IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 4, 2021, accepted January 26, 2022, date of publication February 10, 2022, date of current version February 17, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3149838

Research and Embedded Implementation of Let-Off and Take-Up Dynamic Control Based on Fuzzy Neural Network and Vector Control Optimization

YANJUN XIAO[1,2,3], XIAOLIANG WANG[3], FURONG HAN[1,3], PING LIU[3], AND YUNFENG JIANG[3]

[1]Tianjin Key Laboratory of Power Transmission and Safety Technology for New Energy Vehicles, School of Mechanical Engineering, Hebei University of Technology, Tianjin 300130, China
[2]Career Leader Intelligent Control Automation Company, Suqian, Jiangsu 223800, China
[3]School of Mechanical Engineering, Hebei University of Technology, Tianjin 300130, China

Corresponding authors: Yanjun Xiao (xyj@hebut.edu.cn) and Yunfeng Jiang (awaear168@163.com)

This work was supported in part by the Jiangsu Province Training Fund under Grant BRA2020244, and in part by the National Natural Science Foundation of China under Grant 51675160.

ABSTRACT This paper investigates the warp let-off and take-up mechanism of rapier looms to solve the problem that the warp tension of rapier looms fluctuates greatly and the warp let-off is difficult to maintain constant. The design and hardware implementation of a let-off and take-up control system based on fuzzy neural network (FNN) and vector control (VC) are presented to improve the control level of warp tension and drive performance of the let-off and take-up system. Firstly, the spring-damper dynamic model of the warp is established according to the mechanical properties. The parametric expression of warp tension and the control strategy of fixed angle interval based on let-off and take-up motions are constructed according to the generation mechanism and fluctuation law of warp tension. Then, based on fuzzy reasoning mechanism and neural network model, the fusion theory of fuzzy neural network is introduced, and a tension controller based on T-S fuzzy neural network (FNN) is designed. FNN is trained by introducing genetic optimization and the backpropagation fusion algorithm (GA-BP). In addition, a specialized let-off and take-up hardware circuit is constructed through embedded technology, and the SVPWM algorithm is used as the driving scheme of the hardware circuit. Finally, simulation and actual weaving experiments test the proposed let-off and take-up control system and hardware circuit. The results show that, compared to PID and fuzzy PID, the proposed fuzzy neural network algorithm has higher tension control accuracy and can effectively restrain the rapier loom's warp tension undulation. The designed hardware circuit and SVPWM algorithm have a fast and stable driving ability, which ensures the constant let-off amount.

INDEX TERMS Rapier loom, tension control, vector control, fuzzy neural network, genetic algorithm.

I. INTRODUCTION

The rapier loom is a kind of shuttleless loom widely used in the textile industry because of its high speed, high degree of automation, and good production efficiency [1]. In addition to the characteristics of high speed, high automation and high efficiency of the shuttleless loom, its active weft insertion mode has strong variety adaptability. It can adapt to the weft insertion of all kinds of yarns. Rapier looms also have obvious advantages in multi-color weft weaving [2], [3]. The let-off and take-up mechanism is an important part of the rapier loom. It provides warp yarn for the weaving process, which directly impacts the quality and efficiency of weaving cloth. With the advancement of weaving technology and the demand for the high-speed loom, the traditional mechanical let-off and take-up mechanism has been gradually replaced by electronic let-off and electronic take-up devices, becoming the mainstream driving mode. Therefore, computer technology makes it possible to control the let-off and take-up mechanism more accurately and effectively.

The associate editor coordinating the review of this manuscript and approving it for publication was Bidyadhar Subudhi.

As the speed of the rapier loom increases, the faster weaving speed will have a significant impact on warp yarn. Therefore, the control of warp becomes more and more important. Too large or too small warp tension will affect the quality of the fabric. If the tension is too large, the warp yarn will be broken, and if the tension is too small, the cloth will relax. At the same time, if the let-off quantity is not guaranteed, the weft density of the fabric will be affected, increasing the scrap rate of the fabric. Therefore, it is very important for fabric quality to realize the stable control of warp yarn through the research of let-off and take-up control system.

At present, the research on let-off and take-up control system mainly focuses on warp tension control. The use of intelligent control methods effectively ensures the stability of the warp tension during the loom's high-speed operation, allowing the loom to complete high-quality weaving tasks at various speeds and yarns. Cao [4] developed and applied a warp tension control algorithm based on fuzzy PI parallel control and a constant tension fuzzy control strategy to a yarn winding system. Simulation experiments show that the controller has good robustness and achieves a good control effect. Lu and Yang [5] designed and researched the multi-beam let-off mechanism and established a mathematical model of carbon fiber warp tension. The warp tension was controlled by fuzzy PID, and its effectiveness was verified. To address the warp tension control issue of the carbon fiber multi-layer twill loom's let-off and take-up system, Liu *et al.* [6] established a dynamic model of the let-off and take-up system based on dynamic analysis. The tension network is controlled using adaptive fuzzy PID control. When compared to the PID control, the adaptive fuzzy PID control has fewer overshoots, less output tension fluctuation, a better tracking impact, and more stable simulation under various situations. Wang *et al.* [7] investigated the structure and control method of the loom's let-off and take-up mechanisms and developed a let-off and take-up control model based on a fuzzy neural network adaptive PID algorithm. The results prove that the model enhances the accuracy of the let-off and take-up system and the quality of the textiles. Yang *et al.* [8] applied a linear interpolation fuzzy controller to the yarn tension control system based on a correction factor. By comparing the effect of the classic fuzzy control on yarn tension control, the fuzzy interpolation control can fundamentally eliminate the steady-state deviation and chatter phenomenon of the system. Ni *et al.* [9] proposed a fuzzy multi-attribute group decision-making method, which combines the individual opinions of experts and group similarity to obtain the group opinion decision matrix and then uses the fuzzy algorithm to determine the optimal plan for warp tension detection and control. The findings validate the method's feasibility of the proposed. In addition, equivalent synovial control and speed decentralized control has been applied in warp tension control [10], [11]. Most of these algorithms are based on fuzzy control theory. The simple fuzzification of information by fuzzy control will cause the control accuracy of the system to decrease. To improve accuracy, it is often necessary to increase the number of fuzzy quantities during fuzzification or increase the set of control rules, which will increase the search time of control rules and affect the system's response speed. At the same time, the establishment of membership functions and fuzzy rules highly depends on expert experience [12]. As a result, to overcome the limitations of the aforementioned fuzzy control, Li *et al.* [13] applied the multi-neuron adaptive PID control method on the SAURER400 rapier loom's let-off system. The simulation results of Simulink show that the multi-neuron adaptive PID algorithm has the advantages of fast response and small overshoot compared with fuzzy PID. Liu and Zhang [14] proposed a Kalman filter adaptive PID control algorithm based on RBF neural network. The results show that the control effect and dynamic performance of adaptive PID control based on the neural network are significantly better than conventional PID control. Wang *et al.* [15] proposed a PID tension control algorithm based on RBF neural network tuning in the design of the integrated controller of let off, take-up and fuzzing based on an ARM microcontroller. The debugging results show that the algorithm solves the problems of uneven weft density and tension control of towel loom. Xu and Yang [16] proposed a fusion algorithm, including neural network, fuzzy control algorithm and multi-sensor information fusion algorithm. The neural network obtains the algorithm weight through the chaotic optimization method and is then fused by fuzzy comprehensive evaluation. The experimental results show that the fusion algorithm improves the tension control accuracy of the carbon fiber multilayer loom. The above research shows that using a neural network algorithm further improves the control accuracy of warp tension compared to a single fuzzy control. Although neural networks have strong self-learning and self-adaptation capabilities, neural networks can only describe complex functional relationships between large amounts of data and cannot express the reasoning function of the human brain well. Therefore, it cannot make effective use of expert knowledge. Therefore, combining fuzzy control and neural network algorithm will be a more effective solution. Therefore, fuzzy neural networks are proposed to solve complex control problems and have been applied in various control scenarios [17]–[19]. However, the current research on applying fuzzy neural networks in the warp tension control of rapier looms is still rare. Compared with the traditional fuzzy control and neural network, the fuzzy neural network (FNN) fully considers the complementarity of the two. It has the capabilities of self-learning, self-adaptive and fuzzy information processing. However, fuzzy neural networks often use BP algorithms to train FNN controllers to achieve the expected output. But the use of the BP algorithm makes FNN need more time to reach the ideal state. At the same time, the global optimization ability of the BP algorithm is poor, and it is easy to fall into the local optimal solution, resulting in unsatisfactory final optimization accuracy [20]. Therefore, improving the

training algorithm of FNN, increasing the training speed and enhancing the global optimization ability have become important aspects of research.

The dynamic response performance of the let-off and take-up mechanism is also an important factor affecting fabric quality. The guarantee of let-off quantity requires that the let-off and take-up motors adjust the speed in time to ensure the constant weft density. Therefore, the let-off motor and take-up motor need speed regulation control. The existing control methods include backstepping control (BSC), vector control (VC), direct torque control (DTC), model predictive control (MPC) and other control methods [21]. Among them, the vector control method currently has obvious advantages in the speed control of the motor. Ramasamy and Krishnasamy [22] applies the SVPWM algorithm to a three-phase five-level dual inverter-fed open-winding induction motor. Compared with the traditional PWM method, SVPWM technology provides better performance for a three-phase five-level dual NPC inverter, And also reduces the complexity of switching time calculation and reference vector identification, making the motor run more smoothly. To improve the performance of the four-level open-winding induction motor driver, Lakhimsetty *et al.* [23] uses an improved SVPWM driving method. This solution can achieve better harmonic performance without changing the power supply circuit configuration or equipment voltage rating, thereby improving the driving performance of the motor. Wu *et al.* [24] proposed an efficient synchronous modulation method for dual three-phase motors within the full modulation range based on 24-sector SVPWM. It can improve control performance during a pulse mode change by reducing the abrupt drop in switching frequency. Through the above research, we can see the potential and advantages of the SVPWM algorithm in optimizing the performance of the inverter and driving the motor efficiently and smoothly.

Therefore, based on the fuzzy neural network (FNN) and vector control method, a new let-off and take-up controller for rapier loom is proposed in this paper. The innovations of this paper are as follows:

A spring-damper parallel dynamic model of the warp yarn was proposed based on the dynamic analysis of the let-off and take-up movement. The relationship between the let-off and take-up movement and the warp tension was established. And through the analysis of the influencing factors of tension, the tension sampling strategy of fixed angle interval is proposed.

A fuzzy neural network tension control strategy is proposed based on genetic and backpropagation algorithms (GA-BP FNN). This method is appropriate for controlling loom warp tension with time-varying, nonlinear and multivariable coupling. The application of the GA-BP algorithm greatly improves the training speed and accuracy of FNN. It provides a practical and effective method to improve rapier loom's tension control accuracy and weaving quality.

The hardware circuit of let-off and take-up for rapier loom is designed based on embedded technology. Its reliability and practicability are verified by weaving experiments. This module has high integration, strong flexibility, and is more practical in engineering.

The SVPWM algorithm is used as a key driving technology in the designed let-off and take-up hardware circuit, which improves the dynamic response performance of the let-off and take-up system. Its driving performance is verified by comparison with the SPWM algorithm. The motor no-load experiment and weaving experiment show that it has good drive stability in the rapier loom.

II. WORKING PRINCIPLE OF RAPIER LOOM

Rapier looms as one of the three types of shuttles less looms. The main difference between rapier looms and the other two shuttle-less looms lie in the way of weft insertion [25], [26]. At present, rapier looms usually use a flexible rapier belt to insert weft. Driven by the spindle, the rapier belt introduces the yarn sent by the weft selection finger into the shed through reciprocating motion. During the rapier loom weaving process, as shown in Fig. 1, if the spindle is rotated one week as a weaving cycle, the loom needs to complete the five major movements of opening, weft insertion, beating-up, let-off and take-up according to the rotation angle of the spindle in each weaving cycle.

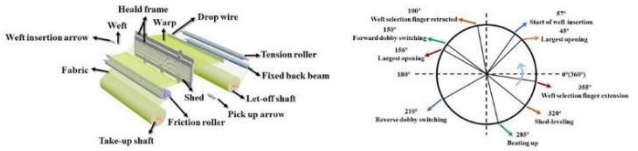

FIGURE 1. Schematic diagram of fabric formation of the rapier loom.

III. RESEARCH AND ANALYSIS OF LET-OFF AND TAKE-UP

A. WARP TENSION DYNAMICS MODEL

In the rapier loom, the let-off and take-up mechanism mainly completes the let-off motion and take-up motion in the weaving process, continuously leading the fabric away from the weaving mouth and sending out a certain amount of yarn in time to ensure the continuous progress of the weaving process. The let-off and take-up mechanisms are critical in maintaining constant warp tension and determining fabric quality. The let-off and take-up structure is shown in Fig. 2. The arrow denotes the warp or fabric's movement direction, the fabric leaves the weaving mouth at a certain speed driven by the friction force of the friction roller B and basically does not slip, and the friction roller B is driven by the take-up motor and the corresponding deceleration device; the take-up shaft C winds the fabric into a coil, and a conveyor belt connects the take-up shaft C and the friction roller B. The let-off shaft A keeps sending out the warp yarn under the control of the let-off motor to recompense for the winding length of the friction roller B and make sure weaving continuity. However, when the take-up mechanism leads the fabric away from the mouth,

the warp shape variable increases, resulting in an increase in warp tension; conversely, when the let-off mechanism sends out the warp, the warp shape variable decreases, resulting in a decrease in warp tension. As a result, the dynamic stability of warp tension in the weaving process can only be guaranteed if the let-off and take-up are kept in dynamic balance.

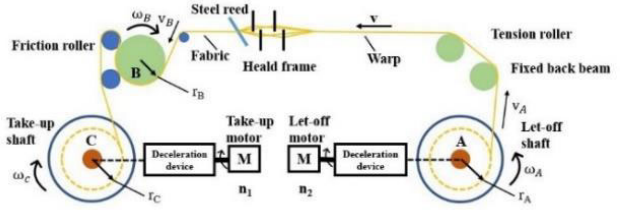

FIGURE 2. Schematic diagram of let-off and take-up structure.

Yarns are continuous thread-like objects made of various textile fibers, fine and soft, with distinct viscoelastic mechanical properties. For the mechanical properties of warp yarns, the most basic viscoelastic components mainly include elasticity and damping. In the warp drive process, to obtain the mathematical relationship between warp speed and warp tension and to achieve real-time and accurate control of warp tension by the controller, a spring-damper parallel dynamics model is used by analyzing the dynamic characteristics of the warp yarn, as shown in Fig. 3.

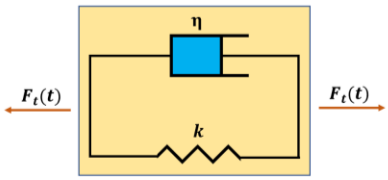

FIGURE 3. Warp spring-damper model.

The warp yarn elasticity and viscosity coefficient are assumed to be constant and not affected by time and warp length. When the warp yarn is naturally straightened, the warp yarn deformation $\varepsilon\,(0) = 0$ and the warp tension $F_t\,(0) = 0$.

At this point, the formula for warp tension is

$$F_t\,(t) = \varepsilon\,(t) \times k + \varepsilon'(t) \times \eta \tag{1}$$

where $F_t\,(t)$ is the instantaneous tension of the warp yarn during weaving; k is the elasticity coefficient in the viscoelastic model and η is the viscous coefficient in the viscoelastic model; $\varepsilon\,(t)$ is the total deformation of the warp yarn due to various motions.

The main factors causing the change of warp tension are let-off motion and take-up motion. Therefore, the warp deformation $\varepsilon\,(t)$ caused by let-off motion and take-up motion can be expressed as:

$$\varepsilon\,(t) = L_j\,(t) + L_s(t) \tag{2}$$

where $L_j\,(t)$ is the warp take-up amount per unit time, and $L_s(t)$ is the warp let-off amount per unit time.

After Laplace transformation of formula (1) and substitution of formula (2), the warp tension can be obtained:

$$F_t\,(s) = k_1(1 + \frac{\eta}{k_1} \times s) \times \left[L_j\,(s) - L_s(s)\right] \tag{3}$$

The coiling amount of friction roller per unit time is:

$$L_j\,(t) = \int_0^t \frac{\omega_B(t) \times (1 - a_f)}{2\pi \times \lambda} dt \tag{4}$$

where a_f is the radial shrinkage of the fabric, which can be taken as 2% for plain cloth yarn, λ is the weft density of the fabric, and $\omega_B(t)$ is the speed of friction roller.

From Laplace transform:

$$L_j\,(s) = \frac{\omega_B(s) \times (1 - a_f)}{s \times 2\pi \times \lambda} \tag{5}$$

Let-off quantity of warp shaft in unit time:

$$L_s\,(t) = \int_0^t r_A(t) \times \omega_A(t) dt \tag{6}$$

where $\omega_A(t)$ is the speed of the let-off shaft and $r_A(t)$ is the radius of let-off shaft.

From Laplace transform:

$$L_s\,(s) = \frac{1}{s} \times r_A(s) \times \omega_A(s) \tag{7}$$

Substituting (5) and (7) into (3),

$$F_t\,(s) = \frac{k_1}{s}(1 + \frac{\eta}{k_1} \times s) \times [\alpha\omega_B(s) - r_A(s)\omega_A(s)] \tag{8}$$

where $\alpha = \frac{(1-a_f)}{2\pi\times\lambda}$.

$\omega_B(t)$ is linearly related to the spindle speed, when the loom is running with a certain spindle speed, $\omega_B(t)$ is also determined. Suppose $\omega_B(t) = \omega_j$, substitute $\omega_A(t) = \mathrm{I}\times\omega(t)$ into (8),

$$F_t\,(s) = \frac{k_1}{s}(1 + \frac{\eta}{k_1} \times s) \times \left[\alpha\omega_j - Ir_A(s)\omega(s)\right] \tag{9}$$

where I is the let-off transmission ratio and $\omega(t)$ is the speed of the let-off motor. The transfer relationship between the let-off motor speed and the warp tension can be obtained from formula (9), which provides a basis for the controller to realize the warp tension control by controlling the let-off motor speed. In the weaving process, the warp axis radius $r_A(s)$ decreases continuously, resulting in the continuous change of $G\,(s)$. To simplify the calculation, make $r_A(s) = r_A(0)$. Before starting the loom, set the fixed reference input, and the initial radius of the warp axis is 620mm. The transfer function of the let-off system model is:

$$G\,(s) = \frac{0.06s + 0.45}{2.64s^2 + 27.12s + 1} \tag{10}$$

B. WARP TENSION CONTROL SCHEME

The opening movement, weft insertion movement and beating up movement will also affect the warp tension.

The heald frame moves up and down during the opening movement to divide the entire warp into two parts, forming a channel (shed) through which the weft yarn can be introduced and intertwined with the warp yarn. In the process of shed production, the yarn often deforms. Because the opening action is periodic in the weaving process, these deformations will have a periodic impact on the warp tension. Fig. 4 shows the elongation deformation of warp yarn when opening.

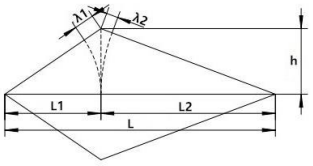

FIGURE 4. Schematic diagram of warp elongation at the opening.

In the figure, L is the length before the warp opening, L_1 and L_2 are the length before and after the shed, respectively. Assuming that the elongation of the warp yarns at the front and back of the shed at time t is λ_1 and λ_2, respectively, and the shed height is h, the elongation deformation of the warp yarn at time t due to the opening is:

$$\varepsilon(t) = \lambda_1 + \lambda_2 = \frac{h^2}{8}\left(\frac{1}{L_1} + \frac{1}{L_2}\right) \tag{11}$$

According to the stress-strain relationship, the change in tension ΔF_t is directly proportional to the square of opening height h. Due to the periodic movement of the opening mechanism, the warp will produce periodic deformation, resulting in periodic fluctuation of warp tension.

After the shed is formed, the weft insertion begins. Following weft insertion, the warp and weft yarns are interwoven to form a fabric using a beating-up movement. Beating-up is when the reed pushes the weft introduced into the shed to the weaving mouth and tightens it. The schematic diagram of the beating-up action is shown in Fig. 5. The reed pushes the weft to the final position of the beating-up, which produces an instant stretch on the warp yarn, which causes the yarn tension to increase rapidly and produce a peak of tension fluctuation. In the weaving process of the loom, the beating-up movement is a periodic movement, so this movement will cause periodic high-frequency fluctuations in the warp tension.

According to the preceding analysis, warp tension is primarily for yarn distortion caused by let-off and take-up movement, resulting in the internal tension of the yarn, and the periodic movement of the opening, weft insertion, and beating up structure will also cause periodic fluctuations to the warp. As a result, if the warp tension is measured in actual time, the average value of periodic tension will vary greatly. Using this value as the tension controller's input will significantly impact the system's stability.

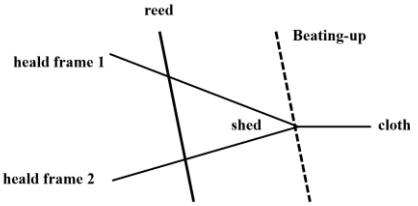

FIGURE 5. Schematic diagram of beating-up action process.

According to the working sequence diagram of the rapier loom in Fig. 1, through reasonable division, try to avoid the peak tension generated during opening and weft beating. The opening is at its max and remains static when the spindle angle is between 45° and 57°. During this moment, the weft insertion action has only just started, but the beating-up has still not started, and the weft has still not entered the shed, so the warp tension is unaffected. Therefore, the sampling angle selected in the system is within the angle range of 45 ° ∼ 55 ° of the working sequence, as shown in Fig. 6. Although the sampling range is reduced, the tension peak caused by opening and beating up is fully avoided.

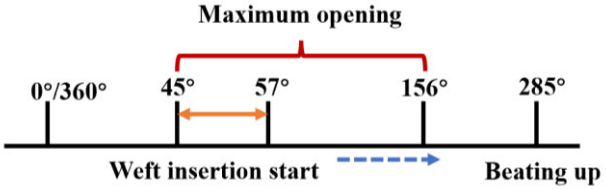

FIGURE 6. Schematic diagram of warp tension sampling interval.

C. DRIVE STRATEGY AND EMBEDDED IMPLEMENTATION

The weft density is generally fixed in the weaving process, so the spindle speed is also fixed. To maintain a constant weft density, the take-up speed should have a linear relationship with the spindle speed, and the take-up motor speed should also remain stable. If variable weft density weaving is required, the speed of the take-up motor needs to be adjusted. According to the relationship between the speed of the let-off motor and the warp tension, the speed of the let-off motor needs to be dynamically adjusted to keep the tension constant. Therefore, the let-off and take-up motors need speed control.

The let-off and take-up motors are controlled by the let-off and take-up control module, and the let-off and take-up movements are completed by receiving the control signals of the main control module through the CAN bus. To improve the accuracy of let-off and take-up, the permanent magnet synchronous motor (PMSM) is used as the executive element, so the drive and control circuit of the motor should be included in the structure of the module. At the same time, the module can also collect tension signals and control the

tension by adjusting the speed of the let-off or take-up motor. The hardware circuit structure of the let-off and take-up control module is shown in Fig. 7.

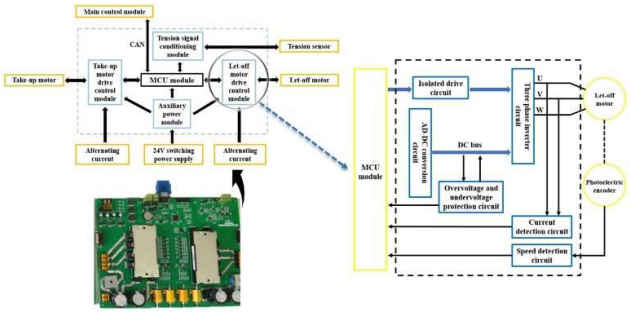

FIGURE 7. Schematic diagram of the hardware structure of let-off and take-up control module.

The vector control method is applied to control the let-off and take-up motors because the let-off and take-up system need good dynamic response-ability to ensure constant weft density. At the same time, we need to implement the control strategy on the hardware circuit, which is a part of building a complete control system. However, this control method has a variety of implementation algorithms. It should be pointed out that SPWM is the most widely used vector algorithm but compared with SVPWM, it has a larger amplitude and more dense harmonic distribution, so its harmonic distortion rate is higher than SVPWM. Therefore, it isn't easy to obtain good driving ability using the SPWM algorithm. The SVPWM (Space Vector Pulse Width Modulation) algorithm is used to control the three-phase inverter module to output the required U, V and W phase voltages of the motor, the inverter circuit, as shown in Fig. 8.

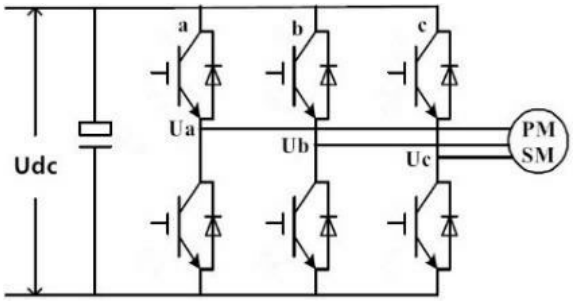

FIGURE 8. Let-off and take-up inverter circuit.

The principle of the SVPWM control algorithm is to simplify the output of the three-phase voltage and the motor rotor into a spatially synthesized vector, thereby converting the control of the three voltage scalars into a voltage vector control. Since the rotors in the motor are evenly distributed, and the difference is 120° in space, the three-phase voltage output by the inverter is $U_A(t)$, $U_B(t)$, $U_C(t)$. When the three-phase voltage is added to the three-phase static coordinate system, the time difference is 120 °, then the combined space vector $\vec{U}_S(t)$ is:

$$\vec{U}_S(t) = U_A(t)e^{j0} + U_B(t)e^{j\frac{2\pi}{3}} + U_C(t)e^{-j\frac{2\pi}{3}} = \frac{3}{2}U_m e^{j\omega t} \tag{12}$$

where, U_m is the phase voltage amplitude, ω is the phase voltage angular frequency. It can be seen that $\vec{U}_S(t)$ is a rotating space voltage vector with an angular frequency of $\omega = 2\pi f$. The purpose of the SVPWM algorithm is to use the switching state of the three-phase bridge to express the $\vec{U}_S(t)$ vector rotating in space.

Since the three-phase bridge arm of the inverter has six switches in total, to study the space voltage vector output by the inverter when the upper and lower bridge arms of each phase are combined with different switches, the switching function S_x (x = a, b, c) is defined as:

$$S_x = \begin{cases} 1, & \textit{upper bridge arm is on} \\ 0, & \textit{lower bridge arm is off} \end{cases} \tag{13}$$

When S_x(x = a, b, c) is 1, the corresponding bridge arm output voltage equals the bus voltage, and when it is 0, the output voltage is 0. The three pairs of bridge arms have a maximum of eight states. Combining formula (12), the inverter output voltage $\vec{U}_{out}(t)$ is:

$$\vec{U}_{out} = \frac{2}{3}U_{dc}\left(S_a + S_b e^{j\frac{2\pi}{3}} + S_c e^{-j\frac{2\pi}{3}}\right) \tag{14}$$

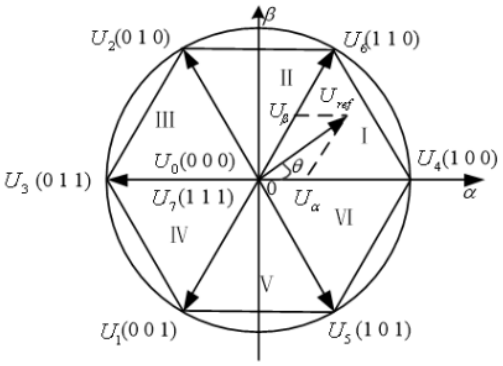

FIGURE 9. Schematic diagram of voltage space vector distribution.

Arrange these eight voltage vectors evenly in phase, as shown in Fig. 9. The figure is divided into six sectors, and in each sector, assuming that the expected voltage vector is $\vec{U}_{ref}$, the adjacent voltage vectors can be synthesized according to the principle of average equivalence. If the switching period of the inverter output voltage is recorded as T, the composite relationship of $\vec{U}_{ref}$ can be expressed by the following formula:

$$\int_0^T \vec{U}_{ref}\,dt = \int_0^{T_x} \vec{U}_x\,dt + \int_{T_x}^{T_x+T_y} \vec{U}_y\,dt + \int_{T_x+T_y}^{T} \vec{U}_0'\,dt \tag{15}$$

where, T_x and T_y respectively correspond to the action time of two non-zero vectors $\vec{U}_x$ and $\vec{U}_y$ in the period T. $\vec{U}'_0$ is the zero vector, which can be $\vec{U}_0$ or $\vec{U}_7$. The action time is T-T_x-T_y.

According to the above principle and the vector decomposition principle, the action time of the two non-zero vectors and the two zero vectors of $\vec{U}_{ref}$ can be calculated by the formula (16).

$$\begin{cases} T_a = (T_s - T_x - T_y)/4 \\ T_b = (T_a + T_x)/2 \\ T_c = (T_b + T_y)/2 \end{cases} \tag{16}$$

where T_x and T_y are the action time of two components $\vec{U}_x$ and $\vec{U}_y$, T_s is the inverter switching time. Then, the on-time corresponding to the three-phase switches of inverters A, B, and C in each sector can be calculated according to formula (16), as shown in Table 1.

TABLE 1. Three-phase conduction time of inverter A, B, C.

Phase	Sector N					
	I	II	III	IV	V	VI
A	T_b	T_a	T_a	T_c	T_c	T_b
B	T_a	T_c	T_b	T_b	T_a	T_c
C	T_c	T_b	T_c	T_a	T_b	T_a

During programming, the action time can be written into the corresponding compare register according to Table 1. In the switch state transition process, to minimize the number of switching actions, 7-segment modulation is used. The zero-vector action time is evenly distributed to generate a symmetrical PWM waveform [27], [28]. The vector control method is used to experiment with the motor using SVPWM and SPWM algorithms respectively, setting the speed as $400rad/s$ and torque as $10N \cdot m$. The system response curve is shown in Fig. 10.

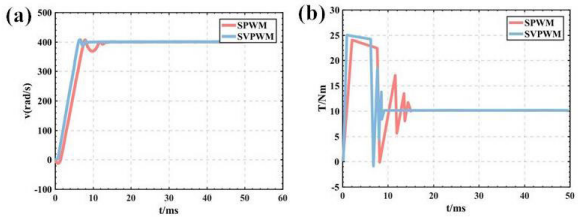

FIGURE 10. Comparison of SVPWM and SPWM motor speed and electromagnetic torque.

Fig. 10(a) shows the motor speed waveform. Under the same conditions, the time required to reach a steady state with the SVPWM algorithm is less than that of the SPWM algorithm. Fig. 10(b) shows the electromagnetic torque waveform. The motor starting torque is slightly higher with the SVPWM algorithm, but the torque pulsation is less than the SPWM algorithm. It can be seen that the SVPWM algorithm has better driving performance than the SPWM algorithm. Therefore, the SVPWM algorithm can make the let-off and take-up motor obtain a faster response speed and smaller adjustment error. To maintain constant weft density and ensure fabric quality.

IV. CONTROLLER DESIGN

With the improvement of the speed of rapier loom and the time-varying, nonlinear and more interference of the warp tension system of the rapier loom, it is limited to use the traditional PID algorithm to control the warp tension. In this paper, fuzzy neural network and traditional PID control are combined to control the tension of the rapier loom. The fuzzified data is input into the neural network using the complementarity between fuzzy control and neural network. The fuzzy rules are extracted through the neural network's learning, making the fuzzy system with generalization capability. While retaining the inherent accuracy of PID control, it enhances the adaptive ability, learning ability and robustness of the PID controller. Therefore, a PID tension closed-loop control system model based on fuzzy neural network (FNN) is established, as shown in Fig. 11:

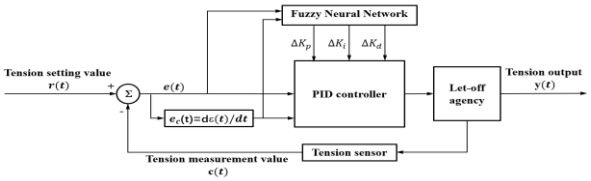

FIGURE 11. Structure diagram of fuzzy neural network PID tension closed-loop control system.

A. FUZZY NEURAL NETWORK

The fuzzy neural network in the system adopts a two input three output control model. The input of the system is the deviation value $e(t)$ between the tension measurement value $c(t)$ and the tension setting value $r(t)$ and the tension error conversion rate $ec(t)$, output as the correction values ΔK_p, ΔK_i and ΔK_d. The three basic parameters K_p, K_i and K_d of the PID controller are adjusted online by the correction value, and the adjustment formula is (17), K_p, K_i and K_d are the set values during system initialization, which can be obtained by the traditional PID tuning method. In order to make the system respond quickly and eliminate the steady-state error as much as possible without causing system oscillation, the initial values of the controller parameters were set to $K_p = 5$, $K_i = 0.8$, and $K_d = 0.4$.

$$\begin{cases} K_p(k) = K_p(k-1) + \Delta K_p \\ K_i(k) = K_i(k-1) + \Delta K_i \\ K_d(k) = K_d(k-1) + \Delta K_d \end{cases} \tag{17}$$

Then, the control variable can be calculated as follows:

$$u(k) = K_p(k)\,e(k) + K_i(k)\sum_{i=1}^{k} e(i) + K_d\left[e(k) - e(k-1)\right] \quad (18)$$

Set the basic domain of the input and output parameters of the fuzzy neural network. The input variables must be converted from the basic domain to the fuzzy domain corresponding to the fuzzy set to carry out the fuzzification process. The input variables need to be multiplied by the corresponding quantization factor:

$$E = e \times K_e, \quad EC = ec \times K_{ec} \quad (19)$$

where, K_e and K_{ec} are input quantization factors.

In addition, the control quantity provided by the fuzzy control algorithm for each sample must be transformed into the basic domain required by the control object. The fuzzy output control quantities ΔK_p, ΔK_i and ΔK_d corresponding to the fuzzy control quantities K_p, K_i and K_d. Then,

$$\begin{cases} \Delta K_p = K_p \times F_{K_p} \\ \Delta K_i = K_i \times F_{K_i} \\ \Delta K_d = K_d \times F_{K_d} \end{cases} \quad (20)$$

where, F_{K_p}, F_{K_i}, and F_{K_d} are output scale factors. Once the quantization and scale factors have been determined, the system can always be mapped to an element on the fuzzy domain. The fuzzy domain corresponding to each parameter are shown in Table 2:

TABLE 2. Domain corresponding table of input and output parameters.

Input and output parameters	Fuzzy domain	Quantification factor	Basic domain
e	[-3,-2,-1,0,1,2,3]	3/5	[-5, 5]
e_c	[-6,-4,-2,0,2,4,6]	1	[-6, 6]
$\triangle K_p$	[-6, -4, -2,0,2,4,6]	1	[-6, 6]
$\triangle K_i$	[-3, -2, -1,0,1,2,3]	3	[-1, 1]
$\triangle K_d$	[-3, -2, -1,0,1,2,3]	3	[-1, 1]

The values in the fuzzy domain in Table 2 are transformed into a fuzzy subset {NB, NM, NS, ZO, PS, PM, PB}. The elements in the subset correspond to seven fuzzy language variables: negative large, negative medium, negative small, zero, positive small, positive medium and positive large. Then, the fuzzy subset of each fuzzy variable is defined. That is, the shape of the membership function of the fuzzy subset is determined. The membership function adopts the Gaussian function to effectively simulate the fuzzy concept of human control activities.

Fuzzy neural network (FNN) uses neural network algorithm structure to realize fuzzy control algorithm. The steps of input and output variable selection, fuzzification, fuzzy reasoning and anti-fuzzification of the fuzzy control system

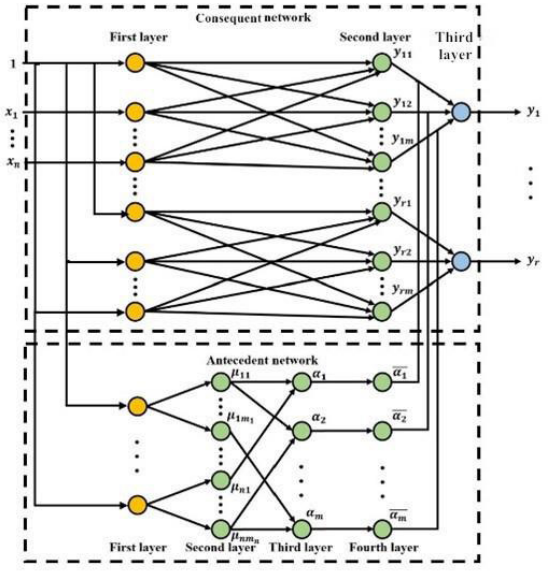

FIGURE 12. T-S fuzzy neural network structure diagram.

are represented by the neural network structure. Fig. 12 shows the T-S fuzzy neural network (T-S FNN). Unlike the standard fuzzy neural network, it directly uses numerical information to establish a specific nonlinear mapping. It uses a small number of fuzzy rules to express a highly nonlinear complex system. Compared with the general fuzzy neural network, its model has more physical significance, fast convergence speed and high precision. The T-S fuzzy neural network consists of the antecedent network and consequent network. The antecedent network has four layers used to match the antecedents of fuzzy rules. The consequent network has three layers used for reasoning to generate fuzzy rules.

Suppose the system input vector $x = [x_1, x_2, \ldots, x_n]^T$, x_i is a fuzzy language variable.

$$T(x_i) = \left\{A_i^1, A_i^2, \ldots, A_i^n\right\} \quad (i = 1, 2, \ldots, n) \quad (21)$$

where, $A_i^j\,(j = 1, 2, \ldots, m)$ is the j th fuzzy set of x_i.

Then for the fuzzy neural network of the T-S model, the fuzzy rules of the input-output relationship are as follows:

$$\begin{aligned} &\textit{if } x_1 \textit{ is } A_1^j \textit{ and } x_2 \textit{ is } A_2^j \textit{ and } \ldots \textit{ and } x_n \textit{ is } A_n^j, \\ &\textit{Then } y_j = p_{j0} + p_{j1}x_1 + p_{j2}x_2 + \cdots + p_{jn}x_n \end{aligned} \quad (22)$$

where, y_j represents the output value obtained by the system according to the fuzzy rule, and $p_{j_0},\ldots, p_{j_n}$ represents the weight value.

In the antecedent network, the first and second layers are the input and fuzzification layers respectively. As mentioned above, Gaussian function is used as the membership function, then:

$$\mu_{ij} = exp\left(-\frac{(x_i - c_{ij})^2}{\sigma_{ij}^2}\right) \quad (\mathrm{i} = 1, 2; \mathrm{j} = 1, 2, \ldots, m_i) \quad (23)$$

where, x_i represents the inputs of the first layer, $x_1 = e$, $x_2 = ec$; m_i stands for x_i fuzzy partition number, $m_1 = 7$, $m_2 = 7$; c_{ij} and σ_{ij} is the antecedent network parameter, which represents the center value and width value of Gaussian function, and each neuron node in the second layer represents a fuzzy language variable.

The third layer is the fuzzy reasoning layer, and each node represents a fuzzy rule; Each input has seven fuzzy language variables, so there are 49 fuzzy rules. Its function is to match the antecedents of the fuzzy rules and calculate the fitness α_k of each rule, which is calculated as:

$$\alpha_k = min\left\{\mu_{1i_1}, \mu_{2i_2}, \ldots \mu_{ni_n}\right\} \quad (\mathrm{k} = 1, 2, \ldots, \mathrm{m};\ \mathrm{m} = \prod_{i=1}^{n} m_i) \tag{24}$$

where, $i_1 \in \{1, 2, \ldots, m_1\}$, $i_2 \in \{1, 2, \ldots, m_2\}$, $\ldots$, $i_n \in \{1, 2, \ldots, m_n\}$.

The fourth layer is the normalization layer, which realizes the normalization calculation. The calculation formula is:

$$\bar{\alpha_k} = \alpha_k / \sum_{n=1}^{m} \alpha_n \tag{25}$$

In the consequent network, the first layer is also the input layer, and each node in the second layer corresponds to a fuzzy rule, which is used to calculate the consequent output of each fuzzy rule, and the formula is:

$$y_{lk} = p_{k0}^{l} + p_{k1}^{l} x_1 + \ldots + p_{kn}^{l} x_n \quad (\mathrm{k} = 1, 2, \ldots, \mathrm{m};\ \mathrm{l} = 1, 2, \ldots, \mathrm{r}) \tag{26}$$

where, $\left\{p_{k0}^{l}, p_{k1}^{l}, \ldots, p_{kn}^{l}\right\}$ is the connection weight from the first to the second layer of the posterior piece network; l is the dimension of the output quantity.

The third layer uses the center of gravity method to calculate the final output of the fuzzy neural network:

$$y_l = \sum_{k=1}^{m} \bar{\alpha_k} y_{lk} \tag{27}$$

where, y_l is the weighted sum of each rule consequent, and the output of the antecedent network α_k is the connection weight of the third layer of the consequent network. Therefore, there are three types of adjustable parameters in the network structure. The first type of parameter p_{kn}^{l} is the connection weight of the consequent network. The second and third types of adjustable parameters are c_{ij} and σ_{ij}, which represent the center and width of the membership function of the antecedent network. These parameters must be obtained by the learning algorithm of the fuzzy neural network.

B. GA-BP ALGORITHM

For the existing fuzzy neural network learning algorithms, the BP learning algorithm is the most widely used. The algorithm uses gradient search technology to continuously adjust the weight of each neuron of the neural network to minimize the mean square error between the actual output value of the network and the expected output value. When adjusting the network weight coefficients, set the error cost function as:

$$E = \frac{1}{2} \sum_{i=1}^{r} (t_i - y_i)^2 \tag{28}$$

where, y_i and t_i are the actual and expected outputs of the network, respectively.

The learning algorithm with parameters c_{ij}, σ_{ij} and p_{kn}^{l} can be described as:

$$\begin{cases} c_{ij}(k+1) = c_{ij}(k) - \beta \dfrac{\partial E}{\partial c_{ij}} \\ \sigma_{ij}(k+1) = \sigma_{ij}(k) - \beta \dfrac{\partial E}{\partial \sigma_{ij}} \\ p_{kn}^{l}(k+1) = p_{kn}^{l}(k) - \beta \dfrac{\partial E}{\partial p_{kn}^{l}} \end{cases} \tag{29}$$

where, β represents the learning rate. However, when the BP algorithm trains fuzzy rules without prior knowledge, its initial state parameter values are randomly set. The network stops training until the error cost function changes very little. This method of randomly selecting initial parameters often causes network learning to fall into a local minimum, causing the learning process to fail.

In recent years, scholars have extensively studied heuristic algorithms such as ant colony algorithms, particle swarm algorithms and genetic algorithms to replace BP neural networks [29], [30]. Genetic algorithms (GA) in intelligent algorithms have the characteristics of searching for the best in the global range and fast search efficiency. The GA can be described as follows: generating several numerical codes of the problem, namely chromosomes, to form the initial population; Individuals with low fitness are removed, and individuals with high fitness are chosen to join in the genetic operation, based on a numerical evaluation of each individual through the fitness function. The individuals after the genetic operation are collected into the next generation of a new population, and the new population is evolved in the next round. Through continuous iteration until the optimal solution is found.

However, it should be pointed out that genetic algorithm is an adaptive heuristic group type iterative global search algorithm with global convergence when optimizing neural network weights and parameters. Training a fuzzy neural network with GA can avoid falling into a local minimum, but the convergence speed will slow down in the later stage of training with GA. Therefore, it is not ideal whether to use GA or BP algorithm alone to train the fuzzy neural network. As a result, this paper uses the GA-BP algorithm to train the FNN. The algorithm flow is shown in Fig. 13.

In the initial stage, GA is used to obtain fast convergence. When the maximum number of iterations or preset accuracy is reached, the optimal initial parameters are assigned to the FNN by the GA. Then the BP algorithm is used to continue the training of the FNN to obtain fast convergence in the later iterations.

Running the GA algorithm requires encoding the fuzzy neural network. The FNN has three training parameters, a total of 28 central and width values for the Gaussian function and 49 connection weights for the consequent network; therefore, the encoding length of the genetic algorithm is 77, and real number encoding is used.

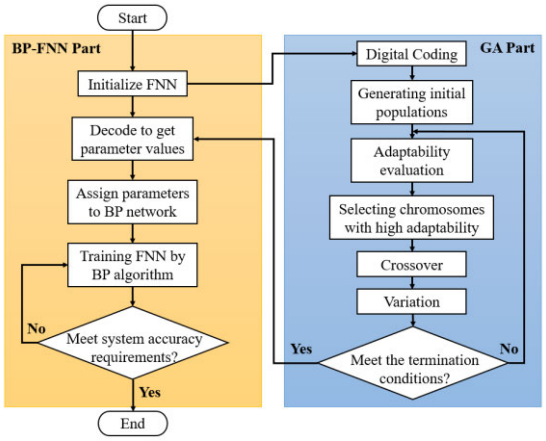

FIGURE 13. GA-BP algorithm flow chart.

The optimization goal of the fuzzy neural network is to make the actual output corresponding to the input data of the network approach the given output of the network to the maximum extent. Take the objective function as shown in formula (30) to minimize it.

$$J = \frac{1}{2n}\sum_{i=1}^{n}(t_i - y_i)^2 \tag{30}$$

where, n represents the number of sample training; t_i represents the expected output of BP neural network; y_i represents prediction output.

Transform the objective function to get the fitness function:

$$Fit(J) = \frac{1}{1+J} \tag{31}$$

The crossover operator selects arithmetic crossover in the GA. The crossover probability is generally 0.4-0.99, and the probability range of the mutation operator is generally 0.0001-0.1. However, improper crossover probability and mutation probability will also affect individual evolution. Therefore, when determining the crossover probability (P_c) and mutation probability (P_m), this paper adopts the adaptive adjustment method to make the randomness of selection and convergence speed reach a certain balance. The formula is:

$$P_c = \begin{cases} k_1 \dfrac{f_{max} - f'}{f_{max} - f_{avg}}, & f' \geq f_{avg} \\ k_3, & f' \leq f_{avg} \end{cases} \tag{32}$$

$$P_m = \begin{cases} k_2 \dfrac{f_{max} - f'}{f_{max} - f_{avg}}, & f' \geq f_{avg} \\ k_4, & f' \leq f_{avg} \end{cases} \tag{33}$$

where, f_{max} is the maximum value of population fitness; f_{avg} is the mean value of population fitness; f is the larger value of fitness of two individuals at the crossover; f' is the fitness value of the individual performing the mutation operation; k_1, k_2, k_3, and k_4 are all constants, and it is guaranteed that $k_2 > k_1$, $k_4 > k_3$.

V. SIMULATION AND EXPERIMENT

A. SIMULATION AND TEST OF SVPWM ALGORITHM

The SVPWM motor control algorithm is modelled and simulated by MATLAB software to verify the feasibility of the let-off and take-up control strategy. When the DC bus voltage is 310V and the switching period of the inverter is the frequency fs = 1/Ts = 10kHz, of the input PWM wave. Fig. 14 shows the three-phase modulation waveform of the SVPWM output. The shape is saddle-shaped. Each term is formed by the superposition of the sine wave and its third harmonic, which can improve the power supply voltage's utilization rate and restrain the harmonic component's output.

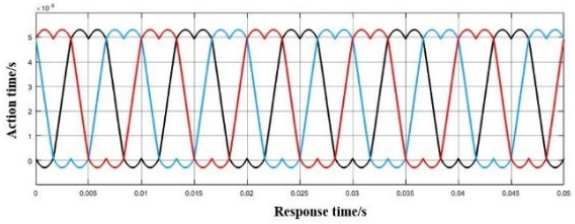

FIGURE 14. Three-phase modulated waveform of SVPWM output.

A hardware platform is constructed to test the motor without load to verify further the driving ability of the SVPWM algorithm and the designed hardware circuit.

The let-off and take-up control module detects the rotational speed in the testing process and obtains the rotational speed detection result shown in Fig. 15. As shown in Fig. 15, the measured motor speed fluctuates, basically remaining at 50 rpm with an error of 0.84%, which is consistent with the expected results.

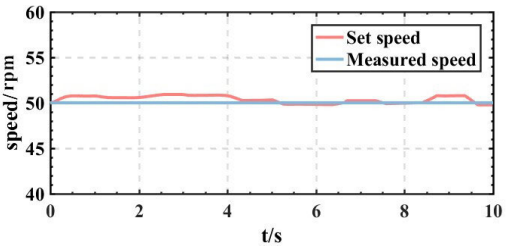

FIGURE 15. Rotational speed test result.

B. TENSION CONTROL ALGORITHM TEST

In the GA-BP FNN algorithm implementation process, the first iteration is performed using the GA. When the maximum number of iterations is reached or the fitness function is optimal, the optimal initial parameters are passed to the BP. Then, training is performed using the BP algorithm to obtain the optimal FNN finally. Therefore, the population number N of the genetic algorithm is set to 50 to generate the initial population, and the maximum number of evolutionary generations G = 30 is set. FNN is trained by the BP algorithm and GA-BP algorithm respectively, and the number

of training is set to 500. The comparison of error convergence is shown in Fig. 16.

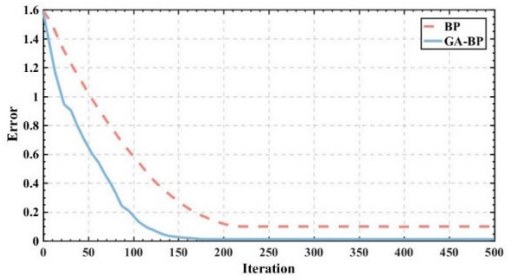

FIGURE 16. Comparison of training error convergence between BP and GA-BP.

It can be seen that the early genetic algorithm plays a major role in searching parameters in the global range. Therefore, the GA-BP algorithm has a fast convergence speed at the beginning of the iteration. After 25 iterations, the training error decreases obviously. At this time, the range of parameters to be optimized is basically determined. Then continue to use the BP algorithm for training. After 150 iterations, the training error tends to be stable. At the end of the iteration, GA-BP can converge to higher accuracy. In contrast, the convergence speed of the BP algorithm is slower than the GA-BP algorithm, reaches a stable state after about 200 iterations, and the convergence accuracy is lower than the GA-BP algorithm. The fuzzy inference rules trained by the GA-BP algorithm are shown in Fig. 17.

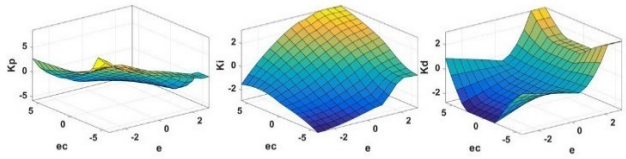

FIGURE 17. Fuzzy reasoning rules.

Tension experiments are performed on the trained model, where the input signal is a step signal of amplitude 10 and the output is the value calculated after PID and fuzzy PID and GA-BP FNN. In the process of experiments, a step disturbance with amplitude 2 and a triangular wave interference with amplitude 2 are added, respectively, corresponding to the influence of beating-up motion and shedding motion on warp tension. After running the software, the control effect comparison diagram shown in Fig. 18 is obtained.

It can be seen from Fig. 18 that GA-BP FNN reaches a stable state in about 2ms, fuzzy PID control reaches a stable state in about 4ms, while traditional PID control reaches a stable state in about 6ms; The fluctuation peak of GA-BP FNN is about 10.15, that of fuzzy PID control is about 11.5, and that of traditional PID control is about 13; after introducing disturbance, GA-BP FNN control is also better than fuzzy PID control in stability.

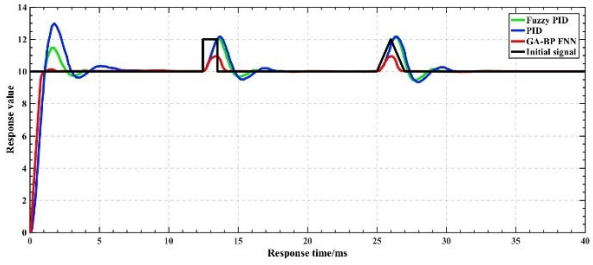

FIGURE 18. PID, Fuzzy-PID and GA-BP FNN step response waveforms.

C. WEAVING EXPERIMENT ON RAPIER LOOM

QJH910 rapier loom is used to test the system. The test experiment is carried out when the spindle speed is 300 rpm. The traditional PID control, fuzzy PID control, and GA-BP FNN control are tested. In the test of traditional PID control, the set value of tension is 110kg, the PID proportional gain K_p is 100%, the integration time is 500ms, the differential control is not used, and the set PID sampling time is 200ms. In fuzzy PID control, the parameters of PID are adjusted online through the fuzzy PID control algorithm. To facilitate comparison, the spindle speed, tension setting value and PID sampling time in fuzzy PID control are consistent with those in traditional PID control. The tension fluctuation curve and error are shown in Fig. 19 and 20. It can be seen that the tension fluctuates wildly during PID control, and the error range is about [- 2.8, + 2.6]. In fuzzy PID control, the tension fluctuation is less than PID control, and the error range is about [− 0.8, + 1.0]. Under the control of the GA-BP FNN algorithm, the tension fluctuation is very small, the tension is the most stable, and the error range is about [− 0.2, + 0.5].

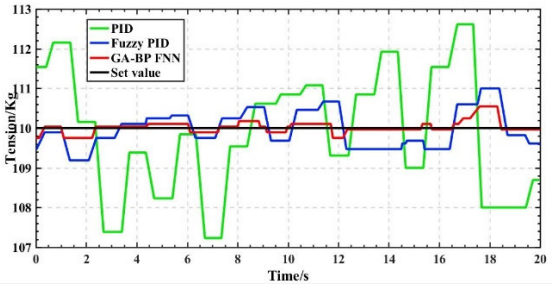

FIGURE 19. PID, Fuzzy PID and GA-BP FNN tension fluctuation curves.

The constant tension weaving experiment shows that both fuzzy PID control and GA-BP FNN control can make the tension fluctuate less near the set value. However, with the increase of loom speed, the interference factors of the warp are also increasing, and the control of tension becomes more difficult. Therefore, it is necessary to experiment with the two control methods at different weaving speeds. The two algorithms are tested at six different speeds, and the tension measurement results are shown in Fig. 21.

According to the tension measurement data in Fig. 21, the tension setting value is 110kg. Under the fuzzy PID

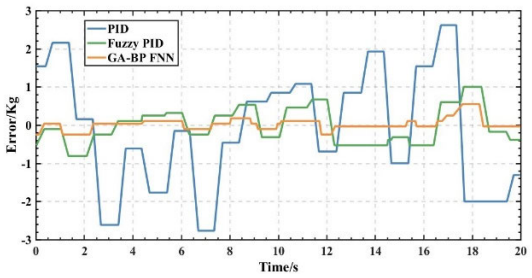

FIGURE 20. PID, Fuzzy PID and GA-BP FNN tension error.

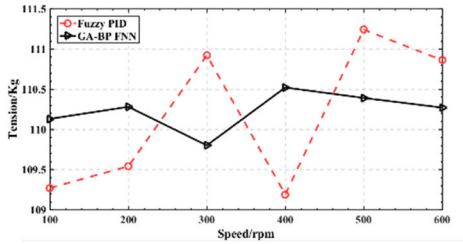

FIGURE 21. Fuzzy PID and GA-BP FNN tension curves at different speeds.

algorithm, the relative tension error is stable within 1.1%. Under the GA-BP FNN algorithm, the tension fluctuation is relatively small and the relative error is stable within 0.47%. The stability of the GA-BP FNN algorithm is better than that of the fuzzy PID algorithm, which can effectively improve the stability of tension control and ensure fabric quality.

VI. CONCLUSION

This paper introduces the design of the control system of the let-off and curling system of a rapier loom and the realization of the hardware circuit. Through the detailed analysis of the action sequence of the whole process and the operation mechanism of the let-off and take-up mechanism, the dynamic relationship between warp tension and let-off and take-up motion is established. On this basis, a tension control scheme based on process interval sampling is proposed. According to the time-varying, nonlinear and multi-coupling characteristics of warp tension, based on PID control, by introducing fuzzy inference and neural network fusion theory, a tension closed-loop control system based on T-S fuzzy neural network is designed. The fuzzy rules are extracted through the neural network's learning ability so that the fuzzy system has the generalization ability. While retaining the inherent accuracy of PID control, it enhances the adaptive ability, learning ability and robustness of the PID controller. The introduction of genetic mechanism and backpropagation fusion algorithm overcomes the defects of long training time and insufficient global optimization ability of FNN. The design of the let-off and take-up hardware circuit based on embedded technology improves the control system's integration and flexibility, reducing the functional redundancy, and the SVPWM algorithm improves the system's response speed and driving ability. Finally, PID control, fuzzy PID control and fuzzy neural network control algorithms are compared through simulation and weaving experiments. The driving performance of the let-off and take-up circuit is verified. The results show that the fuzzy neural network can effectively reduce the fluctuation of warp tension and improve the stability of warp control. At the same time, the driving circuit has good dynamic characteristics and stable driving capability, which ensures the constant let-off amount. The system provides a possible solution to improve the stability of warp tension control of rapier loom and improve fabric quality. At the same time, it also provides references for researchers engaged in relevant textile equipment research.

REFERENCES

[1] M. Gao, W. Wang, Y. Zeng, Z. He, and C. Gao, "A rapier loom HMI system based on an easy cross-platform GUI software," in *Proc. 6th Int. Conf. Instrum. Meas., Comput., Commun. Control (IMCCC)*, Jul. 2016, pp. 874–878.
[2] C. Dong, "Discussion on the development trend of rapier loom," *Mech. Manage. Develop.*, vol. 1, pp. 41–42, Jan. 2012.
[3] L. Ma, Y. Zhao, L. Sun, and F. Jia, "Review and prospect of world textile technology," *China Textile Leader*, vol. 1, pp. 29–47, Jan. 2017.
[4] W. Cao, "Research on fuzzy control strategy of constant tension in yarn winding system," *Adv. Textile Technol.*, vol. 26, no. 2, pp. 80–84, 2018.
[5] X. Lu and J. Yang, "Study on calculation and control algorithm of warp tension of multi-shaft mechanism of carbon fiber multi-layer diagonal loom," *Fiber Reinforced Plastics/Compos.*, vol. 12, no. 14, pp. 14–18, 2017.
[6] W. Liu, X. Wu, X. Du, G. Xu, and S. Wang, "Tension networked control strategy for carbon fiber multilayer diagonal loom," *IEEE Access*, vol. 8, pp. 32280–32289, 2020.
[7] Y. Z. Wang, H. Jing, and H. W. Chen, "Research on tension control model for traction and take-up system of loom," *Adv. Mater. Res.*, vol. 627, pp. 444–448, Dec. 2012.
[8] F. Q. Yang, P. C. Wang, and D. G. Chan, "Application of linear interpolation fuzzy controller in tension control system of yarn," *J. Mech. Electr. Eng.*, vol. 32, no. 11, pp. 1494–1497, Nov. 2015.
[9] M. Ni, P. Li, and L. Yan, "Schemes selection of warp tension measurement and control based on fuzzy multiple-attribute group decision making," *J. Donghua Univ. Natural Sci. Ed.*, vol. 40, no. 3, pp. 282–287, 2014.
[10] G. Xu, R. Zhou, W. Liu, and F. Hao, "The equivalent sliding mode tension control of carbon fiber multilayer diagonal loom," *Int. J. Control, Autom. Syst.*, vol. 17, no. 7, pp. 1762–1769, Jul. 2019.
[11] Z. Peng, M. Guang, and C. Zhou, "Tension and velocity decentralized control of let-off system," *J. Textile Res.*, vol. 32, no. 10, pp. 127–133, 2011.
[12] A. Sala, "On the conservativeness of fuzzy and fuzzy-polynomial control of nonlinear systems," *Annu. Rev. Control*, vol. 33, no. 1, pp. 48–58, Apr. 2009.
[13] L. Li, J. Yang, Y. Zhao, Y. Liu, and L. Cong, "The application of fuzzy-PID and multi-neuron adaptive PID control algorithm in the control of warp tension," in *Proc. 2nd Int. Conf. Comput. Eng. Technol.*, 2010, pp. V7-678–V7-681.
[14] G. Liu and S. Zhang, "Adaptive PID control for warp tension system based on RBF neural network," *J. Textile Res.*, vol. 29, no. 12, pp. 96–99 and 107, Dec. 2008.
[15] Z. Wang, Q. Zhou, and Q. Shen, "Design of embedded controller integrated let-off, take-up and fuzzing of the towel loom," *J. Donghua Univ. Natural Sci. Ed.*, vol. 42, no. 6, pp. 869–874, 2016.
[16] X. Xu and J. C. Yang, "Warp tension detection method of carbon fiber multilayer loom," *Proc. 2nd Int. Conf. Adv. Mech. Eng. Ind. Inform. (AMEII)*, vol. 73, 2016, pp. 325–329.
[17] H. Zhang, R. Zhang, Q. He, and L. Liu, "Variable universe fuzzy control of high-speed elevator horizontal vibration based on firefly algorithm and backpropagation fuzzy neural network," *IEEE Access*, vol. 9, pp. 57020–57032, 2021.
[18] K. Zheng, Q. Zhang, Y. Hu, and B. Wu, "Design of fuzzy system-fuzzy neural network-backstepping control for complex robot system," *Inf. Sci.*, vol. 546, pp. 1230–1255, Feb. 2021.

[19] J. Tang, F. Liu, W. Zhang, R. Ke, and Y. Zou, "Lane-changes prediction based on adaptive fuzzy neural network," *Expert Syst. Appl.*, vol. 91, pp. 452–463, Jan. 2018.
[20] R. Xie, X. Wang, Y. Li, and K. Zhao, "Research and application on improved BP neural network algorithm," in *Proc. 5th IEEE Conf. Ind. Electron. Appl.*, Jun. 2010, pp. 1462–1466.
[21] J. Bai, Q. Teng, and T. Liu, "Comparison of control strategies for permanent magnet synchronous motor," *Control Eng. China*, vol. 22, no. 3, pp. 490–494, May 2015.
[22] P. Ramasamy and V. Krishnasamy, "SVPWM control strategy for a three phase five level dual inverter fed open-end winding induction motor," *ISA Trans.*, vol. 102, pp. 105–116, Jul. 2020.
[23] S. Lakhimsetty, N. Surulivel, and V. T. Somasekhar, "Improvised SVPWM strategies for an enhanced performance for a four-level open-end winding induction motor drive," *IEEE Trans. Ind. Electron.*, vol. 64, no. 4, pp. 2750–2759, Apr. 2017.
[24] L. Wu, J. Li, Y. Lu, and K. He, "Strategy of synchronized SVPWM for dual three-phase machines in full modulation range," *IEEE Trans. Power Electron.*, vol. 37, no. 3, pp. 3272–3282, Mar. 2022.
[25] G. Chen, J. Zhang, and Q. Zhou, "Design of weft-insertion system of three-dimensional loom based on servo-cylinder," *J. Textile Res.*, vol. 34, no. 2, pp. 146–150, 2013.
[26] X. Zhou, Y. Gu, and Z. Wu, "Parameter characteristics of weft insertion mechanism of rapier loom capable of simply adjusting Reed width," *J. Textile Res.*, vol. 40, no. 9, pp. 173–179, 2019.
[27] X. Wang, R. Na, and N. Liu, "Simulation of PMSM field-oriented control based on SVPWM," in *Proc. IEEE Vehicle Power Propuls. Conf.*, Sep. 2009, pp. 7407–7411.
[28] A. Alouane, A. B. Rhouma, M. Hamouda, and A. Khedher, "Efficient FPGA-based real-time implementation of an SVPWM algorithm for a delta inverter," *IET Power Electron.*, vol. 11, no. 9, pp. 1611–1619, Aug. 2018.
[29] H.-K. Jia, L.-D. Yu, Y.-Z. Jiang, H.-N. Zhao, and J.-M. Cao, "Compensation of rotary encoders using Fourier expansion-back propagation neural network optimized by genetic algorithm," *Sensors*, vol. 20, no. 9, p. 2603, May 2020.
[30] T. Shen, J. Chang, and Z. Liang, "Swarm optimization improved BP algorithm for microchannel resistance factor," *IEEE Access*, vol. 8, pp. 52749–52758, 2020.

YANJUN XIAO received the bachelor's degree in industrial automation and the master's degree in machine manufacturing and automation from the Hebei University of Technology. He currently works as a Professor at the School of Mechanical Engineering, Hebei University of Technology. His main research interests include waste heat recovery and industrial control.

XIAOLIANG WANG is currently pursuing the master's degree with the Hebei University of Technology. His major is electronic information. His research interests include embedded technology and intelligent control.

FURONG HAN graduated from the Hebei University of Technology, in 2020. She is currently pursuing the master's degree with the Hebei University of Technology. Her research interests include intelligent control and health management.

PING LIU graduated from the Hebei University of Technology, in 2020. She is currently pursuing the master's degree with the Hebei University of Technology. Her research interests include intelligent control and health management.

YUNFENG JIANG is currently pursuing the Ph.D. degree. He is also an Associate Researcher. His research interests include new perception and intelligent control.