

Research Article

PSO-RBF Neural Network PID Control Algorithm of Electric Gas Pressure Regulator

Yuanchang Zhong,^{1,2} Xu Huang,¹ Pu Meng,¹ and Fachuan Li¹

¹ College of Communication Engineering, Chongqing University, Chongqing 400044, China

² School of Automation, Chongqing University, Chongqing 400044, China

Correspondence should be addressed to Yuanchang Zhong; zyc@cqu.edu.cn

Received 8 April 2014; Accepted 2 June 2014; Published 24 July 2014

Academic Editor: Zidong Wang

Copyright © 2014 Yuanchang Zhong et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The current electric gas pressure regulator often adopts the conventional PID control algorithm to take drive control of the core part (micromotor) of electric gas pressure regulator. In order to further improve tracking performance and to shorten response time, this paper presents an improved PID intelligent control algorithm which applies to the electric gas pressure regulator. The algorithm uses the improved RBF neural network based on PSO algorithm to make online adjustment on PID parameters. Theoretical analysis and simulation result show that the algorithm shortens the step response time and improves tracking performance.

1. Introduction

With the implementation of the strategy on Western development and west-east gas transmission project, the city gas has developed rapidly. How to use gas for higher energy saving and environmental protection and ensure the safety and reliability of the system has been a key problem and this brings opportunity to gas pressure regulating. Gas pressure regulating is one of the most important parts of city gas pipeline system and the key technology of highly efficient use of natural gas resources. The gas pressure regulating can ensure the output pressure in special range; when the pressure reaches a certain value, it can be effectively shut off to ensure safe delivery of gas. Currently, the accuracy of gas pressure regulators at home and abroad that adopt the traditional direct- or indirect-acting regulating principle is not high and susceptible to external environmental factors. The accuracy of gas pressure regulator affects the downstream gas supply system directly and reduces the efficiency of energy conversion, thus resulting in energy waste and environmental pollution and even possible supply security issues [1, 2]. In order to solve the problems of mechanical gas regulator, electric gas pressure regulating has become the main direction of the future pressure regulating technology. The focus and difficult

problem of this field is how to improve the precision and stability and shorten the response time [3, 4]. We had an in-depth study in electric gas pressure regulator under the joint support of national innovation fund and Chongqing science and technology research project and achieved a series of research results; this paper discussed the control algorithms of electric gas pressure regulator.

The current electric gas pressure regulator often adopts the conventional PID control mode to take drive control of the core part (micromotor) of electric gas pressure regulator [5, 6]. For piped gas, because of the unpredictable changes of downstream load, the pipe friction, and temperature, the system is time-varying, nonlinear, and it's difficult to build a precise mathematical model. The parameters of conventional PID controller, set by experts, cannot be modified online, which brings much difficulty to achieve the expected effects [7]. The conventional PID controller ameliorating is needed to further improve accuracy and stability and to shorten response time.

Researches show that RBF [8, 9], with fast convergence speed and strong global approximation capability, can approximate any continuous and nonlinear networks with arbitrary precision and shows better performance in the nonlinear system problem-solving [10–12]. Compared with

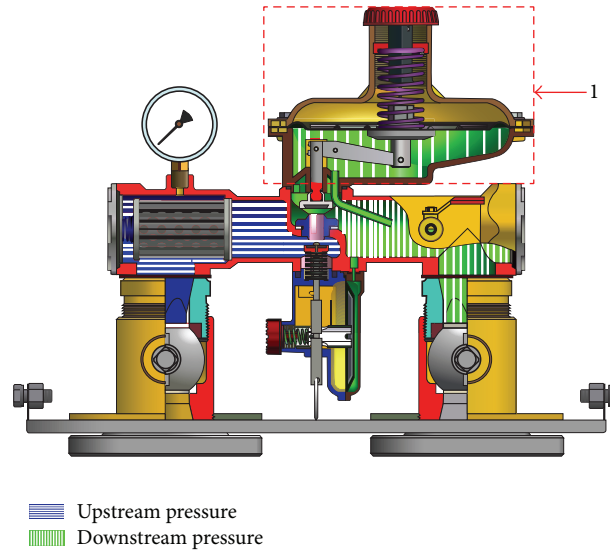


FIGURE 1: Self-actuated regulator.

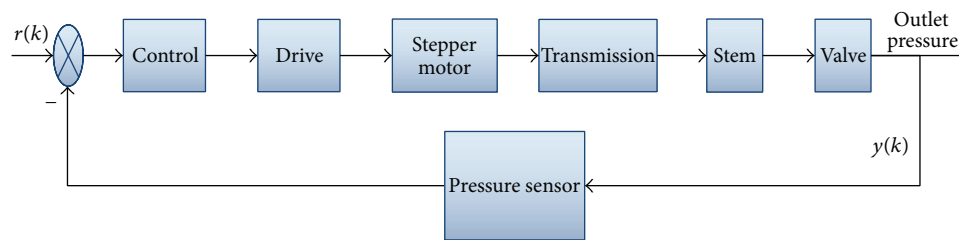


FIGURE 2: The principle diagram of electric gas pressure regulating system.

back propagation (BP) neural network, radial basis function (RBF) neural network has many advantages: faster convergence speed, less training iteration, stronger robustness, and no local minimum, and so forth [13–15]. RBF is extremely sensitive to the base-width vector, the center vector, the radial basis layer, and the selection of the initial value of connection weights of output layer [16–18]. If the parameters of this system are selected inappropriately, the convergence rate of the network will be slow and easy to fall into local minima. To this end, we put forward a kind of improved PID intelligent control algorithm which applies to the electric gas pressure regulator. The algorithm uses the RBF neural network based on PSO [19, 20] to determine the initial parameters of the network [21] and then makes online adjustment on PID parameters to improve the control accuracy, response speed, and tracking performance of the system.

2. Electric Gas Pressure Regulating System

The basic principle of traditional pressure regulating is mechanical balance. The electric gas pressure regulating system range from simple single stage [22] to more complex multistage [23], and the principle of operation is the same in all [24, 25]. The internal structure is shown in Figure 1. When the outlet pressure P_2 is stable, the upward force P_2 acting on the film is balanced against the spring force. If P_2 decreases,

the spring force is greater than the upward force of the film. Then the spring will move down and pull the valve to move upward to increase valve opening and the downstream flow, thereby increasing the gas outlet pressure until the system reaches a new mechanical equilibrium. Obviously, as time goes on, the loss of accuracy of the regulator occurs because of the reduced elasticity of the spring and the friction's damage on diaphragm. Therefore, we replace the original mechanical pressure regulating components with stepping motor and driving system (part 1 of Figure 1) thereof by rebuilding core parts of pressure regulator. The system principle diagram of the improved electric gas pressure regulating is shown in Figure 2. We use the intelligent air pressure sensor array and adopts the information fusion algorithm to detect the electric gas regulator's inlet and outlet pressure [26]. Making differential operation with set pressure value and the result value as the input parameter of controller is to adjust the rotation angle of the stepping motor according to the control algorithm. If the outlet pressure value is less than the set pressure value, the stepping motor will rotate forward, on the contrary, it will reverse. The motor, through the driving system, transforms the angle of motor into linear displacement of output shaft which drives the valve rod to control the valve opening and adjust the outlet pressure. From the working principle, we can draw that the regulator system is a single input/single output closed-loop control system.

For static and dynamic analysis of the system, we need to model the system that mainly includes drive motor model, transmission model, and regulator valve model. According to the pneumatic dynamics, the transfer function of controlled object system of pressure valve [27] can be approximately expressed as

$$G_H(s) = \frac{Y_1 K_1}{Z_4 s^4 + Z_3 s^3 + Z_2 s^2 + Z_1 s + Z_0}. \quad (1)$$

3. PID Controller Based on PSO-RBF Neural Network

3.1. *Parameters Tuning of PID Controller by RBF Neural Network.* Choose the stepper motor as the operator of the electric gas pressure regulator and use digital incremental PID control mode; the control algorithm can be expressed by the following difference equation:

$$\begin{aligned} \Delta u(k) &= u(k) - u(k-1) \\ &= K_p \Delta e(k) + K_i e(k) + K_d (\Delta e(k) - \Delta e(k-1)). \end{aligned} \quad (2)$$

In formula (2), K_p , K_i , and K_d are constant parameters which cannot be adjusted online and expected control effect is impossible to get when the controlled system is nonlinear or is with large delay. Therefore, we adjust PID parameters with RBF neural network online. The performance index of PID parameter is

$$E(k) = \frac{1}{2} [r(k) - y(k)]^2 = \frac{1}{2} e(k)^2. \quad (3)$$

In formula (3), $r(k)$ is the set value of outlet pressure, $y(k)$ is the actual output value of system, and k is the sampling time. In order to minimize the $E(k)$, K_p , K_i , and K_d are sets via the gradient descent method. Consider

$$\begin{aligned} \Delta K_p &= -\eta \frac{\partial E}{\partial k_p} = -\eta \frac{\partial E}{\partial y} \frac{\partial y}{\partial \Delta u} \frac{\partial \Delta u}{\partial k_p} \\ &= \eta_p e(k) \frac{\partial y}{\partial \Delta u} [e(k) - e(k-1)], \end{aligned} \quad (4)$$

$$\begin{aligned} \Delta K_i &= -\eta \frac{\partial E}{\partial k_i} = -\eta \frac{\partial E}{\partial y} \frac{\partial y}{\partial \Delta u} \frac{\partial \Delta u}{\partial k_i} \\ &= \eta_i e(k) \frac{\partial y}{\partial \Delta u} e(k), \end{aligned} \quad (5)$$

$$\begin{aligned} \Delta K_d &= -\eta_d \frac{\partial E}{\partial k_d} = -\eta \frac{\partial E}{\partial y} \frac{\partial y}{\partial \Delta u} \frac{\partial \Delta u}{\partial k_d} \\ &= \eta_d e(k) \frac{\partial y}{\partial \Delta u} \\ &\quad \times [e(k) - 2e(k-1) + e(k-2)]. \end{aligned} \quad (6)$$

To find ΔK_p , ΔK_i , and ΔK_d in (4), (5), and (6), respectively, we must figure out $\partial y / \partial \Delta u$ which can be approximately got by Jacobian matrix of RBF neural network. The radial basis

function is $H = [h_1, h_2, \dots, h_m]$ and it is usually expressed by Gauss function. Consider

$$h_j(X) = \exp\left(-\frac{\|X - C_j\|^2}{2b_j^2}\right), \quad j = 1, 2, \dots, m. \quad (7)$$

In formula (7), $X = \{x_1, x_2, \dots, x_n\}^T$ is the K -dimensional input layer vector; C_j is the central value of j node; and $B = [b_1, b_2, \dots, b_m]^T$ is basis-width vector of the network. $W = [w_1, w_2, \dots, w_m]^T$ is weight vector of the network, so the output layer can be expressed as $y_m = w_1 h_1 + w_2 h_2 + \dots + w_m h_m$. The performance index function of RBF is

$$J = \frac{1}{2} [y(k) - y_m(k)]^2. \quad (8)$$

In formula (8), $y(k)$ is the actual output of the controlled object after K iterations. In order to make J minimum, according to the gradient descent method, the iterative algorithm of output weights, center vector, and basis-width vector parameters is shown as follows:

$$\begin{aligned} \Delta w_j &= [y(k) - y_m(k)] h_j, \\ w_j(k) &= w_j(k-1) + \eta \Delta w_j \\ &\quad + a [w_j(k-1) - w_j(k-2)], \\ \Delta c_{ji} &= [y(k) - y_m(k)] w_j \frac{x_j - c_{ji}}{b_j^2}, \\ c_{ji}(k) &= c_{ji}(k-1) + \eta \Delta c_{ji} \\ &\quad + \alpha [c_{ji}(k-1) - c_{ji}(k-2)], \\ \Delta b_j &= [y(k) - y_m(k)] w_j h_j \frac{\|X - C_j\|}{b_j^3}, \\ b_j(k) &= b_j(k-1) + \eta \Delta b_j \\ &\quad + a [b_j(k-1) - b_j(k-2)]. \end{aligned} \quad (9)$$

In (9), η is the learning rate and α is the momentum factor. Since the online identification ability of neural networks in Jacobian matrix (sensitivity of output versus input change) is powerful, when the neural network identifier can approach the object, we can use $y_m(k)$ instead of the actual output of the controlled object in the system $y(k)$. The Jacobian matrix algorithm of neural network is

$$\frac{\partial y(k)}{\partial \Delta u(k)} = \frac{\partial y_m(k)}{\partial \Delta u(k)} = \sum_{j=1}^m w_j h_j \frac{c_{ji}}{b_j^2}. \quad (10)$$

In formula (10), $x_1 = \Delta u(k)$, $\Delta u(k) = u(k) - u(k-1)$. Taking Jacobian information of the RBF neural network into (4), (5), and (6), three parameter variations (ΔK_p , ΔK_i , ΔK_d) of PID can be obtained. Although RBF neural network can adjust PID parameters online, the selection of the initial parameters

of RBF network will affect the performance of PID control directly, so we use the particle swarm algorithm to optimize RBF neural network to determine initial parameters of the network.

3.2. Realization of Particle Swarm Algorithm to Optimize RBF Neural Network. Suppose there are m particles in a swarm; the information of i th particle is expressed by D vector, so its position and velocity are expressed, respectively, as $X_i = (x_{i1}, x_{i2}, \dots, x_{id})$ and $V_i = \{v_{i1}, v_{i2}, \dots, v_{id}\}$. After each iteration, every particle updates its position and velocity according to individual extremism (pbest) and group extremum (gbest), which are shown as follows:

$$\begin{aligned} v_{id}(k+1) &= \lambda v_{id}(k) \\ &+ c_1 r_1 (p_{id}(k) - x_{id}(k)) + c_2 r_2 (p_{gd}(k) - x_{id}(k)), \end{aligned} \quad (11)$$

$$x_{id}(k+1) = x_{id}(k) + v_{id}(k). \quad (12)$$

In formula (11), c_1 and c_2 are learning factors, r_1 and $r_2 \in (0, 1)$ are random variables, λ is the inertia factor, and k is an iteration number. A large number of experiments show that: with the algorithm iteration performing, λ decreases linearly, improving the convergence of the performance of the algorithm significantly [28]. Consider

$$\lambda = \lambda_{\max} - \frac{k \times (\lambda_{\max} - \lambda_{\min})}{\lambda_{\max}}. \quad (13)$$

In formula (13), λ_{\max} and λ_{\min} are the maximum and minimum values of λ and k_{\max} is the total number of iterations and make $\lambda_{\max} = 0.9$ and $\lambda_{\min} = 0.4$. The PSO algorithm convergence is quick but easy to fall into local optima. This paper makes individual mutate at a certain probability by using genetic algorithm variation theory. The specific steps of center vector, basis-width vector, and output weights of RBF neural network optimized by particle swarm optimization are as follows:

- (1) particle swarm initialization: determine the number of particle swarm, the initial position, speed, individual extremum, and swarm extremum;
- (2) map each individual component to the RBF network parameters and put them into the neural network to calculate the fitness value;
- (3) identify individual extremum and swarm extremum according to fitness value;
- (4) update the particle's velocities and positions according to formulas (11) and (12);
- (5) if the number of iterations reaches a predetermined maximum or meet the performance requirements of minimum error, stop the iteration and output current swarm extremum as initial parameters of the RBF neural network or return to Step (2).

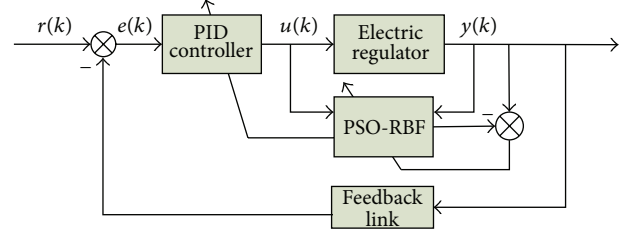


FIGURE 3: The PID control block diagram set by PSO-RBF online.

3.3. PID Controller Design of PSO-RBF Neural Network. RBF neural network-based PID control algorithm can solve the shortage problem of conventional PID control and PSO algorithm can solve the problem of the initial parameters of RBF network. The structure of PID controller of PSO-RBF neural network is shown in Figure 3.

The controller is composed of two parts: the classic PID controller with K_p , K_i , and K_d modified by PSO-RBF online and PSO-RBF network; firstly it uses PSO algorithm to initialize the output weights, the node center, and the basis-width of the RBF neural network and secondly it utilizes the gradient information of RBF neural network online identification to adjust three parameters of PID controller online, control electric regulator valve opening, and adjust the downstream outlet pressure quickly and accurately. The steps of PID intelligent controller algorithm of PSO-RBF neural network are as follows:

- (1) select RBF structure: initialize the input layer node number n , the hidden layer node number m , the learning rate η , and the momentum factor α ;
- (2) determine the output weights, center node, and initial value of node basis-width vector of RBF neural network algorithm by using particle swarm algorithm;
- (3) take the Jacobian information into formulas (4), (5), and (6), obtain PID parameter increments Δk_p , Δk_i , and Δk_d , and correct k_p , k_i , and k_d online;
- (4) make $k = k + 1$ and return to Step (3) until it reaches the cycle index or meets the performance index.

4. Simulation Results and Analysis

The function of electric gas pressure regulator system can refer to formula (1) with the following values: $Z_1 = 5.327 \times 10^4$, $Z_2 = 1.135 \times 10^6$, $Z_3 = 8.193$, $Z_4 = 105.7$, $Y_1 K_1 = 1.328 \times 10^9$, and $Z_0 = 8.688 \times 10^8$; then discretize it considering the sampling period $T = 0.001$ s and get the Z transform:

$$\begin{aligned} G_H(z) &= (5.23 \times 10^{-7} z^3 + 5.75 \times 10^{-6} z^2 \\ &+ 5.75 \times 10^{-6} z + 5.75 \times 10^{-7}) \\ &\times (z^4 - 3.99z^3 + 5.98z^2 - 3.99z + 1.00)^{-1}. \end{aligned} \quad (14)$$

4.1. Step Response. To test the ability of the system response to a step input, RBF network structure is chosen as 3-6-1.

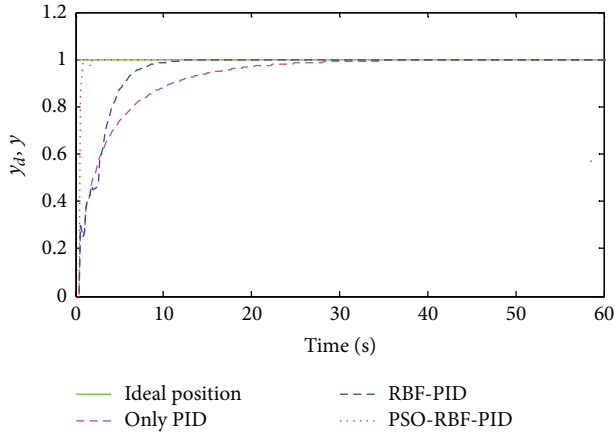


FIGURE 4: The step response of three kinds of algorithms.

Suppose $\eta = 0.50$, $\alpha = 0.05$, $\eta_p = \eta_i = \eta_d = 0.15$, and $k_{p0} = k_{i0} = k_{d0} = 0.1$. Individual information of particle swarm consists of 30 dimension vectors, which are mapped to the RBF network parameters and its evaluation standard is the minimum error of RBF neural network PID controller. The population size of PSO algorithm is 20 and the number of iterations is 100, $c_1 = c_2 = 2$, and individual mutation probability is 0.04. In RBF network, without being optimized by PSO, its initial value is $C_j = 9 \times \text{rands}(3, 6)$; $B = 9 \times \text{rands}(3, 6)$; and $W = 9 \times \text{rands}(6, 1)$. The conventional PID parameters $K_p = K_i = K_d = 0.1$. The step response of three kinds of algorithms in Figure 4 shows that the step response curve time of the system is 2.4 s and is in a stable state under the adaptive control of PID parameter by PSO-RBF. While without using PSO to optimize RBF or adopting conventional PID control, the curve reaches a steady state in 12 s and 32 s, respectively. The conclusion is that PSO-RBF-PID has the shortest response time and the minimum error.

The parameters adaptive tuning curve of PSO-RBF neural network PID controller is shown in Figure 5; it is set finally that $K_p = 0.280$, $K_i = 1.290$, and $K_d = 0.001$.

The minimum error evolution of PSO-RBF neural network is shown in Figure 6. The minimum error changes obviously in the early stage and reaches an optimal value at 63th iteration. RBF network parameters optimized after 100 times are as follows:

$$C_j = \begin{bmatrix} 0.32 & 0.11 & 0.02 & -0.21 & 0.37 & 0.55 \\ -0.36 & -0.13 & 0.30 & -0.03 & -0.30 & -0.06 \\ -0.35 & 0.16 & 0.01 & 0.03 & -0.21 & 0.20 \end{bmatrix};$$

$$B = [0.21 \ 0.11 \ 0.30 \ 0.32 \ 0.36 \ 0.48]^T;$$

$$W = [-0.39 \ 0.16 \ 0.38 \ -0.37 \ -0.57 \ 0.45]^T.$$

(15)

4.2. Tracking Performance. According to the actual needs of the regulator, electric pressure regulator will be formulated according to the input pressure, fast regulate valve opening, and make the outlet pressure stable in the range of errors. So we need to test the tracking performance of the control system. The tracking curves of three kinds of control algorithms

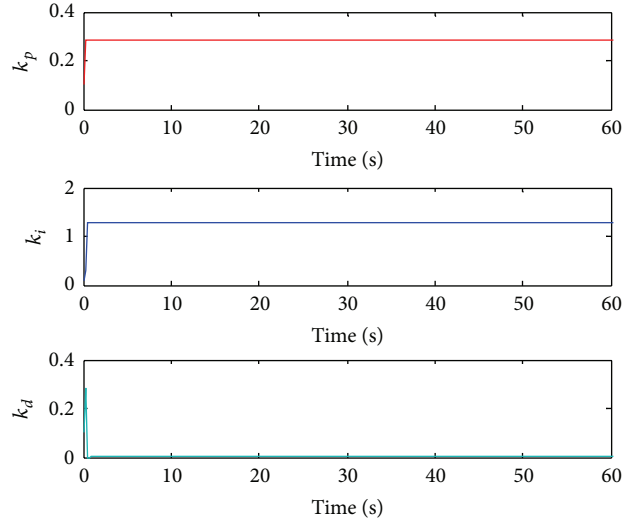


FIGURE 5: The adaptive tuning curve of PSO-RBF neural network PID controller.

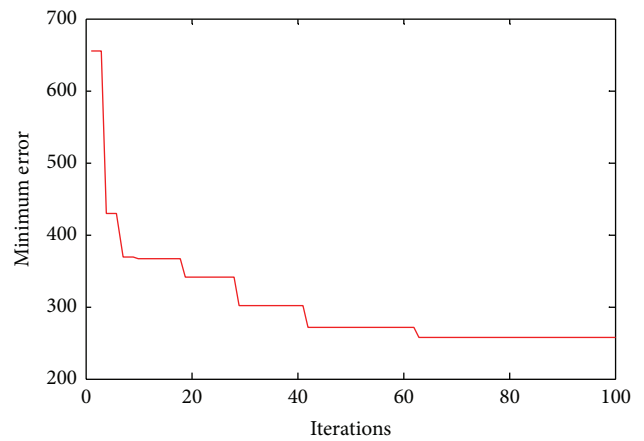


FIGURE 6: The minimum error evolution of PSO-RBF neural network.

are shown in Figure 7. It shows that electric gas pressure regulator which adopts the PSO-RBF-PID has good dynamic performance and higher control precision.

5. Conclusion

An in-depth research has been conducted on the mechanism of electric gas regulator about its nonhigh control precision and stability and a longer response time. It shows that the existing electric gas regulators widely use conventional PID control mode for it is simple and easy to be implemented. However, it is difficult to establish a precise mathematical model given that the pressure regulating system is time-varying and nonlinear. Meanwhile, the conventional PID controller parameters, usually set in line with human experience, cannot be modified online. So, it is hard to achieve the expected control effect to use conventional incremental PID control mode. Therefore, on the basis of the original

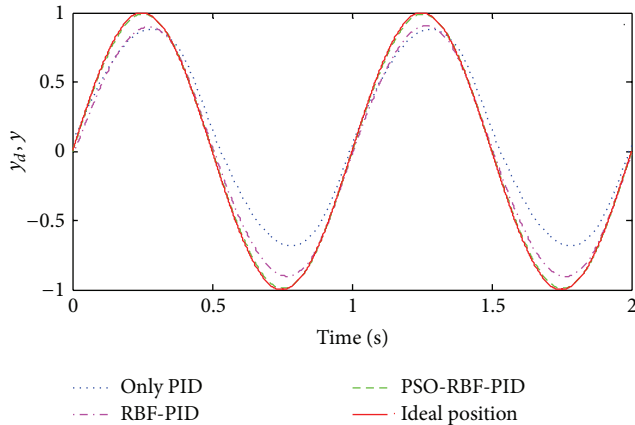


FIGURE 7: The tracking curves of three kinds of control algorithm.

incremental PID control mode, the RBF neural network based on PSO optimization is utilized to determine the initial parameters of the network, and then again set PID parameters online; thus an improved PID intelligent control algorithm suitable for electric gas pressure regulator is established. Theoretical analysis and experimental results demonstrate that the algorithm improves the control accuracy, response speed, and tracking performance of electric gas pressure regulator and can be widely applied to the field of electric gas pressure.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This paper was supported by the National Natural Science Foundation of China (no. 10C26215113031) and the Scientific and Technological Project of Chongqing (no. CSTC2010gg-yyjs40010).

References

- [1] C. Gongjian, S. Fengbin, W. Lili et al., "Analysis on natural gas pressure regulator design principle and its influence factors," *Natural Gas and Oil*, vol. 29, no. 3, pp. 67–71, 2011.
- [2] F. Liang, J. Di, and L. Shuhui, "Study on dynamic model of single-stage self-operated gas regulator," *Gas and Heat*, vol. 29, no. 2, pp. B10–B13, 2009.
- [3] C. Yudong, W. Yong, and L. Yanjun, "Overview of control valve technology," *Control and Instruments in Chemical Industry*, vol. 39, no. 9, pp. 1111–1114, 2012.
- [4] Z. Xiaomu and Z. Mingjun, "Optimum configuration in pressure regulation system at delivery station in gas pipeline," *Natural Gas and Oil*, vol. 23, no. 4, pp. 27–29, 2005.
- [5] Z. Rongrong, Z. Shiwen, and S. Guohua, "The design of an intelligent controller for use with electric valve-actuators," *Industrial Instrumentation Automation*, no. 4, pp. 26–28, 2005.

- [6] L. Zhiguo, L. Jian, and X. Lixin, "Electric pressure regulating valve and self-reliance type pressure regulating valve of combination station," *Oil-Gas Field Surface Engineering*, vol. 31, no. 11, p. 108, 2012.
- [7] L. Wang, W. F. Li, and D. Z. Zheng, "Optimal design of controllers for non-minimum phase systems," *Acta Automatica Sinica*, vol. 29, no. 1, pp. 135–141, 2003.
- [8] Y. D. Song, Q. Cao, X. Du, and H. R. Karimi, "Control strategy based on wavelet transform and neural network for hybrid power system," *Journal of Applied Mathematics*, vol. 2013, Article ID 375840, 8 pages, 2013.
- [9] A. Zhang, C. Chen, and H. R. Karimi, "A new adaptive LSSVR with online multikernel RBF tuning to evaluate analog circuit performance," *Abstract and Applied Analysis*, vol. 2013, Article ID 231735, 7 pages, 2013.
- [10] X. Dong, Y. Zhao, H. R. Karimi, and P. Shi, "Adaptive variable structure fuzzy neural identification and control for a class of MIMO nonlinear system," *Journal of the Franklin Institute*, vol. 350, no. 5, pp. 1221–1247, 2013.
- [11] C. C. Chuang, J. T. Jeng, and P. T. Lin, "Annealing robust radial basis function networks for function approximation with outliers," *Neurocomputing*, vol. 56, no. 1–4, pp. 123–139, 2004.
- [12] J. K. Sing, D. K. Basu, M. Nasipuri, and M. Kundu, "Face recognition using point symmetry distance-based RBF network," *Applied Soft Computing Journal*, vol. 7, no. 1, pp. 58–70, 2007.
- [13] H. Liu, S. Wang, and P. Ouyang, "Fault diagnosis in a hydraulic position servo system using RBF neural network," *Chinese Journal of Aeronautics*, vol. 19, no. 4, pp. 346–353, 2006.
- [14] Z. A. Bashir and M. E. El-Hawary, "Applying wavelets to short-term load forecasting using PSO-based neural networks," *IEEE Transactions on Power Systems*, vol. 24, no. 1, pp. 20–27, 2009.
- [15] C. Zhang, M. Lin, and M. Tang, "BP neural network optimized with PSO algorithm for daily load forecasting," in *Proceedings of the International Conference on Information Management, Innovation Management and Industrial Engineering (ICIII '08)*, vol. 56, pp. 82–85, December 2008.
- [16] K. Meng, Z. Y. Dong, D. H. Wang, and K. P. Wong, "A self-adaptive RBF neural network classifier for transformer fault analysis," *IEEE Transactions on Power Systems*, vol. 25, no. 3, pp. 1350–1360, 2010.
- [17] G. Li, Q. Zhang, and Y. Liang, "GA-based PID neural network control for magnetic bearing systems," *Chinese Journal of Mechanical Engineering*, vol. 20, no. 2, pp. 56–59, 2007.
- [18] W. Lianghong, W. Yaonan, Z. Shaowu, and T. Wen, "Design of pid controller with incomplete derivation based on differential evolution algorithm," *Journal of Systems Engineering and Electronics*, vol. 19, no. 3, pp. 578–583, 2008.
- [19] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of IEEE International Conference on Neural Networks*, pp. 1942–1948, December 1995.
- [20] M. Barari, H. R. Karimi, and F. Razaghian, "Analog circuit design optimization based on evolutionary algorithms," *Mathematical Problems in Engineering*, vol. 2014, Article ID 593684, 12 pages, 2014.
- [21] M. Sheikhan, R. Shahnazi, and S. Garoucy, "Synchronization of general chaotic systems using neural controllers with application to secure communication," *Neural Computing and Applications*, vol. 22, no. 2, pp. 361–373, 2013.
- [22] R. Mooney, "Pilot-loaded regulators: what you need to know," *Gas Industries*, vol. 34, no. 1, pp. 31–33, 1989.
- [23] A. Krigman, *Guide to Selecting Pressure Regulators*, InTech, 1984.

- [24] A. E. Rundell, S. V. Drakunov, and R. A. DeCarlo, "A sliding mode observer and controller for stabilization of rotational motion of a vertical shaft magnetic bearing," *IEEE Transactions on Control Systems Technology*, vol. 4, no. 5, pp. 598–608, 1996.
- [25] N. Zafer and G. R. Luecke, "Stability of gas pressure regulators," *Applied Mathematical Modelling*, vol. 32, no. 1, pp. 61–82, 2008.
- [26] Y. Zhong, F. Wang, L. Zhang, and C. Tao, "The information fusion algorithm on gaspressure regulator inlet/outlet detection," *Journal of Computational Information Systems*, vol. 10, no. 5, pp. 2145–2153, 2014.
- [27] W. Zhi, *Study on Dynamic Characteristic and Control of Electric Control Valve*, Shan Dong University, 2013.
- [28] Y. H. Shi and R. C. Eberhart, "Empirical study of particle swarm optimization," in *Proceedings of the Congress on Evolutionary Computation*, pp. 1945–1950, Piscataway, NJ, USA, 1999.