the dynamic process of the system. In future studies, we will extend the proposed method to other uncertain nonlinear systems.

Author Contributions: Conceptualization, A.F.; methodology, A.F.; software, A.F.; validation, K.-Y.S. and A.F.; formal analysis, A.F.; investigation, A.F.; resources, A.F. and K.-Y.S.; data curation, A.F.; writing—original draft preparation, A.F.; writing—review and editing, A.F.; visualization, A.F.; supervision, K.-Y.S.; project administration, A.F. and K.-Y.S.; funding acquisition, T.-T.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported in part by the National Natural Science Foundation of China (grant 52074088) and in part by the Northeast Petroleum University Provincial Outstanding Youth Reserve Talent Project (grant SJQH202002).

Data Availability Statement: The data that support the findings of this study are available from the corresponding author upon reasonable request.

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

References

1. Grzesikiewicz, W.; Wakulicz, A.; Zbiciak, A. Non-linear problems of fractional calculus in modeling of mechanical system. *Int. J. Mech. Sci.* **2013**, *70*, 90–98. [CrossRef]
2. Machado, J. Fractional calculus: Application in modeling and control. In *Integral Methods in Science and Engineering: Progress in Numerical and Analytic Techniques*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 279–295.
3. Lazarevic, M. Elements of mathematical phenomenology of self-organization nonlinear dynamical systems: Synergetics and fractional calculus approach. *Int. J. Non-Linear Mech.* **2015**, *73*, 31–42. [CrossRef]
4. Machado, J.; Kiryakova, V.; Mainardi, F. Recent history of fractional calculus. *Commun. Nonlinear Sci. Numer. Simul.* **2011**, *16*, 1140–1153. [CrossRef]
5. Yang, S.; Hu, C.; Yu, J. Projective synchronization in finite-time for fully quaternion-valued memristive networks with fractional-order. *Chaos Solitons Fractals* **2021**, *147*, 110911. [CrossRef]
6. Zhang, W.; Cao, J.; Wu, R. Lag projective synchronization of fractional-order delayed chaotic systems. *J. Frankl. Inst.* **2019**, *356*, 1522–1534. [CrossRef]
7. Peng, R.; Jiang, C.; Guo, R. Partial anti-synchronization of the fractional-order chaotic systems through dynamic feedback control. *Mathematics* **2021**, *9*, 718. [CrossRef]
8. Mahmoud, E.E.; Higazy, M.; Alotaibi, H. Quaternion anti-synchronization of a novel realizable fractional chaotic model. *Chaos Solitons Fractals* **2021**, *144*, 110715. [CrossRef]
9. Mohammadzadeh, A.; Ghaemi, S. Robust synchronization of uncertain fractional-order chaotic systems with time-varying delay. *Nonlinear Dyn.* **2018**, *93*, 1809–1821. [CrossRef]
10. Kavikumar, R.; Sakthivel, R.; Kwon, O.M. Robust tracking control design for fractional-order interval type-2 fuzzy systems. *Nonlinear Dyn.* **2022**, *107*, 3611–3628. [CrossRef]
11. He, S.; Sun, K.; Wang, H. Generalized synchronization of fractional-order hyperchaotic systems and its DSP implementation. *Nonlinear Dyn.* **2018**, *92*, 85–96. [CrossRef]
12. Martínez, F.O.; Montesinos-García, J.J.; Gómez-Aguilar, J.F. Generalized synchronization of commensurate fractional-order chaotic systems: Applications in secure information transmission. *Digit. Signal Process.* **2022**, *126*, 103494. [CrossRef]
13. Wang, R.M.; Zhang, Y.N.; Chen, Y.Q. Fuzzy neural network-based chaos synchronization for a class of fractional-order chaotic systems: An adaptive sliding mode control approach. *Nonlinear Dyn.* **2020**, *100*, 1275–1287. [CrossRef]

14. Li, R.G.; Wu, H.N. Adaptive synchronization control with optimization policy for fractional-order chaotic systems between 0 and 1 and its application in secret communication. *ISA Trans.* **2019**, *92*, 35–48. [CrossRef] [PubMed]
15. Javan, A.A.K.; Jafari, M.; Shoeibi, A. Medical images encryption based on adaptive-robust multi-mode synchronization of chen hyper-chaotic systems. *Sensors* **2021**, *21*, 3925. [CrossRef]
16. Modiri, A.; Mobayen, S. Adaptive terminal sliding mode control scheme for synchronization of fractional-order uncertain chaotic systems. *ISA Trans.* **2020**, *105*, 33–50. [CrossRef]
17. Deepika, D.; Kaur, S.; Narayan, S. Uncertainty and disturbance estimator based robust synchronization for a class of uncertain fractional chaotic system via fractional order sliding mode control. *Chaos Solitons Fractals* **2018**, *115*, 196–203. [CrossRef]
18. Huynh, T.T.; Lin, C.M.; Le, T.L. Wavelet interval type-2 fuzzy quad-function-link brain emotional control algorithm for the synchronization of 3D nonlinear chaotic systems. *Int. J. Fuzzy Syst.* **2020**, *22*, 2546–2564. [CrossRef]
19. Sabzalian, M.H.; Mohammadzadeh, A.; Zhang, W. General type-2 fuzzy multi-switching synchronization of fractional-order chaotic systems. *Eng. Appl. Artif. Intell.* **2021**, *100*, 104163. [CrossRef]
20. Jahanshahi, H.; Yousefpour, A.; Munoz-Pacheco, J.M. A new multi-stable fractional-order four-dimensional system with self-excited and hidden chaotic attractors: Dynamic analysis and adaptive synchronization using a novel fuzzy adaptive sliding mode control method. *Appl. Soft Comput.* **2020**, *87*, 105943. [CrossRef]
21. Atan, Ö.; Kutlu, F.; Castillo, O. Intuitionistic fuzzy sliding controller for uncertain hyperchaotic synchronization. *Int. J. Fuzzy Syst.* **2020**, *22*, 1430–1443. [CrossRef]
22. Zhang, X.; Huang, W. Robust H∞ adaptive output feedback sliding mode control for interval type-2 fuzzy fractional-order systems with actuator faults. *Nonlinear Dyn.* **2021**, *104*, 537–550. [CrossRef]
23. Li, J.F.; Jahanshahi, H.; Kacar, S. On the variable-order fractional memristor oscillator: Data security applications and synchronization using a type-2 fuzzy disturbance observer-based robust control. *Chaos Solitons Fractals* **2021**, *145*, 110681. [CrossRef]
24. Moradi, Z.M.; Shoja-Majidabad, S. Chaos synchronization using an improved type-2 fuzzy wavelet neural network with application to secure communication. *J. Vib. Control.* **2022**, *28*, 2074–2090. [CrossRef]
25. Mohammadzadeh, A.; Ghaemi, S. Synchronization of uncertain fractional-order hyperchaotic systems by using a new self-evolving non-singleton type-2 fuzzy neural network and its application to secure communication. *Nonlinear Dyn.* **2017**, *88*, 1–19. [CrossRef]
26. Mohammadzadeh, A.; Ghaemi, S.; Kaynak, O. Robust H∞ Based Synchronization of the Fractional-Order Chaotic Systems by Using New Self-Evolving Nonsingleton Type-2 Fuzzy Neural Networks. *IEEE Trans. Fuzzy Syst.* **2016**, *24*, 1544–1554. [CrossRef]
27. Wang, H.; Luo, C.; Wang, X. Synchronization and identification of nonlinear systems by using a novel self-evolving interval type-2 fuzzy LSTM-neural network. *Eng. Appl. Artif. Intell.* **2019**, *81*, 79–93. [CrossRef]
28. Lin, F.J.; Chen, C.I.; Xiao, G.D. Voltage stabilization control for microgrid with asymmetric membership function-based wavelet Petri fuzzy neural network. *IEEE Trans. Smart Grid* **2021**, *12*, 3731–3741. [CrossRef]
29. Fuhg, J.N.; Kalogeris, I.; Fau, A. Interval and fuzzy physics-informed neural networks for uncertain fields. *Probabilistic Eng. Mech.* **2022**, *68*, 103240. [CrossRef]
30. Gridach, M. A framework based on (probabilistic) soft logic and neural network for NLP. *Appl. Soft Comput.* **2020**, *93*, 106232. [CrossRef]
31. Mirjalili, S.; Lewis, A. The whale optimization algorithm. *Adv. Eng. Softw.* **2016**, *95*, 51–67. [CrossRef]
32. Nadimi-Shahraki, M.H.; Zamani, H.; Mirjalili, S. Enhanced whale optimization algorithm for medical feature selection: A COVID-19 case study. *Comput. Biol. Med.* **2022**, *148*, 105858. [CrossRef]
33. Guo, W.; Liu, T.; Dai, F. An improved whale optimization algorithm for forecasting water resources demand. *Appl. Soft Comput.* **2020**, *86*, 105925. [CrossRef]
34. Chen, H.; Li, W.; Yang, X. A whale optimization algorithm with chaos mechanism based on quasi-opposition for global optimization problems. *Expert Syst. Appl.* **2020**, *158*, 113612. [CrossRef]
35. Kalananda, V.K.R.A.; Komanapalli, V.L.N. A combinatorial social group whale optimization algorithm for numerical and engineering optimization problems. *Appl. Soft Comput.* **2021**, *99*, 106903. [CrossRef]
36. Yousefpour, A.; Jahanshahi, H.; Munoz-Pacheco, J.M. A fractional-order hyper-chaotic economic system with transient chaos. *Chaos Solitons Fractals* **2020**, *130*, 109400. [CrossRef]
37. Laarem, G. A new 4-D hyper chaotic system generated from the 3-D Rösslor chaotic system, dynamical analysis, chaos stabilization via an optimized linear feedback control, it's fractional order model and chaos synchronization using optimized fractional order sliding mode control. *Chaos Solitons Fractals* **2021**, *152*, 111437.
38. Ni, J.; Liu, L.; Liu, C. Fractional order fixed-time nonsingular terminal sliding mode synchronization and control of fractional order chaotic systems. *Nonlinear Dyn.* **2017**, *89*, 2065–2083. [CrossRef]
39. Akgül, A.; Rajagopal, K.; Durdu, A. A simple fractional-order chaotic system based on memristor and memcapacitor and its synchronization application. *Chaos Solitons Fractals* **2021**, *152*, 111306. [CrossRef]
40. Mohammadzadeh, A.; Ghaemi, S.; Kaynak, O. Robust predictive synchronization of uncertain fractional-order time-delayed chaotic systems. *Soft Comput.* **2019**, *23*, 6883–6898. [CrossRef]
41. Shao, K.Y.; Xu, Z.H.; Wang, T.T. Robust finite-time sliding mode synchronization of fractional-order hyper-chaotic systems based on adaptive neural network and disturbances observer. *Int. J. Dyn. Control.* **2021**, *9*, 541–549. [CrossRef]

42. Wang, Y.L.; Jahanshahi, H.; Bekiros, S. Deep recurrent neural networks with finite-time terminal sliding mode control for a chaotic fractional-order financial system with market confidence. *Chaos Solitons Fractals* **2021**, *146*, 110881. [CrossRef]
43. Sabzalian, M.H.; Mohammadzadeh, A.; Lin, S. Robust fuzzy control for fractional-order systems with estimated fraction-order. *Nonlinear Dyn.* **2019**, *98*, 2375–2385. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.