


ITC 1/50 Information Technology and Control Vol. 50 / No. 1 / 2021 pp. 76-88 DOI 10.5755/j01.itc.50.1.25412	Application of Fuzzy Pushdown Automaton on Prediction of Quality Control for Spinning Yarn	
	Received 2020/03/03	Accepted after revision 2021/02/17
	 http://dx.doi.org/10.5755/j01.itc.50.1.25412	

HOW TO CITE: Wu, Q., Wang, X., Wang, Y., Chen, Z., Chen, H., Sun, D., Han, Z. (2021). Application of Fuzzy Pushdown Automaton on Prediction of Quality Control for Spinning Yarn. *Information Technology and Control*, 50(1), 76-88. <https://doi.org/10.5755/j01.itc.50.1.25412>

Application of Fuzzy Pushdown Automaton on Prediction of Quality Control for Spinning Yarn

QingE Wu, Xing Wang, Yonghua Wang, Zhiwu Chen

School of Electrical and Information Engineering, Zhengzhou University of Light Industry, Zhengzhou, 450002, P. R. China; e-mails: wqe969699@163.com

Hu Chen

School of Electrical and Information Engineering, Zhengzhou University of Light Industry, Zhengzhou, 450002, P. R. China; e-mails: wqe969699@163.com

Henan Qingyao Information Technology Co., Ltd., Zhengzhou, 450000, P. R. China

Dong Sun, Zhenyu Han

Engineering Training Center, Zhengzhou University of Light Industry, Zhengzhou, 450002, P. R. China

Corresponding authors: wqe969699@163.com, wyh@zzuli.edu.cn, sundyfirst@126.com

In order to perform better recognition, tracking and control for fuzzy and uncertain thing, this paper will design a suitable fuzzy pushdown automaton (FPDA) control method to solve the problem. Firstly, the control design structure of FPDA and the decision reasoning rules in control are given. Secondly, the application of FPDA in prediction of quality control for spinning yarn is discussed in the practical problem. Finally, the comparison of FPDA and other control methods on the target control is given. The simulation results show that the control speed and the average precision of designed FPDA are faster by 12ms and higher by 4.98% than that of traditional method, which its control precision is 96.87%.

KEYWORDS: Fuzzy pushdown automaton (FPDA), Decision reasoning rule, Quality control, Prediction, Spinning yarn.

1. Introduction

In the aircraft fault monitoring and diagnosis, when the aircraft fails due to the different structure of the aircraft components, the fault frequency of the different components included in the vibration signal is distributed in different frequency ranges. If the aircraft hides an early weak defect of a component, its defect information is overwhelmed by the running vibration signals and random noise of other components. If the root cause of the failure cannot be discovered early, it will cause immeasurable economic losses. It is vital to be able to trouble shoot early and timely.

Previously, Blanco et al. [3] discussed fuzzy knowledge equivalence representations between neural networks, fuzzy systems and models of automata. Sharma et al. [17] studied the concepts of homomorphism, fuzzy multiset transformation semigroup and coverings of fuzzy multiset finite automata, and established their basic properties for the algebraic study of fuzzy multiset automata theory. From a control point of view, fuzzy finite-state automata (FFA) with recurrent neural networks [11] for often imitating fuzzy dynamical systems were very useful. Gulistan et al. [7] discussed a relationship on FFA between the generalized cubic soft ideals and characteristic functions and cubic level sets, as well as two different lattice structures in semigroups. Previously, works had been shown how FFA can be mapped into recurrent neural networks with second-order weights using a crisp representation of FFA states [14].

A variety of fuzzy automata can solve the above problems, so the derivation of different automata is critical. Therefore, many researchers used neural networks to give the derivation of automata. Pan et al. [15] proposed the notion of nondeterministic fuzzy automata with membership values in a complete residuated lattice L , called lattice-valued nondeterministic fuzzy automata (L-NFAs). Sunita and Jiwari [6] generalized and extended the concept of transitive closure for the conversion of fuzzy automata into fuzzy regular expressions. For a fuzzy automaton M , where r was a fuzzy regular expression obtained using the proposed approach, $L(M)=L(r)$. Li and Wang [12] introduced the notion of universal fuzzy automaton with membership values in a complete residuated lattice, whose states were the factorizations of this fuzzy language and transition function was defined

using the inclusion degree of related fuzzy languages, and defined the homomorphism of fuzzy automata. Moreover, this literature gave a method to construct the universal fuzzy automaton by a deterministic fuzzy automaton accepting the given fuzzy language, which was effective in the case that this deterministic fuzzy automaton is finite. Mendivil [8] dealt with the application of Brzozowski's minimization procedure to fuzzy finite automata with truth-values in a complete residuated (zero-divisor-free) lattice. Although neural networks are quite different from fuzzy systems, their functional forms were similar [16]. Fuzzy reasoning had been widely used in the fields of control systems and intelligence [10]. Using fuzzy sets and using closed intervals to represent uncertain data had similar functions [4, 13]. Therefore, in the early 1980s, the Polish school proposed the interval analysis method as a tool to be used together with the fuzzy set method [18]. Since the combination of fuzzy set method [21] and interval analysis had a better effect, the concept of interval value fuzzy set was proposed and used for fuzzy reasoning [2].

On the basis of interval-valued fuzzy relations, Wei and Li [19] studied two forms of reasoning of simple interval-valued fuzzy reasoning and multi-interval-valued fuzzy reasoning, but did not considered the case of deterministic factors or weights in the fuzzy reasoning. The ordered weighted mean operator theory given by Yager [20] was a good example for this idea.

Pan et al. [15] constructed a nondeterministic fuzzy automaton with membership values in complete residuated lattice. Feng et al. [5] gave an EDAS Method for Extended Hesitant Fuzzy Linguistic Multi-criteria Decision Making. Adamatzky [1] discussed an interplay between excitability and geometry by using back propagation neural network model, which had higher prediction accuracy.

The characteristics of many objective things were uncertain and ambiguous [9]. In order to better identify, track and control the fuzzy and uncertain objective things in the machine simulation method, this paper proposes a fuzzy pushdown automata (FPDA) method to deal with such problems. On the basis of the theory of automata, control design structure, decision rules in control and inference rules are given for FPDA. Then its application is discussed in practical

problems. Through actual examples, control methods and steps for FPDA, as well as tracking and control to targets are grasped.

2. Design of FPDA

2.1. FPDA

Definition 1: FPDA is an eight-tuple $M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F, \mu)$, where Q represents a finite state set of controller; Σ represents a finite set of input character; Γ represents a finite set of push stack character; $q_0 \in Q$ represents an original state; $Z_0 \in \Gamma$ represents an original stack character; $F \subseteq Q$ represents a fuzzy set of final state; μ represents a membership degree set of transition function. $\delta: Q \times (\Sigma \cup \{\varepsilon\}) \times \Gamma \rightarrow Q \times \Gamma^*$ represents a mapping on a finite subset, that is, $\delta(q_i, a, z, \mu) = \{(q_j, \alpha)\}$, where $q_i, q_j \in Q; z, \alpha \in \Gamma, a \in \Sigma$, which it means: at the current state q_i and the stack top z , when the character a is input, the automaton arrives at the state q_j with a certain degree of membership μ , and at this time the stack top character z is replaced by α , i.e., the stack top character is α now. At the same time, the pointer moves to the right for one grid.

2.2. Design of FPDA

FPDA control is a free model for establishing a non-linear control algorithm by using fuzzy logic, especially in systems, where traditional quantitative techniques are used to analyze overly complex processes

or provide information that is qualitative, inaccurate, and non-deterministic.

The FPDA designed in this paper consists of four modules: fuzzification, inferential decision making, precision and top-of-stack band. Its control structure is shown in Figure 1.

At sampling time t , the error and its change of error are defined as follows:

$$e_k = y_r - y_k. \quad (1)$$

$$\Delta e_k = e_k - e_{k-1}. \quad (2)$$

(1) Fuzzification

The fuzzification module mainly performs the following task. It converts the measurement of the input represented by the digital into the ordinal number of a certain limited code usually expressed by the language value. Each limited code represents a fuzzy subset within the universal set and is determined by its membership function.

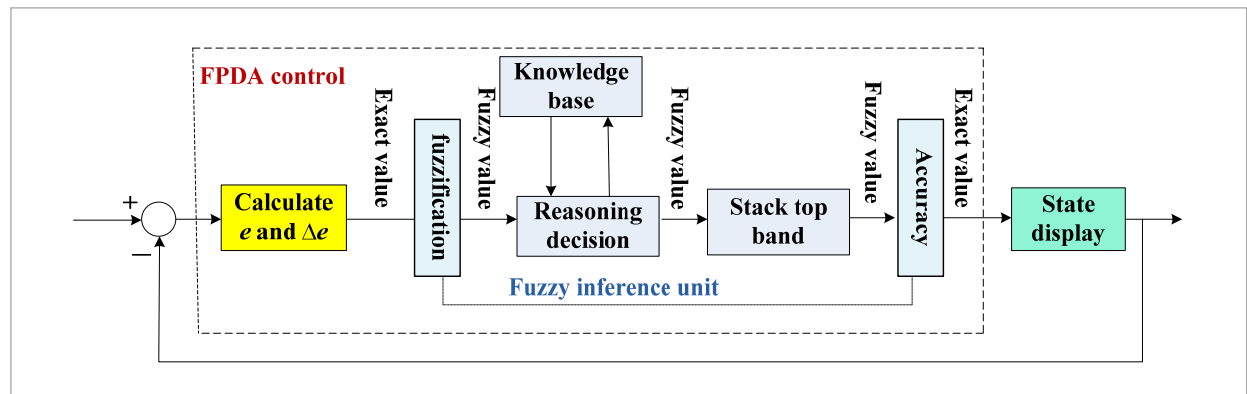
In here, the membership function used is a normal function as follows:

$$f(t) = ae^{-\left(\frac{t-t_a}{\tau}\right)^2}. \quad (3)$$

where a and τ all represent the coefficient of shape; t_a represents center point of definition domain. These parameters can be calculated by the histogram of dataset through using the least squares algorithm.

Figure 1

The structure of the FPDA control system



(2) Reasoning Decision

Fuzzy reasoning is a thought process that leads a new judgment or proposition from one or more known judgments or propositions according to certain principles. In general, reasoning consists of two parts of judgment. One is a known judgment as a starting point of reasoning and it is called the premise. Second is a new judgment induced by the proposition and is called conclusion.

According to the characteristics of FPDA processing fuzzy signals, the following inference rules are given.

Premise: If A and B , then C .

Conclusion: If A' and B' , then $C'=?$.

The inference step is given as follows:

step 1: First solve $D = A \times B$, assume $d_{xy} = \mu_A(x) \wedge \mu_B(y)$, then the matrix D can be obtained as follows:

$$D = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1n} \\ d_{21} & d_{22} & \cdots & d_{2n} \\ \vdots & \vdots & & \vdots \\ d_{m1} & d_{m2} & \cdots & d_{mn} \end{bmatrix}. \quad (4)$$

step 2: Let D be written as a column vector DT , i.e.,

$$DT = [d_{11}, d_{12}, \dots, d_{1n}, d_{21}, \dots, d_{mn}]^T, \quad (5)$$

step 3: Solve a relationship matrix R , i.e., $R = DT \times C$.

step 4: D' can be obtained by A' and B' , i.e., $D' = A' \times B'$.

step 5: According to step 2. Convert D' to row vector DT' .

step 6: Finally, the fuzzy inference output C' is obtained, i.e., $C' = DT' \circ R$, where \circ is a synthesis operation.

(3) Accuracy

The result of reasoning through FPDA is a fuzzy set. In actual use, especially in fuzzy control, there must be a certain value to control or drive the executive agency. The accurate process is defined by an operating process taking an accurate value in the inferred fuzzy set which it can best represent the likelihood of this FPDA inference result. Many different approaches can be taken to achieve the accuracy. The results obtained by different methods are also different. The method of precision calculation given here is the maximum membership method.

$$\mu_c(u^*) = \max_{u \in U} \{\mu_c(u)\}, \quad (6)$$

where c represents a fuzzy set; $\mu_c(u^*) = \max_{u \in U} \{\mu_c(u)\}$ represents maximum degree of membership for all objects discussed.

(4) Stack Top Band of FPDA

The control of fuzzy push-down automata with the stack top band module has two great advantages. First, when it successfully processes a signal, it can make a mark on the top of the stack, so we can clearly know how many signals can be successfully dealt with by it and how many the length of signal processing is. Second, during signal processing, when it stops working, it can automatically save its processed nodes, and then it can continue processing from the last disconnected node at the next time, instead of starting from original. It not only saves a lot of time and space for processing signals, but also saves a lot of energy.

FPDA successfully processes fuzzy signals in two forms. The one is done in the final state. The other is done in the empty stack state.

$$L(FPDA) = \left\{ (\omega, \mu) \left| \omega \in \Sigma^*, \mu = \bigvee_{\alpha} \mu', (q_0, \omega, z_0) \xrightarrow{\mu'^*} (q, \varepsilon, \alpha), \exists q \in F, \forall \alpha \in \Gamma^* \right. \right\}. \quad (7)$$

$$L(FPDA) = \left\{ (\omega, \mu) \left| \omega \in \Sigma^*, \mu = \bigvee_q \mu', (q_0, \omega, z_0) \xrightarrow{\mu'^*} (q, \varepsilon, \varepsilon), \forall q \in Q \right. \right\}. \quad (8)$$

If a fuzzy signal L is successfully processed by the fuzzy pushdown automaton, the necessary and sufficient condition is that there is a FPDA, so that $L=L(FPDA)$.

(5) Processing Method of FPDA to Fuzzy Signal

A fuzzy signal can be processed or identified by a FDFA, that is, this refers to how it recognizes or handles its fuzzy language.

Theorem 1. A fuzzy language is recognizable by a FPDA, then its necessary and sufficient condition is that the fuzzy language must be of type 2, i.e., a context-free fuzzy language.

This theorem is also introduced as follows:

For a fuzzy language L , the following two conditions are equivalent.

(i) L is recognizable by a FPDA.

(ii) L is of type 2, i.e., a context-free fuzzy language.

Proof: (i) \Rightarrow (ii) Consider the assertion that (i) implies (ii). Every type-2 fuzzy language is generated by a fuzzy grammar $G = (N, T, S, P, V)$.

The non-terminal alphabet N of the type-2 analytic fuzzy grammar consists of a corresponding finite set of the states Q . The terminal alphabet T of the type-2 analytic fuzzy grammar consists of a corresponding finite set of the input alphabet Σ and the pushdown stack alphabet Γ of FPDA. The initial state q_0 of FPDA is suitable for the initial alphabet S of the type-2 fuzzy grammar. The fuzzy degree of membership set V of the type-2 fuzzy grammar is the same as that of FPDA. A fuzzy set of the final states and a start stack letter of FPDA are suitable for some part non-terminal alphabets and a terminal alphabet of the type-2 fuzzy grammars respectively, these non-terminal alphabets and a terminal alphabet conclude in the generation formulas that produce finally any fuzzy language. The fuzzy generation formulas set $q_i \xrightarrow{\mu} q_j a$ of the type-2 analytic fuzzy grammar in P that recognize each fuzzy language (7) or (8) consist of the fuzzy transitions $\delta(q_i, a, z, \mu) = \{(q_j, \alpha)\}$ of FPDA, and the additional set generation formula $S \xrightarrow{\mu} \varepsilon$ of the type-2 fuzzy grammar in P consists of the fuzzy transition $\delta(S, \varepsilon, z, \mu) = \{(S, \alpha)\}$ of FPDA, where $q_i, q_j \in Q$; $S = q_0, Z, \alpha \in \Gamma, a \in \Sigma, \mu \in V$.

Assume $(\omega, \mu) \in L(FPDA)$. A fuzzy language (ω, μ) recognized by a FPDA is equal to the union of all

fuzzy languages of the form (7) or (8). The fuzzy transitions of FPDA are the $\delta(q_0, \omega, z_0, \mu^\alpha) = \{(q, \alpha)\}$ in the final state or $\delta(q_0, \omega, z_0, \mu^q) = \{(q, \varepsilon)\}$ in the empty stack for recognizing the language (ω, μ) , and then the corresponding generation formula of G is $q_0 \xrightarrow{Z_0/\alpha, \mu^{\alpha,*}} \omega q$ or $q_0 \xrightarrow{Z_0/\varepsilon, \mu^q,*} \omega q$, where $\mu = \bigvee_{\alpha} \mu^\alpha$ or $\mu = \bigvee_q \mu^q, \forall \mu^\alpha, \mu^q \in V; q_0 = S; \omega \in \Sigma^* \text{ or } \omega \in T^*; \exists q \in F \text{ or } \forall q \in Q; \forall \alpha \in \Gamma^*$. So, there exists $(\omega, \mu) \in L(G)$. Thus, we conclude that the $L(FPDA)$ is of type-2.

(ii) \Rightarrow (i) To show that (ii) implies (i), we consider a fuzzy language L recognized by a type-2 analytic fuzzy grammar. $G = (N, T, S, P, V)$.

Now, we prove the equality of the following formula.

$$L(FPDA) = L(G). \quad (9)$$

The input alphabet Σ and pushdown stack alphabet Γ of FPDA consist of the terminal alphabet T of G . The set of states Q of FPDA consist of the non-terminal alphabet N of G . The initial state q_0 is the initial alphabet S of type-2 fuzzy grammar. A start stack letter Z_0 is one of the terminal alphabets of G . The final state set F of FPDA consists of some subsets of N , which contain the letter S . The fuzzy degree of membership set V of FPDA is the same as that of the type-2 fuzzy grammar. Finally, the transition function δ of FPDA is defined as follows:

For each pair (p, a) , there is the transition $\delta(q, a, z, \mu) = \{(p, \alpha)\}$, where $p, q \in Q, Z \in \Gamma, \alpha \in \Gamma^*, a \in \Sigma, \mu \in V$. We now prove that (9) holds true.

Assume $(\omega, \mu) \in L(G)$, where $\omega = a_1 a_2 \cdots a_n$ and for each $a_i \in T$. According to G , then there is a derivation of:

$$S \xrightarrow{\mu_1} A_1 \xrightarrow{\mu_2} a_1 A_2 \xrightarrow{\bar{\mu}^*} a_1 a_2 \cdots a_{n-1} A_n \xrightarrow{\mu_n} a_1 a_2 \cdots a_n, \quad (10)$$

$$\mu = \bigvee_{S \Rightarrow \omega} (\mu_1 \wedge \mu_2 \wedge \bar{\mu} \wedge \mu_n),$$

where $A_i \in (N \cup T)^* (i = 1, 2, \dots, n)$.

By the above transition function δ of FPDA defined, there is:

$$\begin{aligned}
\delta(q_0, \omega, Z_0, \mu) &= \delta(q_0, a_1 a_2 \cdots a_n, Z_0, \mu) = \delta_2(\delta_1(q_0, a_1, Z_0, \mu_1), a_2 \cdots a_n, \mu_2) =, \\
\delta_3(\delta_2(q_1, a_2, Z_1, \mu_2), a_3 \cdots a_n, \mu_3) &= \delta_n^*(\delta_{n-1}(q_{n-2}, a_{n-1}, Z_{n-2}, \bar{\mu}), a_n, \mu_n) =, \\
\delta_n(\delta_{n-1}(q_{n-2}, a_{n-1}, Z_{n-2}, \bar{\mu}), a_n, \mu_n) &=, \\
\delta_n(q_{n-1}, a_n, Z_{n-1}, \mu_n) &= \{(q_n, Z_n)\}, \\
\mu &= \bigvee_{\delta(q_0, \omega, Z_0, \bullet) = \{(q_n, Z_n)\}}^* (\mu_1 \wedge \mu_2 \wedge \bar{\mu} \wedge \mu_n),
\end{aligned} \tag{11}$$

where \bullet denotes the degree of membership, Z_n is a stack top alphabet of FPDA in a final state or in an empty stack, q_n contains S and is a final state. Since the fuzzy transition $\delta(q_j, a, Z_j, \mu) = \{(q_j, \alpha)\}$ of FPDA is equally powerful as the fuzzy productions $q_i \xrightarrow{Z_i/\alpha, \mu} a q_j$ of the type-2 analytic fuzzy grammar, there is $(\omega, \mu) \in L(FPDA)$.

On the other hand, a derivation of the form (10) can be inferred from (11). Consequently, (9) holds true and we have shown that (ii) implies (i). Q.E.D.

3. Prediction Control Process by FPDA for Quality of Spinning Yarn

In the middle processing layer of FPDA, there are output indicators such as mightiness, neps, yarn levelness, hairiness and single strength. Data processing is to perform first for removing data that does not meet with production requirements. According to the principle of mathematical statistics, 70 groups are selected as training samples and 10 groups are used as prediction data. Then build and evaluate the prediction model whether it is feasible and optimal or not. If the predicted data cannot meet with the production requirements, then the model needs to be corrected. The prediction control model built by FPDA is discussed as follows:

3.1. Normalization of Data

In the normalization layer of FPDA, data processing is first step to remove data that does not satisfied the production requirements. Normalize the data between the interval $[0, 1]$ to eliminate the effects of the original indicator units. The normalization model is given as follows:

$$x_i = \frac{x - x_{\min}}{x_{\max} - x_{\min}}, \tag{12}$$

where x_i represents the normalized data; x represents the unprocessed data; x_{\min} represents the minimum value of unprocessed data; and x_{\max} represents the maximum value of unprocessed data.

3.2. Establishment of Model

A mapping relationship established R from U to V can be determined by transforming from input vector to output vector, where, $U = \{u_1, u_2, \dots, u_n\}$ is the input, and $V = \{v_1, v_2, \dots, v_m\}$ is the output. The mapping matrix R is from U to V can be determined as shown in the formula (13).

$$U \circ V = R, \tag{13}$$

where \circ represents a synthesis operator, u_i ($i = 1, 2, \dots, n$) represents the value of each input parameter, and v_j ($j = 1, 2, \dots, m$) represents the output.

3.3. Evaluation of Model

Assume $U = \{u_1, u_2, \dots, u_n\}$ has n influencing factors and $V = \{v_1, v_2, \dots, v_m\}$ has m evaluation results. For u_i , an assessment can be made separately $R(u_i)$. The assessment method is as follows: the model relationship R can be obtained from U to V for u_i . The above mapping is the evaluative model.

Evaluation model matrix is: a single factor assessment $\{r_{i1}, r_{i2}, \dots, r_{im}\}$ is established by the model for each factor u_i . The overall evaluation matrix $R = (r_{ij})_{n \times m}$ can be obtained by the synthesis between multiple single factors through the state transition of FPDA. Then, a weighted average model

$b_j = \sum (a_i \cdot r_{ij}) (1 \leq i \leq n, j = 1, 2, \dots, m)$ can be obtained, where a_i is the weight. This model can balance all factors. Assume $c_{ij} (i = 1, 2, \dots, n; j = 1, 2, \dots, m)$ denotes the value of membership of the y^{th} output indicator and the x^{th} input factor, then the factor r_{ij} in R can be obtained by the following formula.

$$r_{ij} = \frac{c_{ij}}{\sum_{j=1}^m c_{ij}}, \quad (14)$$

The method of determining the weight can also use the frequency statistics method to determine the weight of each factor, which is given as follows:

- a For each factor u_j , the weight a_{ij} is given by k experts. The maximum value M_j and the minimum value m_j of weights are calculated by the following formula, respectively.

$$M_j = \max \{a_{ij} | 1 \leq i \leq k; j = 1, 2, \dots, m\}. \quad (15)$$

$$m_j = \min \{a_{ij} | 1 \leq i \leq k; j = 1, 2, \dots, m\}. \quad (16)$$

- b The appropriate positive integer p is selected, and the weight a_{ij} corresponding to the factor u_j is divided into p groups from small to large. The group distance S is calculated by the following formula:

$$S = \frac{M_j - m_j}{p}. \quad (17)$$

- c Calculating the frequency of the weights that fall within each group.
d Taking the value (or the adjacent value) in the group, where the maximum frequency is located as the weight of the factor U .

3.4. Calibration of Model

(1) Correction of mapping matrix

Getting the model relationship R from U to V , and assuming

$$R = \begin{pmatrix} r_{11} & r_{12} & \dots & r_{1m} \\ r_{21} & r_{22} & \dots & r_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ r_{n1} & r_{n2} & \dots & r_{nm} \end{pmatrix}.$$

Then

$$\begin{pmatrix} \omega_{11}x_{11} & \omega_{12}x_{12} & \dots & \omega_{1n}x_{1n} \\ \omega_{21}x_{21} & \omega_{22}x_{22} & \dots & \omega_{2n}x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \omega_{i1}x_{i1} & \omega_{i2}x_{i2} & \dots & \omega_{in}x_{in} \end{pmatrix} \circ \begin{pmatrix} r_{11} & r_{12} & \dots & r_{1m} \\ r_{21} & r_{22} & \dots & r_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ r_{n1} & r_{n2} & \dots & r_{nm} \end{pmatrix} = \begin{pmatrix} y_{11} & y_{12} & \dots & y_{1m} \\ y_{21} & y_{22} & \dots & y_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ y_{i1} & y_{i2} & \dots & y_{im} \end{pmatrix}, \quad (18)$$

where, $x_{i1}, x_{i2}, \dots, x_{in}$ represents a n -dimensional vector that represents the value of Micronaire for original cotton, the neps of original cotton, the impurity rate of raw cotton and the strength of original cotton, etc. $y_{i1}, y_{i2}, \dots, y_{im}$ represents a m -dimensional vector, which represents the output factors of mightiness, neps, yarnlevelness, hairiness and single strength, etc. The relationship matrix R is the established by the input-output relationship matrix. In addition, r represents the elements in the matrix. ω_{ij} represents the weights, and ω_{ij} meets with the following formula:

$$\omega_{i1} + \omega_{i2} + \dots + \omega_{in} = 1. \quad (19)$$

It can be known from formula (19) that the corrected mapping matrix R is achieved by correcting the weights. Moreover, the above formula (19) can be solved by using the following iterative formula.

$$\omega_n = \omega_{n-1} + \mu(\omega_n - \omega_{n-1}), \quad (20)$$

where, ω_n represents the current weight; ω_{n-1} represents the weight calculated from the previous calculation; ω_0 is the initial weight and it is necessary to obtain some empirical values based on sample training; μ represents the learning rate. The mapping relationship R is gradually adjusted by FPDA learning through a limited number of iterations.

3.5. Experiment and Results Analysis

Figures 2-Figure 6 show the actual value and predicted value that is the strength, hairiness, neps, single strength, and yarn levelness. As shown in Figure 2, the error between the actual value and the predicted val-

Figure 2

Comparison between the actual and predictive values on strength

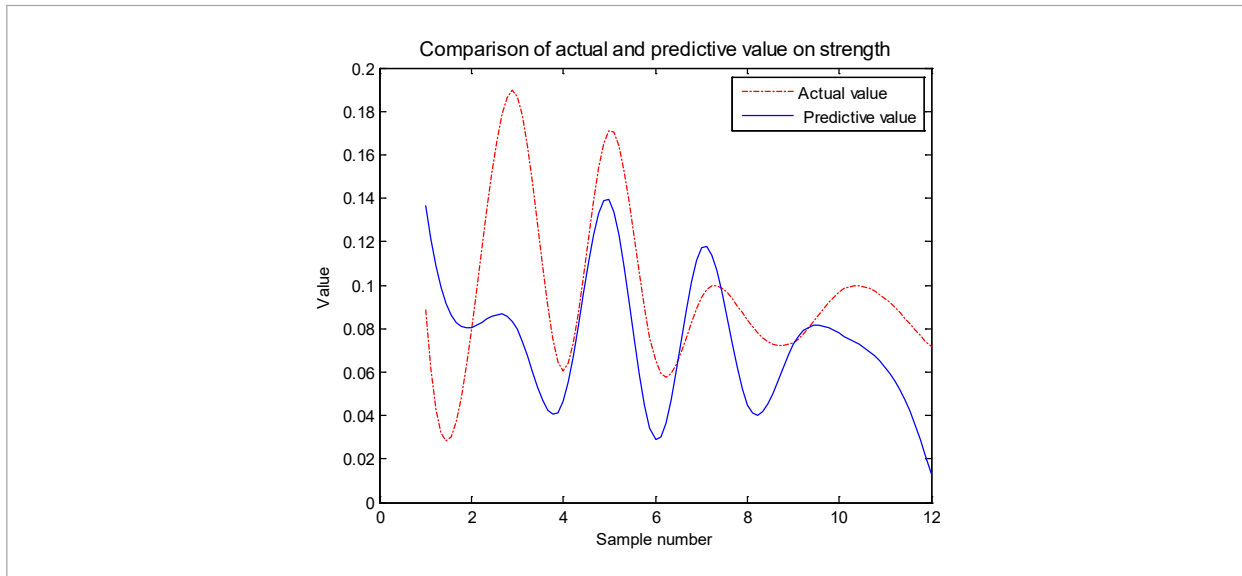
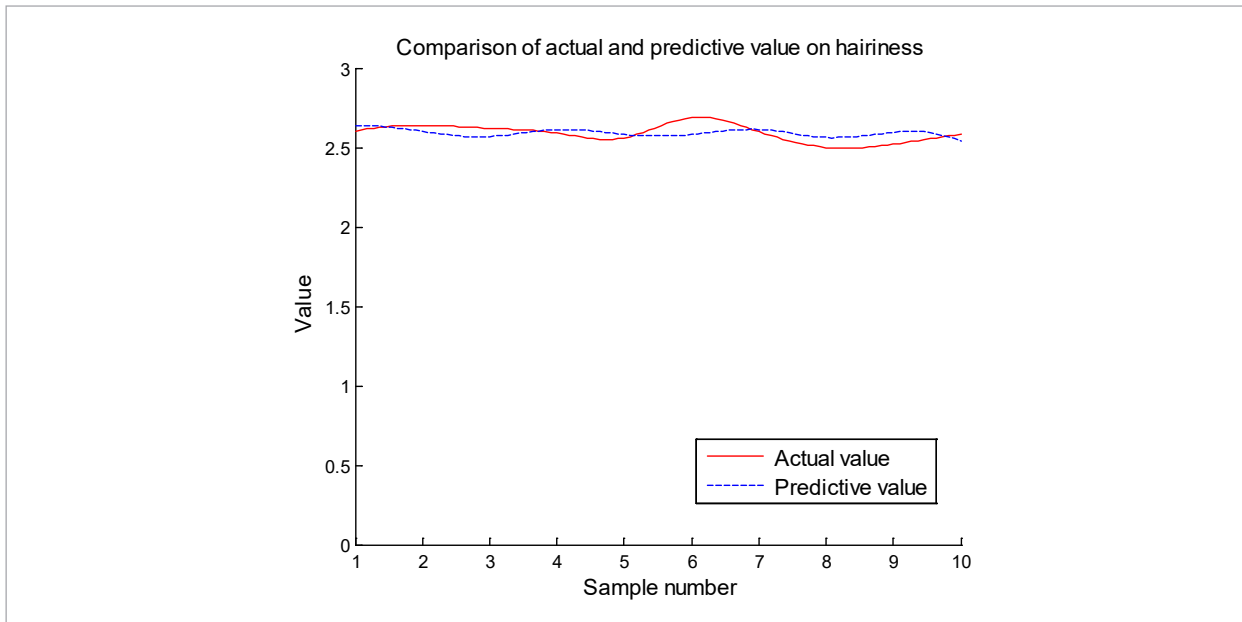


Figure 3

Comparison between the actual and predictive values on hairiness



ue for the strength is small, and the trend of change is the same. Because the value of hairiness is small, the magnitude of the change is not very large, and the actual value is close to the trend of the predicted value,

as shown in Figure 3. As shown in Figure 4, the predicted value of the neps is consistent with the trend of the true value, even if the true variation is sometimes large. As shown in Figure 5, although the predicted

Figure 4

Comparison between the actual and predictive values on neps

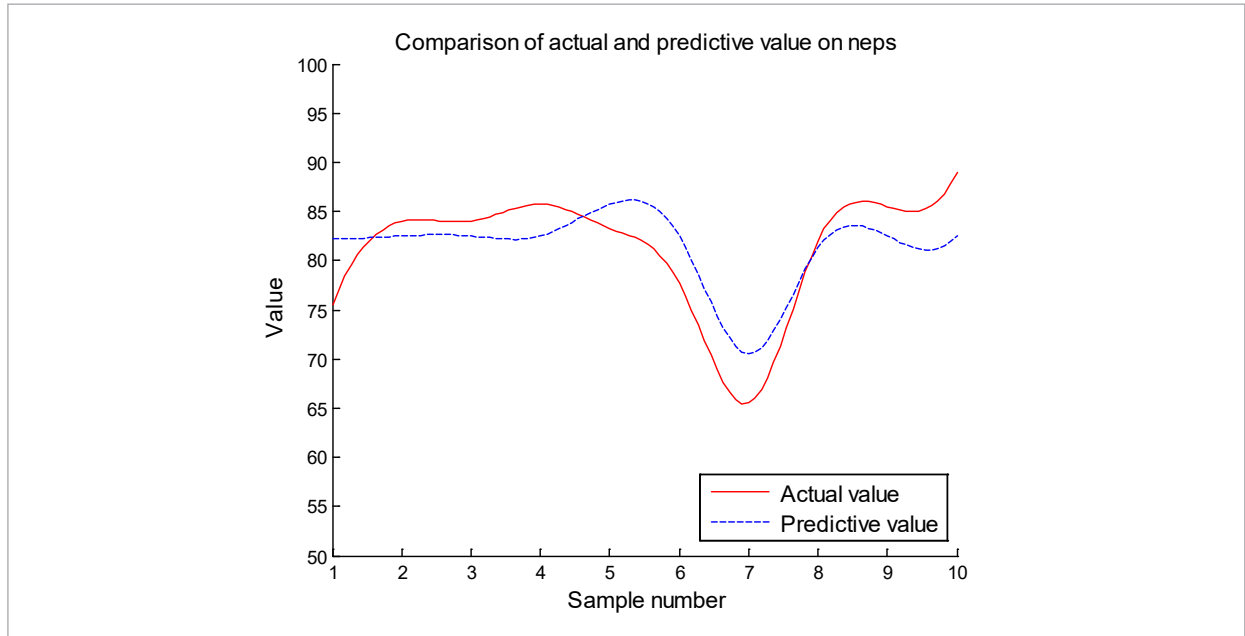


Figure 5

Comparison between the actual and predictive values on single strength

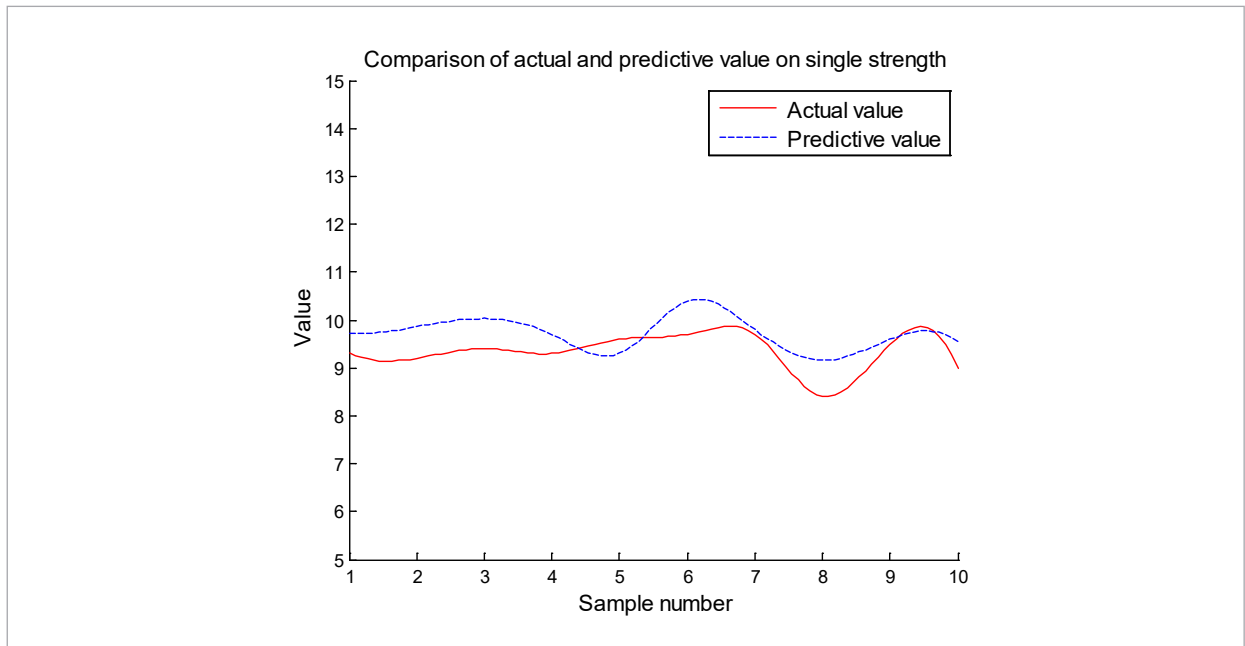
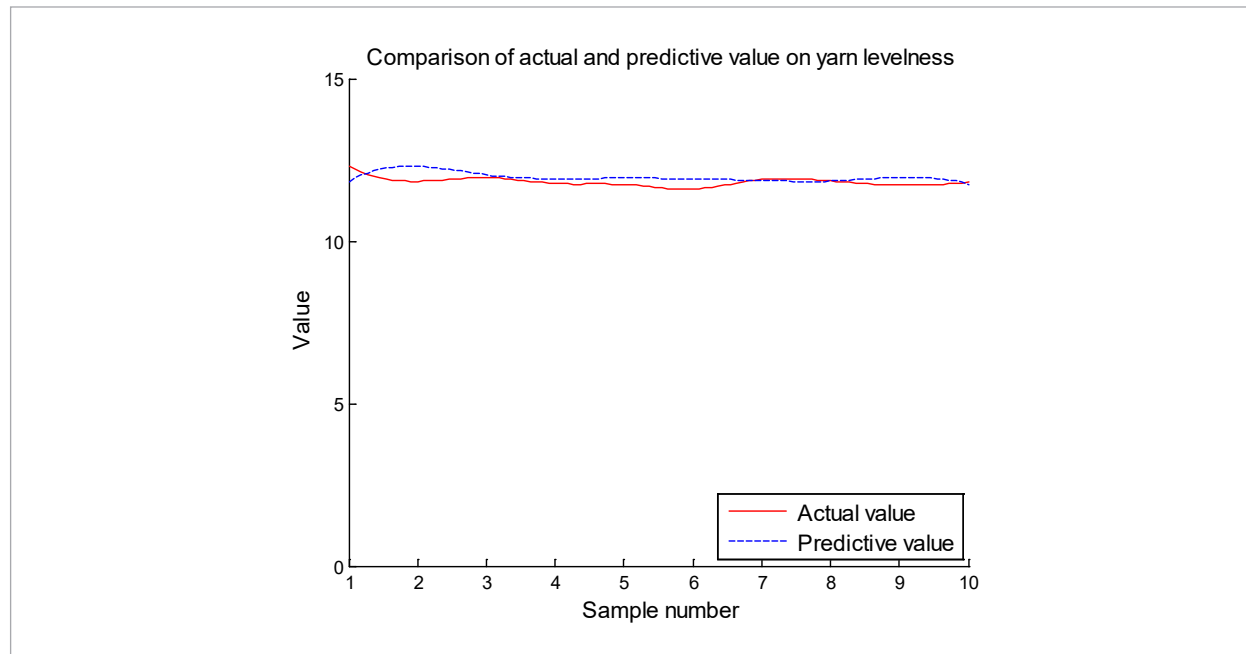


Figure 6

Comparison between the actual and predictive values on yarn levelness



value of the single strength is somewhat different from the actual value, the absolute error is less than 8%. As shown in Figure 6, the predicted value is close to the trend of the actual value of yarn levelness. According to the production test of the textile group, the cost is reduced by 25%, the benefit is increased by 20%, and the productivity of the enterprise is greatly improved.

4. Comparison of FPDA and Existing Control Method in Target Tracking

To simplify the procedure, in the simulation, FPDA control and traditional PID control [4, 19] are compared to control three targets. Let the target do a uniform linear motion and a uniform left and right turning motion, and the deflection rate is $\omega_1 = 10^\circ$ or $\omega_1 = -10^\circ$. The control weights of FPDA to three targets are $a_1 = (0.6, 0.4, 0.3, 0.2)$, $a_2 = (0.2, 0.4, 0.6, 0.8)$ and $a_3 = (0.1, 0.5, 0.2, 0.4)$, respectively. The control matrix is given as follows:

$$\Psi = \begin{bmatrix} 1 - e^{-\omega_2 t} & 1 - e^{-\omega_2 (t+1)} & \dots \\ e^{-\omega_2 t} & e^{-\omega_2 (t+1)} & \dots \\ e^{\omega_3 t} & e^{\omega_3 (t+1)} & \dots \end{bmatrix}, \quad (19)$$

where $\omega_2 = 8^\circ$, $\omega_3 = -9^\circ$, t represents a time variable. Ψ represents a $3 \times T_0$ matrix. T_0 denotes the simulation sampling and $T_0 = 200$. The sampling rate is 2 seconds. The simulation results are shown in Figure 7 and Figure 8.

From Figure 7, the designed FPDA control method in this paper has better tracking control than the traditional controller to the target. The tracking curve of the FPDA control method to target is almost close to the actual orbit of the target. From Figure 8, the tracking error of the FPDA control method gradually decreases and tends to a stable value.

Compare with existing tracking control method of a traditional controller [4, 19] such as PID controller, that of FPDA controller not only has faster processing speed, lower occupancy of storage space, but also

Figure 7

Comparison of FPDA and existing control method for target tracking control

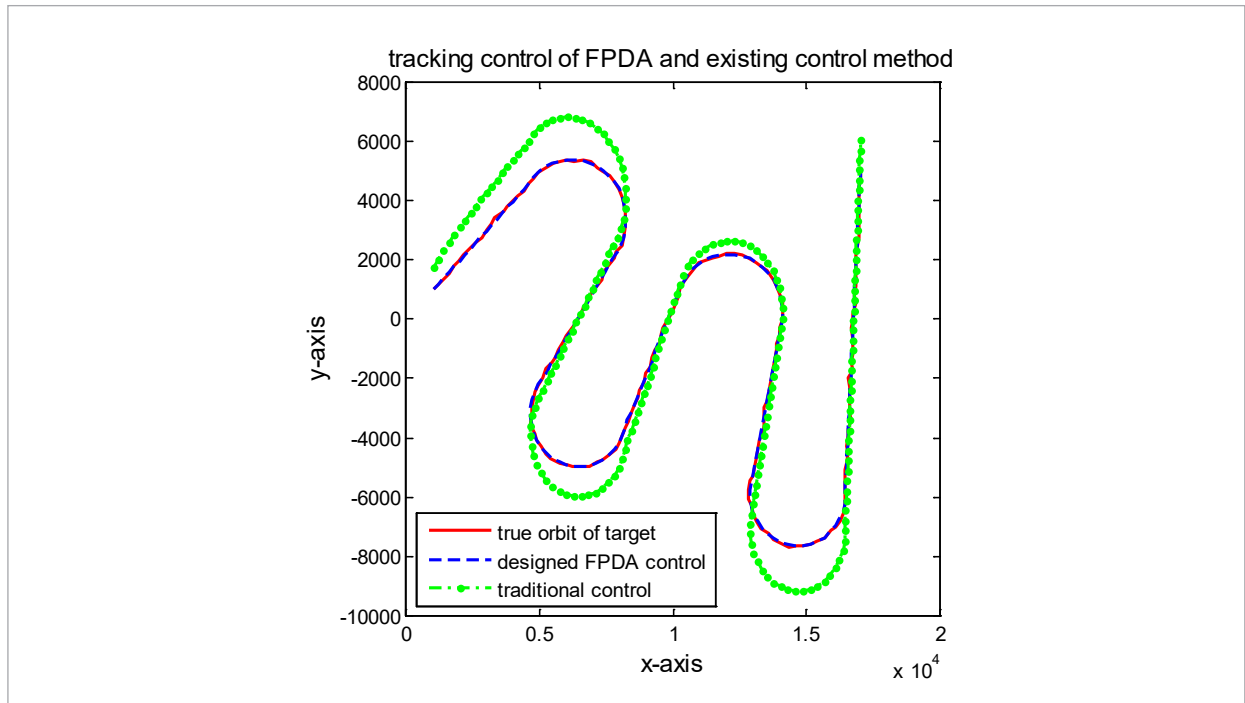


Figure 8

Comparison of FPDA and existing control method for control error of target

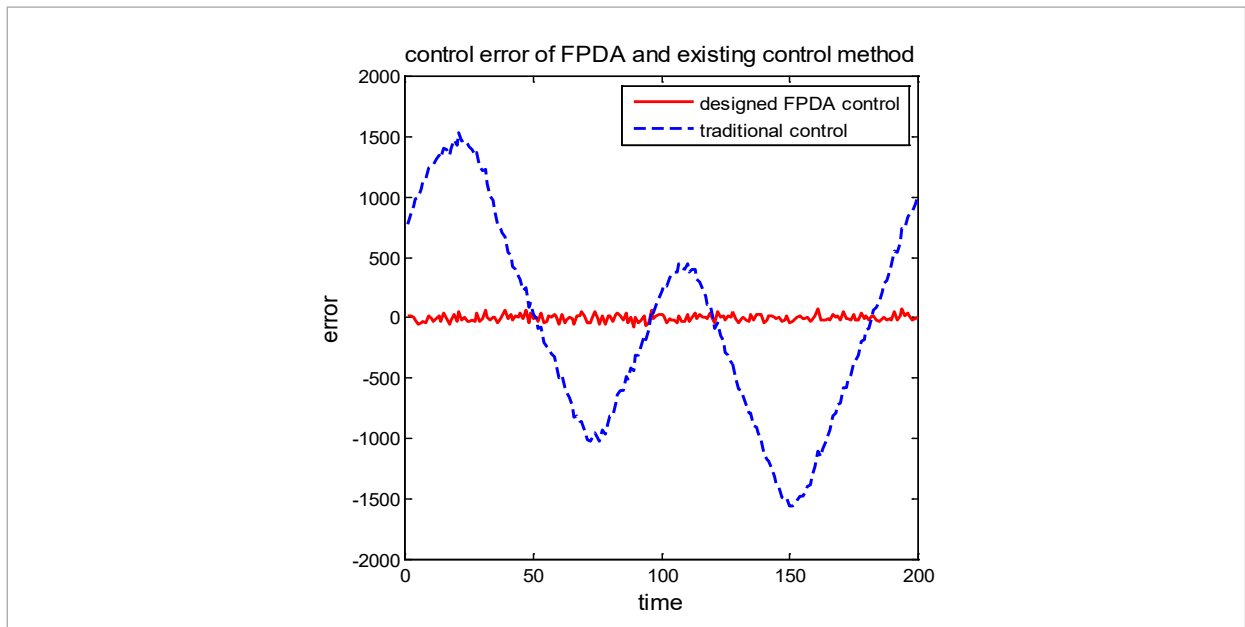
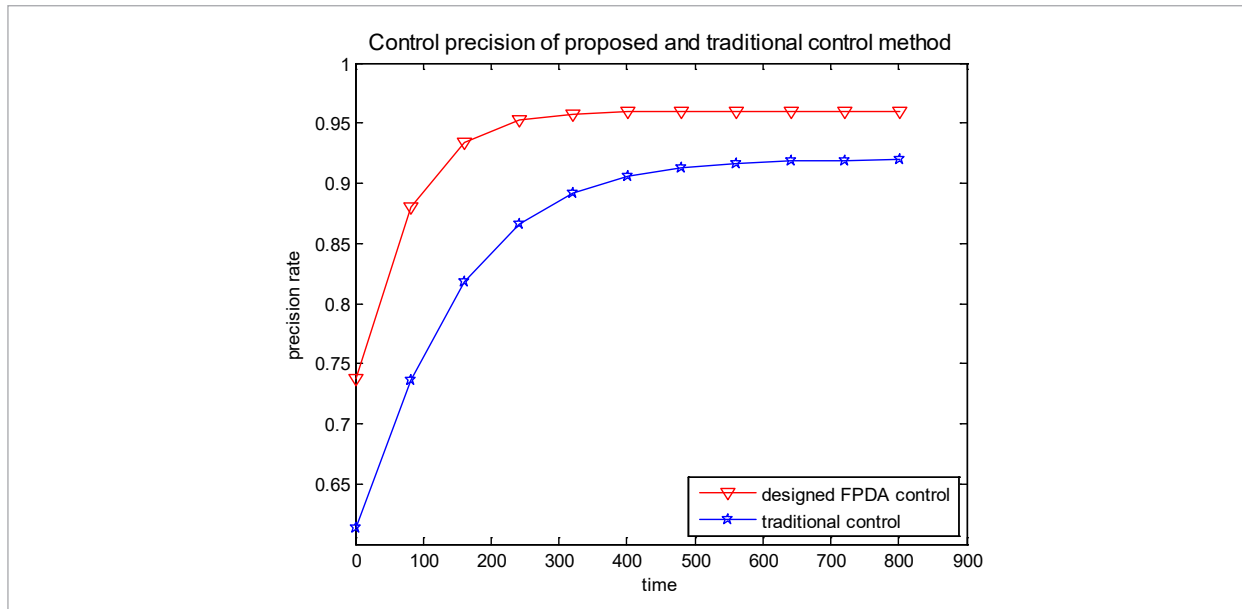


Figure 9

The accuracy of FPDA and existing control method for target tracking control



better control effect. The average precision of FPDA control is 96.87%. However, that of traditional controllers is 91.89%. It shows that the average precision of FPDA control is 4.98% higher than that of the traditional control method, and its control speed is 12s faster than that of the traditional control method. The comparison result is shown in Figure 9.

From Figure 9, the accuracy of the designed FPDA control method in this paper is better than that of existing control method for target control.

5. Conclusion

Based on the fuzzy set theory, this paper designs a control method of fuzzy pushdown automata. The control design model and control decision inference rules of FPDA are given. Secondly, the prediction control pro-

cess by FPDA for quality of spinning yarn is discussed. Then, the comparison of FPDA and other control method in target control is given. The simulation results show that the control method of designed FPDA has faster control speed and higher control precision.

In the future, it is also necessary for scholars to conduct in-depth research in some attributes of FPDA, such as its lattice, semigroup, group, etc.

Acknowledgement

This work is supported by Center Plain Science and Technology Innovation Talents (194200510016); Science and Technology Innovation Team Project of Henan Province University (19IRTSTHN013); Fourth Intelligent Compilation Zhengzhou 1125 Science and Technology Innovation Talents (192101059006); Henan Association for Science and Technology (HKNKJZK 2020 15C), respectively.

References

1. Adamatzky, A. On Interplay Between Excitability and Geometry. *Biosystems*, 2020, 187-188. <https://doi.org/10.1016/j.biosystems.2019.104034>
2. Benmessahel, B., Touahria, M., Nouioua, F. Predictability of Fuzzy Discrete Event Systems. *Discrete Event Dynamic Systems*, 2017, 27(2), 1-33. <https://doi.org/10.1007/s10626-017-0256-7>

3. Blanco, A., Delgado, M., Pegalajar, M. C. Identification of Fuzzy Dynamic Systems Using Max-Min Recurrent Neural Networks. *Fuzzy Set and Systems*, 2000, 1(48), 63-70. [https://doi.org/10.1016/S0165-0114\(00\)00069-5](https://doi.org/10.1016/S0165-0114(00)00069-5).
4. Du, Y. B., Zhu, P. Fuzzy Approximations of Fuzzy Relational Structures. *International Journal of Approximate Reasoning*, 2018, 98, 1-10. <https://doi.org/10.1016/j.ijar.2018.04.003>
5. Feng, X. Q., Wei, C. P., Liu, Q. EDAS Method for Extended Hesitant Fuzzy Linguistic Multi-criteria Decision Making. *International Journal of Fuzzy Systems*, 2018, 20(8), 2470-2483. <https://doi.org/10.1007/s40815-018-0504-5>
6. Garhwal, S., Jiwari, R. Conversion of Fuzzy Automata Into Fuzzy Regular Expressions Using Transitive Closure. *Journal of Intelligent and Fuzzy Systems*, 2016, 30(6), 3123-3129. <https://doi.org/10.3233/IFS-152038>
7. Gulistan, M., Feng, F., Khan, M., Sezgin, A. Characterizations of Right Weakly Regular Semigroups in Terms of Generalized Cubic Soft Sets. *Mathematics*, 2018, 6(12), 293. <https://doi.org/10.3390/math6120293>
8. Jose, R., Mendivil, G. D. Conditions for Minimal Fuzzy Deterministic Finite Automata via Brzozowski's Procedure. *IEEE Transactions on Fuzzy Systems*, 2018, 26(4), 2409-2420. <https://doi.org/10.1109/TFUZZ.2017.2775601>
9. Kacur, J., Flegner, P., Durdan, M., Laciak, M. Model Predictive Control of UCG: An Experiment and Simulation Study. *Information Technology and Control*, 2019, 48(4), 557-578. <https://doi.org/10.5755/j01.itc.48.4.23303>.
10. Karthikeyan, V. Directable Fuzzy Automata. *International Journal of Computer Applications*, 2015, 125(8), 1-4. <https://doi.org/10.5120/ijca2015906119>
11. Kosmatopoulos, E. B., Polycarpou, M. M., Christodoulou, M. A., Ioannou, P. A. High-order Neural Networks for Identification of Dynamical Systems. *IEEE Trans. Neural Networks*, 1995, 6(1), 422-431. <https://doi.org/10.1109/72.363477>
12. Li, Y. M., Wang, Q. The Universal Fuzzy Automaton. *Fuzzy Sets and Systems*, 2014, 249(8), 27-48. <https://doi.org/10.1016/j.fss.2013.08.002>
13. Micic, I., Jancic, Z., Ignjatovic, J., Ciric, M. Determinization of Fuzzy Automata by Means of the Degrees of Language Inclusion. *IEEE Transactions on Fuzzy Systems*, 2015, 23(6), 2144-2153. <https://doi.org/10.1109/TFUZZ.2015.2404348>
14. Omlin, C. W., Thornber, K. K., Giles, C. L. Fuzzy Finite State Automata Can Be Deterministically Encoded Into Recurrent Neural Networks. *IEEE Transaction on Fuzzy Systems*, 1998, 6(1), 76-89. <https://doi.org/10.1109/91.660809>
15. Pan, H., Li, Y., Cao, Y., Li, P. Nondeterministic Fuzzy Automata with Membership Values in Complete Residuated Lattices. *International Journal of Approximate Reasoning*, 2017, 82(3), 22-38. <https://doi.org/10.1016/j.ijar.2016.11.020>
16. Shamsizadeh, M. Intuitionistic General Fuzzy Automata. *Soft Computing*, 2016, 20(9), 3505- 3519. <https://doi.org/10.1007/s00500-015-1969-x>
17. Sharma, B. K., Tiwari, S. P., Sharan, S. On Algebraic Study of Fuzzy Multiset Finite Automata. *Fuzzy Information and Engineering*, 2016, 8(3), 315-327. <https://doi.org/10.1016/j.fiae.2016.09.001>
18. Turksen, I. B., Yao, D. D. Representation of Connectives in Fuzzy Reasoning: The View Through Normal Forms. *IEEE Transaction on Systems, Man and Cybernetics*, 1984, 14, 146-151. <https://doi.org/10.1109/TSMC.1984.6313277>
19. Wei, X. J., Li, Y. M. Fuzzy Alternating Automata over Distributive Lattices. *International Journal of Approximate Reasoning*, 2017, 425, 34-47. <https://doi.org/10.1016/j.ijar.2017.07.007>.
20. Yager, R. R. Families of OWA Operators. *Fuzzy Sets and Systems*, 1993, 59, 125-148. [https://doi.org/10.1016/0165-0114\(93\)90194-M](https://doi.org/10.1016/0165-0114(93)90194-M)
21. Yang, C., Li, Y. Approximate Bisimulation Relations for Fuzzy Automata. *Soft Computing*, 2017, 22(14), 4535-4547. <https://doi.org/10.1007/s00500-017-2913-z>

