



Article An Improved PID Controller for the Compliant Constant-Force Actuator Based on BP Neural Network and Smith Predictor

Guojin Pei ^{1,2,3}, Ming Yu ^{1,2}, Yaohui Xu ^{1,2}, Cui Ma ^{1,2}, Houhu Lai ^{1,2}, Fokui Chen ^{1,2} and Hui Lin ^{1,2,*}

- ¹ Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China; gj.pei@siat.ac.cn (G.P.); ming.yu@siat.ac.cn (M.Y.); yh.xu@siat.ac.cn (Y.X.); cui.ma@siat.ac.cn (C.M.); hh.lai@siat.ac.cn (H.L.); fg.chen@siat.ac.cn (F.C.)
- ² Shenzhen Key Laboratory of Precision Engineering, Shenzhen 518055, China
- ³ College of Mechanical and Transportation Engineering, China University of Petroleum, Changping, Beijing 102200, China
- * Correspondence: hui.lin@siat.ac.cn

Abstract: A compliant constant-force actuator based on the cylinder is an important tool for the contact operation of robots. Due to the nonlinearity and time delay of the pneumatic system, the traditional proportional–integral–derivative (PID) method for constant force control does not work so well. In this paper, an improved PID control method combining a backpropagation (BP) neural network and the Smith predictor is proposed. Through MATLAB simulation and experimental validation, the results show that the proposed method can shorten the maximum overshoot and the adjustment time compared with traditional the PID method.

Keywords: compliant force control; PID; neural network; Smith predictor

1. Introduction

With the continuous development of industrial automation, the application of robots is becoming more extensive [1]. Robots are required to have the capabilities of precise force perception and force control in contact operations such as grinding, polishing and assembly, as subtle changes in the contact force will affect processing quality [2,3]. In recent years, a number of compliant constant-force actuators have been developed based on gasbag [4], voice coil motor [5], cylinder [6–9], artificial muscle [10,11] and special mechanical structures [12,13]. Among them, the pneumatic system based on the cylinder is the most widely used because air has good compliance. However, the pneumatic system is a complex nonlinear system, due to the compressibility of air and the inevitable static friction [14]. Therefore, achieving precise force control in the pneumatic system is both practical and challenging.

In recent years, intelligent control algorithms such as fuzzy logic, neural networks and expert systems have been developed rapidly, providing new ideas for the design of a pneumatic force control system [15–21]. The traditional proportional–integral–derivative (PID) method is a classic controller that was used for years because of its simplicity and effectiveness. However, with the requirement of low force control error in modern industry, it is difficult for a traditional PID controller to achieve the expected control quality indices in such complex nonlinear systems. Scholars made great efforts in the combination of intelligent control and traditional PID control, especially the combination of a backpropagation neural network (BPNN) and a PID controller. Kang et al. applied the BPNN-PID control method to the pneumatic force control system and demonstrated that the controller had better robustness and control performance [17–19].

These intelligent controllers have improved the performance of force control in the pneumatic nonlinear system to some extent. Nevertheless, besides the nonlinear property, there is a pure time delay link in the pneumatic systems based on pneumatic cylinders [22].



Citation: Pei, G.; Yu, M.; Xu, Y.; Ma, C.; Lai, H.; Chen, F.; Lin, H. An Improved PID Controller for the Compliant Constant-Force Actuator Based on BP Neural Network and Smith Predictor. *Appl. Sci.* **2021**, *11*, 2685. https://doi.org/10.3390/ app11062685

Academic Editor: Manuel Armada

Received: 16 December 2020 Accepted: 12 March 2021 Published: 17 March 2021

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



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). In the implementation of such stabilizing feedback control, even a small time delay may cause destabilization of the controller [23], because the time delay makes the controlled variables unable to reflect the disturbance of the system in time, which will produce a larger overshoot and longer adjustment time [24].

In some other control systems, such as chemical engineering, oil refining, metallurgy and heat engineering process control systems, the Smith predictor has been used effectively for compensating for the pure time delay [25,26]. For a given signal, the Smith predictor estimates the dynamic characteristics of the system under disturbance in advance and then compensates the time delay error to reduce the overshoot of the system and shorten the adjustment process [27]. However, the Smith predictor depends on the system model, so its performance is very sensitive to modeling errors. Pneumatic force control systems are usually complex and time-varying, so it is difficult to build accurate mathematical models. As a result, the Smith predictor is rarely used in such systems to the best of our knowledge.

In order to improve the modeling accuracy and make the Smith predictor suitable for use in a pneumatic force control system, this paper brings the neural network algorithm into the structure of the Smith predictor. Moreover, the improved Smith predictor is connected with the BPNN-PID controller to optimize the overall control performance. Therefore, the structure of this improved controller (backpropagation neural network-Smith predictorproportional–integral–derivative (BPNN-SP-PID)) proposed in this paper is mainly divided into two parts. One part is to combine the artificial neural network (ANN) with the PID controller. An online learning backpropagation neural network ANN1 is taken to adjust the PID parameters adaptively to improve the dynamic performance of the controller. The other part is to combine the artificial neural network with the Smith predictor. A neural network ANN2 is used for identifying the nonlinear model of the controlled object to improve its modeling accuracy, so that the Smith predictor can compensate for the pure time delay well. The integration of a BPNN-PID controller and the Smith predictor improves the robustness and speed of the pneumatic force control system.

The rest of this paper is organized as follows. Section 2 introduces the system structure and model of the compliant constant-force actuator. Section 3 introduces the design of the BPNN-SP-PID controller. Section 4 shows the simulation and experimental results and gives a discussion. Section 5 gives the conclusive points.

2. System Description

As Figure 1 shows, the pneumatic constant force actuator mainly included a tilt sensor (KELAG Kabel variante kas90), a cylinder (SMC MGPM20-30Z), a force sensor (LH Z05A-200N), a base, a proportional pressure regulator (Festo VPPE-3-1-1/8-10-010-E1) and an electromagnetic valve (Parker A05PS25X-1s). Among them, the base was connected with the robot by bolts, and the force sensor had its thread for connecting the workpiece.



Figure 1. Compliant constant-force actuator structure diagram.

The pneumatic system control diagram is shown in Figure 2. The force sensor fed back the contact force F to the electrical signal Uf in real time, and the tilt sensor was used for obtaining the attitude signal Ut of the actuator. The signals were transmitted to the

host computer through the data acquisition (DAQ) board. Then, the host computer ran the controller and adjusted the control voltage Uc of the proportional pressure regulator, making the output force F meet the set value. It also controlled the on and off state of the electromagnetic valve to change the direction of the force. As a result, the system had force perception and force control capabilities.



Figure 2. Pneumatic system control diagram.

The pneumatic system has nonlinear factors that are difficult to quantify [13]. To understand the characteristics of the system intuitively and to facilitate the following simulations, a linear approximate model of the system needed to be obtained first.

The mathematical model G(s) of the system represents the corresponding relationship between the control voltage of the proportional pressure regulator Uc(s) and the output force of the actuator F(s), including three links: the control voltage Uc(s) to the output pressure of the proportional pressure regulator Pu(s); the Pu(s) to the input pressure of the cylinder action chamber Pd(s); and the Pd(s) to the output force F(s).

According to the device structure above, it is divided into three parts to calculate the mathematical model. The first part is the proportional pressure regulator. The schematic diagram of its internal structure is shown in the Figure 3. The following equation is the force balance equation of the proportional pressure regulator spool:

$$P_{pv}A_1 + P_{dv}A_2 + mg - P_{dv}A_1 - P_{sv}A_2 - c\dot{x}_v - k_f(x_v + x_0) - F_c = 0$$
(1)

where P_{pv} is the air pressure in the pilot cavity; P_{dv} is the output pressure of the proportional pressure regulator; P_{sv} is the supply pressure of the proportional pressure regulator; A_1 is the effective area of the diaphragm; A_2 is the cross-sectional area of the valve chamber; m is the self-weight of the valve spool; x_v is the displacement of the valve spool; x_0 is the initial deformation of the spring; c is the viscous damping coefficient; k_f is the equivalent spring stiffness; and F_c is the Coulomb friction.

Ignoring the influence of the Coulomb friction, we can write Equation (1) in incremental form and perform a Laplace transform:

$$\left(cs+k_f\right)\Delta x_v(s) - \Delta P_{pv}(s)A_1 + \Delta P_{dv}(s)(A_1 - A_2) = 0$$
⁽²⁾

According to the properties of the proportional pressure regulator, the spool displacement x_v and the air pressure P_{pv} of the pilot cavity can be approximated as proportional to the control voltage U_c .



Figure 3. The internal structure of the proportional pressure regulator. (SUP = supply; EXH = exhaust; and OUT = output).

Then, we can obtain the following relationship:

$$\frac{\Delta P_u(s)}{\Delta U_c(s)} = \frac{K_1 c s + K_1 k_f - K_2 A_1}{A_1 - A_2} = k_1 s + k_2 \tag{4}$$

where K_n , k_n (n = 1, 2, 3...) are all undetermined coefficients.

The second part is the gas pipeline.

Assuming that the trachea is circular and the airflow movement is laminar, according to Anderson theory [28], we have

$$\Delta q_m(s) = K_3(\Delta P_u(s) - \Delta P_d(s)) \tag{5}$$

where q_m is the mass flow rate.

The gas enters the cylinder through the proportional pressure regulator and the gas pipe. Assume that the gas in the cylinder is ideal and satisfies the ideal gas law:

$$P_d = \rho_d R_c T_d \tag{6}$$

where R_c is the proportionality coefficient; ρ_d is the density; T_d is the temperature; and V_d is the volume of the gas.

Assuming that the temperature in the gas circulation process satisfies the adiabatic process, q_m is equal to the rate of change of the gas mass m in the cavity. Combining the above, we get

$$q_m = \frac{1}{k} \frac{V_d}{R_c T_d} \frac{dP_d}{dt} \tag{7}$$

where *k* is an undetermined coefficient.

Write Equation (7) in incremental form and perform a Laplace transformation:

$$\Delta q_m(s) = \frac{1}{k} \frac{V_d}{R_c T_d} s \Delta P_d(s) = K_4 s \Delta P_d(s)$$
(8)

(3)

Combine Equations (5) and (8) to get

$$\frac{\Delta P_d(s)}{\Delta P_u(s)} = \frac{K_3}{K_4 s + K_3} = \frac{1}{k_3 s + 1} \tag{9}$$

The third part is the cylinder.

The force analysis of the cylinder is carried out according to Newton's second law, and the friction is ignored:

$$Ms^{2}\Delta Y(s) + Cs\Delta Y(s) + \Delta F(s) = \Delta P_{d}(s)A_{3}$$
⁽¹⁰⁾

where *M* is the total mass of the cylinder rod; *C* is the viscous damping coefficient; *F* is the output force; A_3 is the cylinder cross-section; and *Y* is the displacement of the cylinder.

The force causes slight deformation of the workpiece, and the actuator passively generates displacement *y*. Set the equivalent stiffness coefficient to *K*:

$$\Delta F(s) = K \Delta Y(s) \tag{11}$$

Combine Equations (10) and (11) to get

$$\frac{\Delta F(s)}{\Delta P_d(s)} = \frac{KA_3}{Ms^2 + Cs + K} = \frac{k_7}{k_4s^2 + k_5s + k_6}$$
(12)

According to the above derivation, the transfer function of each link is listed in Figure 4.

$$\underbrace{\text{Uc(s)}}_{k_1s+k_2} \xrightarrow{\text{Pu(s)}} \underbrace{\frac{1}{k_3s+1}}_{k_3s+1} \xrightarrow{\text{Pd(s)}} \underbrace{\frac{k_7}{k_4s^2+k_5s+k_6}}_{k_5s+k_6} \xrightarrow{\text{F(s)}}$$

Figure 4. The transfer function of the system.

Therefore, the open-loop mathematical model $G_0(s)$ of the system can be obtained as follows:

$$G_o(s) = \frac{k_7(k_1s + k_2)}{(k_3s + 1)(k_4s^2 + k_5s + k_6)}$$
(13)

where k_n (n = 1, 2, ..., 7) in Figure 4 and Equation (13) represents undetermined coefficients. It can be seen that the mathematical model of the system has three poles and one zero.

The undetermined coefficients are identified through the gray box experimental method. The system is loaded with an input signal, and the corresponding output signal is collected over time. When the loaded signal is a step signal, the output signal always takes ~70 ms before it starts to respond, as shown in Figure 5, so a time delay link needs to be added to the mathematical model. Afterward, the MATLAB system identification tool is used for analyzing the relationship between the collected input signal and output signal [29]. As a mathematical model with three poles, one zero and a pure time delay link are used for identification. The approximate mathematical model including the pure time-delay *G*(*s*) is obtained:

$$G(s) = \frac{2s + 15.6}{s^3 + 15.8s^2 + 130.7s + 350.7} e^{-0.07s}$$
(14)

The step response curve of the mathematical model is calculated in MATLAB Simulink and compared with the actual curve for model validation. It can be seen from Figure 5 that the identified mathematical model matches the actual one well. The approximate mathematical model will be used in the simulation of Section 4.



Figure 5. Comparison of the open-loop step response between the mathematical and actual models.

3. Controller Design

The structure of the improved PID control method proposed in this paper is shown in Figure 6. The BP neural network-Smith predictor-PID (BPNN-SP-PID) controller was divided into two parts. One part was the BPNN-PID controller. ANN1 could realize selfoptimization of the PID parameters and improve the control performance. The other part was the Smith predictor. The Smith predictor was used for reducing the influence of the pure time delay [27]. ANN2 was used for identifying the open-loop model of the controlled object Go(s), eliminating the pure time delay. Higher identification accuracy brought by ANN2 could make the Smith predictor compensate the time delay more effectively. The two parts are described in detail below.



Figure 6. Backpropagation neural network-Smith predictor-proportional–integral–derivative (BPNN-SP-PID) controller structure diagram.

3.1. BPNN-PID

The neural network has a strong self-adaptive ability and can be combined well with other control methods [18]. On the basis of PID control, in order to optimize the performance of the controller, the online learning type BP neural network ANN1 was applied.

The BPNN-PID controller consisted of two parts: the PID controller and the BP network ANN1. The PID controller directly controlled the controlled object in the closed loop, and the BPNN adjusted the three parameters of the PID controller in real time according to the error function J.

The structure of the neural network ANN1 is shown in Figure 7.



Figure 7. Schematic diagram of the neural network ANN1's structure, part of the BPNN-PID controller depicted in Figure 6.

The input and output of the network are

$$\begin{bmatrix} O_i(t) = \begin{bmatrix} R(t) & E(t) & F(t) \end{bmatrix}^T \\ net_k(t) = \begin{bmatrix} k_p & k_i & k_d \end{bmatrix}^T$$
(15)

where *t* is the discrete time series (t = 1, 2, 3...); R(t) is the set force at time *t*; E(t) is the error at time *t*; F(t) is the actual output force at time *t*; and k_p , k_i , k_d indicate the proportional, integral and differential parameters of the PID controller, respectively.

The activation function of the hidden layer was the tanh function. Since PID parameters cannot be negative, the activation function of the output layer was taken as a sigmoid function. Based on the previous paper [30], other parameters of the neural network were determined, as shown in Table 1.

Table 1. Parameters of neural network ANN1.

| Туре | Value |
|----------------------------------|------------------------------|
| Learning type | Online |
| Learning rate (η) | 0.3 |
| Inertia index (α) | 0.15 |
| Number of hidden layer neurons | 8 |
| Hidden layer activation function | Tanh |
| Output layer activation function | Sigmoid |
| Initialization weight method | 0.03rands(n, 1) ¹ |

 1 rands(n, 1) returns an random matrix of n rows and one column, and the range of elements in the matrix is (-1, 1).

The incremental PID control algorithm is used for calculating the control voltage Uc(t):

$$U_{c}(t) = U_{c}(t-1) + \begin{bmatrix} E(t) - E(t-1) & E(t) & E(t) - 2E(t-1) + E(t-2) \end{bmatrix} \cdot net_{k}(t)$$
(16)

The error function J [18,19] is expressed as

$$J = \frac{1}{2}(R(t) - F(t))^2$$
(17)

According to the gradient descent method, the weights of the output layer and hidden layer of the network are adjusted in turn, and an inertia term $\alpha \Delta \omega (t-1)$ is added to avoid falling into the local minimum:

$$\Delta\omega(t) = -\eta \frac{\partial J(t)}{\partial\omega} + \alpha \Delta\omega(t-1)$$
(18)

In Equation (18), $\Delta \omega(t)$ is the weights adjusting value; η is the learning rate; and α is the coefficient of inertia.

In application, this BPNN-PID control algorithm could be roughly divided into the following steps, as shown in Figure 8:

- 1. Determine the structure of the BP neural network. The parameters of the neural network are listed in Table 1;
- 2. Give the initial value of each layer's weights;
- 3. The input data R(t), F(t) is sampled, and the error E(t) is calculated;
- 4. Calculate the input and output values of each layer of neurons, and the output values of the output layer are PID parameters k_p , k_i , k_d ;
- 5. Calculate the control voltage *Uc*(*t*) based on the incremental PID control algorithm;
- 6. Perform online learning of neural networks, and adjust the weights of the hidden layers and the output layer;
- 7. Return to Step 3 for the next round of training.



Figure 8. ANN1 training flowchart.

3.2. Smith Predictor

To optimize the response speed of the control system, the Smith predictor is added for predictive compensation. Its structure is shown in Figure 6, and the transfer function $G_m(s)$ of the Smith predictor compensator is expressed as

$$G_m(s) = G_o(s) (1 - e^{-\tau s})$$
(19)

where $G_o(s) \cdot e^{-\tau s}$ is the open-loop transfer function of the controlled object considering the time delay.

When the Smith predictor is not added, the closed-loop transfer function of the system is

$$G(s) = \frac{G_c(s)G_o(s)e^{-\tau s}}{1 + G_c(s)G_o(s)e^{-\tau s}}$$
(20)

where $G_c(s)$ is the transfer function of the controller.

After adding the Smith predictor, the closed-loop transfer function of the system changes:

$$G(s) = \frac{G_c(s)G_o(s)}{1 + G_c(s)G_o(s)}e^{-\tau s}$$
(21)

By comparison of Equations (20) and (21), it can be found that the Smith predictor uses compensation to separate the pure time delay $e^{-\tau s}$ and $G_o(s)$, which is equivalent to putting the pure time delay of the controlled object outside the control closed loop.

As the Smith predictor depends on the prior system model, a BP neural network shown in Figure 9 was used for identifying the system to improve modeling accuracy.



Figure 9. Schematic diagram of the neural network ANN2 structure, part of the BPNN-PID controller depicted in Figure 6.

The controlled object was a single-input single-output (SISO) nonlinear system, which can be described as follows:

$$F(t+1) = f[F(t), \dots, F(t-n), U_c(t), \dots, U_c(t-m)]$$
(22)

where F(t) and Uc(t) are the input and output of the object at time t, respectively; m and n mark the orders of F(t) and Uc(t), respectively; and $f[\bullet]$ represents a nonlinear function.

The input $O_i^1(t)$ of the network is

$$O_i^1(t) = \begin{cases} U_c(t-i+1), 0 \le i \le m+1\\ F(t-i+m-2), m+2 \le i \le m+n+2 \end{cases}$$
(23)

The output $net_k(t)$ is

$$net_k(t) = F(t+1) \tag{24}$$

The neural network ANN2 also used the BP algorithm, and its structural parameters were determined as shown in Table 2.

Table 2. Parameters of neural network ANN2.

| Туре | Value | |
|----------------------------------|------------------|--|
| learning type | Offline | |
| Learning rate (η) | 0.3 | |
| Inertia index (α) | 0.15 | |
| Number of hidden layer neurons | 15 | |
| Hidden layer activation function | tanh | |
| Output layer activation function | tanh | |
| Initialization weight method | 0.03rands (n, 1) | |
| Time orders m,n | 4 | |

The neural network ANN2 was trained offline. As Figure 10 shows, the main training steps of the neural network ANN2 are listed as follows:

- 1. Determine the structure of the neural network, including the number of hidden layers, neuron activation functions and training samples;
- 2. Take a small initial value for the weight;
- 3. Import samples for training;
- 4. Calculate the input and output of each layer of neurons;
- 5. For each iteration, calculate the weight adjustment of the hidden layer and the output layer;
- 6. Check whether the network training error is less than the error tolerance. If it meets the requirements, go to Step 7, and if not, go to Step 8;
- 7. Solidify and save each weight. The training is successful, and you can call it through Labview or MATLAB Simulink;
- 8. Check whether the current iteration number exceeds the upper limit. If not, return to Step 3 for further training; otherwise, the training failed. Return to Step 1 to restart training.



Figure 10. ANN2 training flowchart.

4. Results and Discussion

In this section, the performance of the proposed BPNN-SP-PID method is validated through simulation and experiments. First, according to the mathematical model deduced in Section 2, a simulation block diagram of the system was built in MATLAB Simulink. Three types of controllers (PID, BPNN-PID and BPNN-SP-PID) were tested and compared. Then, an experimental prototype was built, and the force response curves under step input and sinusoid input were collected. Finally, a comprehensive discussion was made based on the results of the simulation and experiment.

4.1. Simulation Results

In the block diagram shown in Figure 11, ANN2 is a neural network that describes the controlled object. It was trained and solidified to generate a Simulink module and added to the block diagram. The inside of the BPNN-PID structure was the packed neural network ANN1 and a PID controller. The online learning neural network ANN1 was realized by using the S-function module. There were two kinds of input signals as the setting force: a step signal with an amplitude of 20 N and a sin signal with an amplitude of 20 N and the three controllers were exported through the simout module.



Figure 11. MATLAB Simulink simulation block diagram.

4.1.1. Step Response

The response curve under step signal input with an amplitude of 20 N is shown in Figure 12.

All three controllers could make the output response meet the target values. However, there were differences at the beginning stage and the approaching stage. At the beginning stage, the BPNN-SP-PID controller acted first, and the reaction time was almost zero. The action time of BPNN-PID and PID was almost the same, taking ~80 ms to start the action. At the approaching stage, the PID method produced the largest overshoot of 6.1% among the three, and the overshoots of BPNN-PID and BPNN-SP-PID were 0.45% and 0.15%, respectively. At the stable stage, the residuals of the three were all close to zero due to the integral part of the PID controller.

By comparison of the three response curves, two phenomena could be observed. First, the use of a BPNN greatly enhanced the robustness of the system. Due to the self-adjusting ability of neural networks, the approaching processes of BPNN-PID and BPNN-SP-PID were more stable than PID, and there was almost no overshoot or oscillation. Second, the

application of the Smith predictor could significantly improve the speed. Though BPNN-PID and BPNN-SP-PID showed good robustness, BPNN-SP-PID had a much faster reaction than BPNN-PID, especially at the beginning stage. This was mainly because the Smith predictor based on ANN2 could fit the system model precisely, resulting in a compensation for the inherent time delay of the system.



Figure 12. Comparison of step responses of the three controllers in the simulation.

4.1.2. Sinusoid Track

A sinusoidal signal with an amplitude of 20 N and frequency of 0.2 Hz was set for the simulation, and the response curves of the three controllers were obtained, as shown in Figure 13.

All three controllers could smoothly follow the setting curve but had different degrees of lag. BPNN-SP-PID produced the smallest lag, followed by BPNN-PID, and the worst was from the PID method. The difference in response time also brought about a difference in control accuracy. The maximum force error of PID reached 3.22 N, and the average error reached 2.09 N. The maximum error of BPNN-PID was 2.65 N, and the average error was 1.68 N. The maximum error of BPNN-SP-PID was 2.63 N, and the average error was 1.59 N.

It can be seen that the BPNN-SP-PID controller had the best dynamic adjustment performance among the three controllers.



Figure 13. Comparison of sinusoid tracking response curves in the simulation.

4.2. Experimental Validation

A prototype of the compliant constant-force actuator was built and mounted on the robot ABB 1200 for the experiment as shown in Figure 14. The whole experimental platform mainly included an air compressor, a control cabinet, a robot, a computer and the constant-force actuator prototype. The DAQ board (Zhongtai USB7660BD) in the control cabinet was used for signal input and output. The control and data acquisition program on the host computer was programmed through Labview, shown in Figure 15. The practical application of the BPNN-SP-PID controller in this system was completed by the joint compilation of Labview and MATLAB [31,32]. In order to reduce the influence of the posture and the surface shape of the workpiece, the posture of the robot arm was kept vertical during the entire experiment, and a flat steel plate was selected as the workpiece.



Air compressor



| | Une-aimensiona | a constant Force Actuator | |
|-------------------------------------|----------------|---------------------------|-------|
| | | Designed By Guojin.Pei | NOTE: |
| Connection status Displacement | Actual Force | | - |
| 10- | 150- | | |
| 5- | 140- | | |
| Tilt angle of X (±90°) | 130- | | |
| 0.21 | 120- | | |
| Tilt angle of Y (±90°) -5 - | 110- | | |
| 0.56 | 100- | | |
| -10 | 90- | | |
| | Ē 80- | | |
| 150 A Land Lande | d \$ 70- | | |
| arget force (N) Actual force (N) | 60 - | | |
| \$5 54.21 START | 50- | | ~ |
| ad weight (N) Identified weight (N) | 40- | | |
| | 30- | | |
| PAUSE | | | |
| | | | |
| Manual load Zloaded EXCIT | 10 | | |
| | 0- | | |

Figure 15. Software interface diagram in Labview.

4.2.1. Step Response

When the input was a step signal with an amplitude of 20 N, the response curve of the output force obtained was as shown in Figure 16.

The experimental results coincided with the simulation results. The BPNN-SP-PID controller was the fastest at the beginning stage, which needed ~30 ms of action time, while BPNN-PID and PID took ~70 ms and ~80 ms, respectively. At the approaching stage, BPNN-SP-PID was still the fastest to reach the set value of 20 N and to enter the ideal

range. The adjustment times of the BPNN-SP-PID, BPNN-PID and PID were 0.16 s, 0.46 s and 0.48 s, respectively. The three controllers also presented different degrees of overshoot. The overshoot of BPNN-SP-PID was the smallest, being only 1.6%, while the overshoots of BPNN-PID and PID were 4.9% and 9.7%, respectively. At the stable stage, the residuals of the three were different from the ideal scenario in the simulation, but the steady state error could also be guaranteed to be within 0.2 N.

By comparison with Section 4.1.1, it can be found that the experimental results were a bit worse than the simulation due to the influence of uncontrollable nonlinear factors under the actual experiment. BPNN-SP-PID still presented the best performance among the three controllers in the experiment, which could significantly improve the reaction speed while maintaining stability.



Figure 16. Comparison of the step responses of three controllers in the experiment.

4.2.2. Sinusoid Track

The input was then changed to the sinusoidal signal with an amplitude of 20 N and frequency of 0.2 Hz, and the response curve is shown in Figure 17. The data collection rate was 1 sampling/10 ms, and the total number of samples was 900 in 9 s. The error data between the set value and the actual value collected in Figure 17a was calculated and statistically analyzed, and the probability distribution of the 900 error data within \pm 3 N is shown in Figure 17b.

The sinusoid curve tracking in the experiment was not as smooth as that in the simulation (Figure 13), and the response curve fluctuated up and down around the target curve. Among the three controllers, the BPNN-SP-PID gave the smoothest response and the smallest fluctuation. The maximum error was 1.88 N, and the average error was only 0.68 N. Meanwhile, the maximum errors of the BPNN-PID and PID controllers were both above 2 N, and the average errors reached 0.74 N and 0.98 N, respectively. The force control accuracy was expected to be ± 1 N in some robot contact operations, such as robot grinding, and BPNN-SP-PID could meet the requirements well.

According to the histogram of the tracking errors, it can be seen that the error distribution of BPNN-SP-PID was concentrated around zero, showing better precision than the other two controllers. The addition of the neural network and Smith predictor made the control system more robust and quicker, resulting in a reduction of the tracking error.



Figure 17. Comparison of actual sinusoid tracking response curves in the experiment. (**a**) Response curves of the three controllers. (**b**) Histogram of the tracking errors.

4.3. Discussion

The specific indicators [33] of the three controllers are listed in Table 3.

It can be learned that there were some gaps between the experiment and simulation results. First, the steady state error in the experiment could not be ignored like that in the simulation because of the environmental noise, the sensor error and other nonlinear factors (such as the motion of the robot, air source instability and cylinder friction). Second, some indicators in the experiment, such as the settling time and the errors of the sinusoid track, were even better than those in the simulation. This was mainly because the simulation started from the linear approximation model, while the experimental model was estimated through experimental data. Despite this, the simulation and experiment results both showed the same result: the BPNN-SP-PID had the best performance among the three controllers.

BPNN-SP-PID and BPNN-PID were obviously better than traditional PID, as the BPNN could adaptively adjust the PID parameters when dealing with the influence of nonlinear factors. The indicator of the maximum overshoot proved that BPNN-SP-PID had the best robustness. On the other hand, it is interesting to find that the performance gap between BPNN-SP-PID and BPNN-PID was different in the step response and sinusoid track. In the case of the step response, the BPNN-SP-PID showed a huge improvement compared with BPNN-PID. The maximum overshoot and settling time of BPNN-SP-PID were reduced by 68% and 65% in the experiment, respectively. In the case of the sinusoid track, however, the BPNN-SP-PID was just a litter better than the BPNN-PID controller. The maximum error and average error of the BPNN-SP-PID were only reduced by 9% and 8% in the experiment, respectively. The reason for this was probably that the setting force in the sinusoid track process was time-varying, which would partially affect the fitting precision of the Smith predictor and weaken the effect of it. A potential solution is adding a neural network identifier to update the ANN2 to further enhance the matching degree of the theoretical model and the actual one.

| | Performance Indicators of Three Controllers | | | |
|---|---|----------|-------------|--|
| Туре | PID | BPNN-PID | BPNN-SP-PID | |
| Simulation | | | | |
| Step response | | | | |
| Steady-state error (N) | ~0 | ~0 | ~0 | |
| Settling time ^{1} (s) | 0.76 | 0.76 | 0.59 | |
| Maximum overshoot (%) | 6.10 | 0.45 | 0.15 | |
| Sinusoid track | | | | |
| Maximum error (N) | 3.22 | 2.65 | 2.63 | |
| Average error (N) | 2.09 | 1.68 | 1.59 | |
| Experiment | | | | |
| Ŝtep response | | | | |
| Steady-state error (N) | 0.18 | 0.12 | 0.12 | |
| Settling time ^{1} (s) | 0.48 | 0.46 | 0.16 | |
| Maximum overshoot (%) | 9.70 | 4.90 | 1.56 | |
| Sinusoid track | | | | |
| Maximum error (N) | 2.70 | 2.06 | 1.88 | |
| Average error (N) | 0.98 | 0.74 | 0.68 | |

 Table 3. Comparison of overall performance indicators of three controllers.

¹ The minimum time required for the system response to reach and maintain a value within 5% of the final value.

5. Conclusions

Aiming at the nonlinearity and time delay of pneumatic constant force actuators, this paper proposes an improved PID control method combining a BPNN and the Smith predictor. An on-line BPNN ANN1 is used for adjusting the PID coefficients intelligently, which can enhance the robustness of a nonlinear system. The Smith predictor, with the aid of off-line neutral network ANN2, compensates the pure time delay, which can improve the adjustment speed of the system. The simulation and experiment results both showed that the BPNN-SP-PID control method was feasible and superior compared with the traditional PID controller.

The matching error between the theoretical model and the actual model is concerned with the control performance. In a simulation, Smith predictor can compensate almost all of the time delay. In practical applications, the posture of the actuator is time-varying, and the system model will also change accordingly. As the offline ANN2 cannot fully fit the time-varying nonlinear system, the time delay compensation effect is not as good as that found in simulation. Adding a neural network identifier to update ANN2 will be beneficial for identifying the actual model in time so that the adaptability of the controller will be stronger. This will be the focus of future work.

Author Contributions: H.L. (Hui Lin) contributed to the main idea of this paper; G.P. performed the simulation and experiments, analyzed the data and wrote the paper; Y.X. and F.C. helped to build the prototype; M.Y. and H.L. (Houhu Lai) helped to design the data acquisition code; C.M. and H.L. (Hui Lin) reviewed and edited the paper. All authors have read and agreed to the published version of the manuscript.

Funding: The presented work was supported by the following grants: National Natural Science Foundation of China Grant No. U1713210 and Shenzhen Fundamental Research Program Grant No. JCYJ20170818163928953.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

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

References

- 1. Berg, A.; Buffie, E.F.; Zanna, L.F. Should We Fear the Robot Revolution? (The Correct Answer is Yes). *J. Monet. Econ.* **2018**, *97*, 117–148. [CrossRef]
- Zhang, X.; Chen, H.; Yang, N. A structure and control design of constant force polishing end actuator based on polishing robot. In Proceedings of the 2017 IEEE International Conference on Information and Automation (ICIA), Macau, China, 18–20 July 2017; pp. 764–768.
- 3. Liu, M.H.; Tolle, H. Force-Controlled Robotic Deburring. IFAC Proc. Vol. 1993, 26, 855–858. [CrossRef]
- 4. Jin, M.; Ji, S.; Pan, Y.; Ao, H.; Han, S. Effect of downward depth and inflation pressure on contact force of gasbag polishing. *Precis. Eng. J. Int. Soc. Precis. Eng. Nanotechnol.* **2017**, 47, 81–89. [CrossRef]
- 5. Mohammad, A.E.K.; Hong, J.; Wang, D. Design of a force-controlled end-effector with low-inertia effect for robotic polishing using macro-mini robot approach. *Robot. Comput. Integr. Manuf.* **2018**, *49*, 54–65. [CrossRef]
- Rouzbeh, B.; Bone, G.M. Optimal Force Allocation and Position Control of Hybrid Pneumatic–Electric Linear Actuators. *Actuators* 2020, 9, 86. [CrossRef]
- Naderer, R.; Ferrara, P.; Rohrhofer, A. Active Handling Apparatus and Method for Contact Tasks. U.S. Patent 9,375,840, 28 June 2016.
- 8. Ning, F.; Shi, Y.; Cai, M.; Wang, Y.; Xu, W. Research Progress of Related Technologies of Electric-Pneumatic Pressure Proportional Valves. *Appl. Sci.* 2017, *7*, 1074. [CrossRef]
- 9. Ben-Dov, D.; Salcudean, S.E. A force-controlled pneumatic actuator. Int. Conf. Robot. Autom. 1995, 11, 906–911. [CrossRef]
- Liu, S.; Wang, F.; Zhang, G.; Liu, Z.; Zhang, W.; Tian, Y.; Zhang, D. A Novel Dual-Drive Soft Pneumatic Actuator with the Improved Output Force. In Proceedings of the International Conference on Intelligent Robotics and Applications, Shenyang, China, 8–11 August 2019; pp. 16–25.
- 11. Robinson, R.M.; Kothera, C.S.; Sanner, R.M.; Wereley, N.M. Nonlinear Control of Robotic Manipulators Driven by Pneumatic Artificial Muscles. *IEEE-ASME Trans. Mechatron.* **2016**, *21*, 55–68. [CrossRef]
- 12. Liu, Y.; Zhang, Y.; Xu, Q. Design and Control of a Novel Compliant Constant-Force Gripper Based on Buckled Fixed-Guided Beams. *IEEE-ASME Trans. Mechatron.* 2017, 22, 476–486. [CrossRef]
- Joo, S.; Waki, H.; Miyazaki, F. On the mechanics of elastomer shear pads for remote center compliance (RCC). In Proceedings of the IEEE International Conference on Robotics and Automation, Minneapolis, MN, USA, 22–28 April 1996; Volume 9, pp. 291–298.
- 14. Fan, J.; Zhong, J.; Zhao, J.; Zhu, Y. BP neural network tuned PID controller for position tracking of a pneumatic artificial muscle. *Technol. Health Care* **2015**, *23*, S231–S238. [CrossRef]
- Guoku, L.; Quanlin, D.; Meilin, S. Design of Control System of Pneumatic Regulating Valves Based on STM32. In Proceedings of the 2018 IEEE 4th International Conference on Control Science and Systems Engineering (ICCSSE), Wuhan, China, 21–23 August 2018.
- 16. Hägglund, T. A friction compensator for pneumatic control valves. J. Process Control 2002, 12, 897–904. [CrossRef]
- 17. Kang, J.; Meng, W.; Abraham, A.; Liu, H. An adaptive PID neural network for complex nonlinear system control. *Neurocomputing* **2014**, 135, 79–85. [CrossRef]
- Zhao, D.; Sun, C.; Wang, Q.; Yang, W. Neural Network Based PID Control for Quadrotor Aircraft. In *IScIDE 2015, Proceedings* of the 5th International Conference on Intelligence Science and Big Data Engineering, Suzhou, China, 14–16 June 2015; Big Data and Machine Learning Techniques; Springer: Berlin/Heidelberg, Germany, 2015; Volume 9243, pp. 287–297.
- 19. Cheng, F.; Fan, K.-C.; Miao, J.; Li, B.-K.; Wang, H.-Y. A BPNN-PID based long-stroke nanopositioning control scheme driven by ultrasonic motor. *Precis. Eng. J. Int. Soc. Precis. Eng. Nanotechnol.* **2012**, *36*, 485–493. [CrossRef]
- Milovanović, M.B.; Antić, D.S.; Milojković, M.T.; Nikolić, S.S.; Perić, S.L.; Spasić, M.D. Adaptive PID control based on orthogonal endocrine neural networks. *Neural Netw.* 2016, *84*, 80–90. [CrossRef] [PubMed]
- 21. Fan, C.; Hong, G.S.; Zhao, J.; Zhang, L.; Zhao, J.; Sun, L. The integral sliding mode control of a pneumatic force servo for the polishing process. *Precis. Eng. J. Int. Soc. Precis. Eng. Nanotechnol.* **2019**, *55*, 154–170. [CrossRef]
- 22. Edjekouane, E.; Riachy, S.; Ghanes, M.; Barbot, J.-P. Homogeneous Control of Pneumatic Cylinders Based on Time Delay Model and Artstein Transformation. *Asian J. Control* **2017**, *19*, 792–798. [CrossRef]
- 23. Guo, F.; Yu, W.; Huang, F. Robustness with respect to small delays for exponential stability of non-autonomous systems. *J. Math. Anal. Appl.* **2003**, *288*, 671–679. [CrossRef]
- 24. Feng, D.-M.; Pan, F.; Han, R.-C. Improved self-adaptive Smith predictive control scheme for time-delay system. In Proceedings of the International Conference on Machine Learning and Cybernetics, Beijing, China, 4–5 November 2002; Volume 1, pp. 463–466.
- Dehghani, A.; Khodadadi, H. Designing a neuro-fuzzy PID controller based on smith predictor for heating system. In Proceedings of the 2017 17th International Conference on Control, Automation and Systems (ICCAS), Jeju, Korea, 18–21 October 2017; pp. 15–20.
- 26. Özbek, N.S.; Eker, İ. Design of an optimal fractional fuzzy gain-scheduled Smith Predictor for a time-delay process with experimental application. *ISA Trans.* **2020**, *97*, 14–35. [CrossRef] [PubMed]
- Fliess, M.; Marquez, R.; Mounier, H. An extension of predictive control, PID regulators and Smith predictors to some linear delay systems. *Int. J. Control* 2002, 75, 728–743. [CrossRef]
- 28. Parr, A. Hydraulics and Pneumatics. Plast. Syst. Eng. 1999, 4, 85–124.

- 29. Abo-Elmagd, M.; Sadek, A.M. Development of a model using the MATLAB System identification toolbox to estimate 222Rn equilibrium factor from CR-39 based passive measurements. *J. Environ. Radioact.* **2014**, *138*, 33–37. [CrossRef] [PubMed]
- 30. Hecht-Nielsen, R. Theory of the backpropagation neural network. In Proceedings of the International 1989 Joint Conference on Neural Networks, Washington, DC, USA, 1 February 1989; pp. 593–605.
- Somwanshi, D.; Bundele, M.; Kumar, G.; Parashar, G. Comparison of Fuzzy-PID and PID Controller for Speed Control of DC Motor using LabVIEW. *Procedia Comput. Sci.* 2019, 152, 252–260. [CrossRef]
- 32. National Instruments. LabVIEW User Manual [Z]; National Instruments: Austin, TX, USA, 1998.
- 33. Dorf, R.C.; Sinha, N.K. Modern control systems. IEEE Trans. Syst. Man Cybern. 1981, 11, 580. [CrossRef]