

Research Article

System Identification of Heat-Transfer Process of Frequency Induction Furnace for Melting Copper Based on Particle Swarm Algorithm

Zhi-gang Jia and Xing-xuan Wang

Department of Electronic Engineering, Fudan University, Shanghai 200433, China

Correspondence should be addressed to Zhi-gang Jia; 523562419@qq.com

Received 28 October 2014; Revised 17 February 2015; Accepted 17 February 2015

Academic Editor: Hung-Yuan Chung

Copyright © 2015 Z.-g. Jia and X.-x. Wang. 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.

An adaptive evolutionary strategy in standard particle swarm optimization is introduced. Adaptive evolution particle swarm optimization is constructed to improve the capacity of global search. A method based on adaptive evolution particle swarm optimization for identification of continuous system with time delay is proposed. The basic idea is that the identification of continuous system with time delay is converted to an optimization of continuous nonlinear function. The adaptive evolution particle swarm optimization is utilized to find an optimal solution of continuous nonlinear function. Convergence conditions are given by the convergence analysis based on discrete time linear dynamic system theory. Numerical simulation results show that the proposed method is effective for a general continuous system with time delay and the system of heat-transfer process of frequency induction furnace for melting copper.

1. Introduction

Frequency induction furnace has been developed into a kind of smelting equipment widely used. It is composed of four main parts: furnace body, electrical system, water cooling system, and hydraulic system [1]. Its power supply is industrial frequency (50 or 60 Hz) current. The research object in this paper is a vertical frequency induction furnace in a copper company. After adding raw material, workers regulate transformer output-voltage so that material melts. Copper smelting and heat preservation need lower transformer output-voltage. The molten copper is kept in proper temperature. Finally pour the molten copper [2].

The heat-transfer process in frequency induction furnace system consists of copper melting process and heat-transfer process of molten copper to furnace wall. The approximation of most industrial processes is low-order continuous system. For the sake of convenience, the approximations of those two processes are first-order continuous systems, respectively [2]. The following mathematical model can be set up by frequency

induction furnace input voltage and measured data of furnace wall temperature

$$G(s) = \frac{K}{(T_0s + 1)(T_1s + 1)}, \quad (1)$$

where K , T_0 , and T_1 denote heat-transfer gain, time constant of the process of copper melting, and time constant of heat-transfer process of molten copper to furnace wall [2]. But this industrial process has time delay. Therefore time delay is introduced to model (1) in literature [1]. The mathematical model of frequency induction furnace heat-transfer process is proposed as follows:

$$G(s) = \frac{K}{(T_0s + 1)(T_1s + 1)} e^{-Ls}. \quad (2)$$

The controller design of frequency induction furnace heat-transfer process needs identification of system (2).

Identification methods of second-order system with time delay in literature [3–6] set up identification model by integral

computation of step response and measured data of input signal in time domain. Then identify parameters through Newton-Raphson method and least square method. Methods in literature [3–6] can only be used for second-order system with time delay.

The continuous system identification methods based on particle swarm optimization (PSO) proposed in this paper are effective for general continuous system with time delay. The input signal has no limitation.

2. Adaptive Evolution PSO (AEPSo)

2.1. Standard PSO. The particle swarm optimization algorithm is a new evolutionary computation method proposed by Kennedy and Eberhart in 1995. It is a simulation of the natural phenomenon of a flock of birds searching for food. As a key optimization technique, PSO has been used extensively in many fields, including function optimization, neural network, and fuzzy system control [7]. PSO algorithm belongs to the evolutionary algorithm similar to genetic algorithm. It starts from the random solutions and searches for the optimal solution through iteration [8].

In D dimensional solution space, the position of i th particle can be expressed as a D dimensional vector

$$X_i = [x_{i1} \ x_{i2} \ x_{i3} \ \cdots \ x_{iD}]. \quad (3)$$

The velocity vector of i th particle is expressed as follows:

$$V_i = [v_{i1} \ v_{i2} \ v_{i3} \ \cdots \ v_{iD}]. \quad (4)$$

The position of the optimal solution of i th particle can be expressed as a D dimensional vector as follows:

$$P_i = [p_{i1} \ p_{i2} \ p_{i3} \ \cdots \ p_{iD}]. \quad (5)$$

The position of the optimal solution of the swarm which has the maximum fitness value in the swarm can be expressed as follows:

$$P_g = [p_{g1} \ p_{g2} \ p_{g3} \ \cdots \ p_{gD}]. \quad (6)$$

The dynamic range of the particles is as follows:

$$X_{\min} = [x_{\min 1} \ x_{\min 2} \ x_{\min 3} \ \cdots \ x_{\min D}] \quad (7)$$

$$X_{\max} = [x_{\max 1} \ x_{\max 2} \ x_{\max 3} \ \cdots \ x_{\max D}].$$

The dynamic range of the velocity of the particles is as follows:

$$V_{\max} = [v_{\max 1} \ v_{\max 2} \ v_{\max 3} \ \cdots \ v_{\max D}]. \quad (8)$$

f is a real function which is defined in D dimensional space. The fitness value of i th particle can be expressed as follows:

$$\text{fitness}_i = f(X_i). \quad (9)$$

Particles' velocity and position are updated from generation t to $t + 1$ according to the following equations:

$$V_i^{t+1} = \omega * V_i^t + c_1 * \text{rand}_1 * (P_i^t - X_i^t) \quad (10)$$

$$+ c_2 * \text{rand}_2 * (P_g^t - X_i^t)$$

$$X_i^{t+1} = X_i^t + V_i^{t+1}, \quad (11)$$

where rand_1 and rand_2 are random number from 0 to 1. c_1 and c_2 are local learning factor and global learning factor, respectively. c_2 is larger than c_1 generally. ω is inertia weight.

Studies show that the standard PSO algorithm can find the local optimal solution quickly, but it also can produce premature convergence phenomenon. Each particle adjusts its search direction based on individual optimal position and global optimal position in standard PSO algorithm. But particles are more and more close to the global optimal particle along with iteration. Thus particles' velocity decreases and tends to zero gradually. So this particle swarm loses ability to evolve further. The standard PSO is considered to be convergent at this time. In many cases, the convergence results do not converge to the global optimal solution, not even converge to a local optimal solution sometimes [9].

In order to avoid premature convergence phenomenon, many improved PSO algorithms have been proposed. These algorithms tend to focus on improving inertia weight ω or introduce special mutation operation to particles [10]. This paper will introduce adaptive inertia weight $\omega(t)$ and special mutation operation to the swarm to avoid premature convergence phenomenon.

2.2. The Introduction of Adaptive Inertia Weight. PSO algorithm which has introduced adaptive inertia weight needs to do the following modifications. Equation (10) can be modified as follows:

$$V_i^{t+1} = \omega(t) * V_i^t + c_1 * \text{rand}_1 * (P_i^t - X_i^t) \quad (12)$$

$$+ c_2 * \text{rand}_2 * (P_g^t - X_i^t).$$

Time-varying inertia weight $\omega(t)$ has been used in (12). If a linear decreasing inertia weight $\omega(t)$ is used in iterative process, the PSO algorithm has good global search capability in the beginning and has a good local search ability in the later stage [11]. A good global search capability makes the PSO algorithm locate the area around the global optimal solution quickly in the beginning. The good local search ability makes the PSO algorithm obtain the global optimal solution accurately in the later stage. The adaptive inertia weight can be determined by the following equation in iteration:

$$\omega(t) = \omega_{\max} - \frac{(\omega_{\max} - \omega_{\min})}{G} * t, \quad (13)$$

where G is the number of evolution generation. ω_{\max} and ω_{\min} are constants. Let adaptive inertia weight $\omega(t)$ vary in the range of $[\omega_{\min}, \omega_{\max}]$. ω_{\max} and ω_{\min} are usually 0.9 and 0.1, respectively, according to experience.

2.3. Mutation Operation. Mutation operation is a natural biological model of gene mutation induced by various accidental factors in genetic environment. It takes very small probability to change the position of the particle randomly. In order to obtain high quality optimal solution as far as possible, mutation operation must be taken. Mutation operation plays an auxiliary role in the genetic algorithm but is a necessary operation step of genetic algorithm. In this paper, the basic

mutation strategy of genetic algorithm is introduced to the standard PSO algorithm [12]. The particles which will do mutation operation are selected according to the probability. Then the positions of the selected particles do mutation operation. The position density distribution model of each particle is considered to be uniform distribution density function in this paper [13]

$$p(X_{ij}) = \frac{1}{X_{\max j} - X_{\min j}}, \quad (14)$$

where X_{ij} denotes j th parameter of i th mutation particle. Then a new particle can have the following representation:

$$X_i = X_{\min} + (X_{\max} - X_{\min}) * r, \quad (15)$$

where r is a random variable which satisfies uniform distribution in the range of $[0, 1]$.

AEPSO consists of the following steps.

- (1) Initialization: generate N particles in D dimensional solution space which have random position and velocity vectors.
- (2) Fitness evaluation: calculate the fitness value of i th particle and find out the optimal particle in the swarm which has the maximum fitness value, $i = 1, 2, 3, \dots, N$.
- (3) Updating of particles' position and velocity: update the position and velocity of each particle with (12) and (11), respectively. Check whether the position and velocity of each particle are out of bounds.
- (4) Mutation operation: the positions of particles do mutation operation with (15).
- (5) Comparison of current fitness value of i th particle $f(X_i)$ and individual optimal fitness value of i th particle $f(P_i)$: if $f(X_i) > f(P_i)$, then $P_i = X_i$.
- (6) Comparison of current fitness value of i th particle $f(X_i)$ and global optimal fitness value of the swarm $f(P_g)$: if $f(X_i) > f(P_g)$, then $P_g = X_i$.
- (7) Checking criterion for stopping iteration of algorithm: if the criterion is satisfied, the iteration of the algorithm is stopped. Otherwise, go to Step 3 and continue searching for the optimal solution until the criterion is satisfied.

3. System Identification of General Continuous System with Time Delay

Consider the general continuous system as follows:

$$g(s) = \frac{b_m s^m + b_{m-1} s^{m-1} + \dots + b_1 s + b_0}{a_n s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0} e^{-Ls}, \quad (16)$$

where $m \leq n$. The time delay of such a general continuous system needs approximate processing according to the identification method of second-order continuous system with

time delay in literature [3]. The time delay can be represented by Taylor series as follows:

$$e^{-Ls} = 1 - Ls + \frac{(Ls)^2}{2} + \dots \quad (17)$$

Now the parameter identification of L becomes very difficult. After approximate processing, L has high math power. The form of the general continuous system will become complicated after multiplication. Parameters of second-order continuous system with time delay can be identified well by Newton-Raphson method. But it does not work well for high-order system. The identification of the general continuous system can be converted to optimization of nonlinear function. Then the AEPSO algorithm mentioned in this paper can be used to solve this optimization problem. The parameters which need to be identified can be expressed as the following vector:

$$\theta = [b_m \ b_{m-1} \ \dots \ b_1 \ a_n \ a_{n-1} \ \dots \ a_1 \ L]. \quad (18)$$

The nonlinear function which needs to be optimized can be expressed as follows:

$$J(\hat{\theta}) = (Y - \hat{Y})^T (Y - \hat{Y}), \quad (19)$$

where Y is the measured output vector of real system. \hat{Y} is calculated output vector of estimation model. The fitness function f can be expressed as follows:

$$f = \frac{1}{J(\hat{\theta})}. \quad (20)$$

Therefore the parameter identification problem of general continuous system can be regarded as a standard optimization problem, minimization of (19) according to the input and output sequences. Search for the parameter vector which can minimize (19) in $n + m + 1$ dimensional space.

4. Analysis of Convergence

Analyze one element of the parameter vector without loss of generality:

$$\begin{aligned} V_{ij}^{t+1} &= \omega * V_{ij}^t + c_1 * \text{rand}_1 * (P_{ij}^t - X_{ij}^t) \\ &\quad + c_2 * \text{rand}_2 * (P_{gj}^t - X_{ij}^t) \\ X_{ij}^{t+1} &= X_{ij}^t + V_{ij}^{t+1}. \end{aligned} \quad (21)$$

Equations (21) can be simplified as follows:

$$\begin{aligned} v(t+1) &= \omega v(t) + c_1 * \text{rand}_1 * (p_i - x(t)) \\ &\quad + c_2 * \text{rand}_2 * (p_g - x(t)) \\ x(t+1) &= x(t) + v(t+1). \end{aligned} \quad (22)$$

Equations (22) can be converted into the following discrete dynamic system:

$$\begin{aligned}
v(t+1) &= \omega v(t) - (c_1 * \text{rand}_1 + c_2 * \text{rand}_2) \\
&\quad \cdot \left(x(t) - \frac{c_1 * \text{rand}_1 * p_i + c_2 * \text{rand}_2 * p_g}{c_1 * \text{rand}_1 + c_2 * \text{rand}_2} \right) \\
x(t+1) &= \frac{c_1 * \text{rand}_1 * p_i + c_2 * \text{rand}_2 * p_g}{c_1 * \text{rand}_1 + c_2 * \text{rand}_2} \\
&= \omega v(t) + (1 - (c_1 * \text{rand}_1 + c_2 * \text{rand}_2)) \\
&\quad \cdot \left(x(t) - \frac{c_1 * \text{rand}_1 * p_i + c_2 * \text{rand}_2 * p_g}{c_1 * \text{rand}_1 + c_2 * \text{rand}_2} \right). \tag{23}
\end{aligned}$$

The individual optimal position of a particle and the global optimal position of the swarm are usually unchangeable. The above two equations can be simplified as follows:

$$\begin{aligned}
v(t+1) &= \omega v(t) - cz(t) \\
z(t+1) &= \omega v(t) + (1-c)z(t). \tag{24}
\end{aligned}$$

So the above system can be rewritten as a discrete autonomous system:

$$Y(t+1) = AY(t), \tag{25}$$

where $Y(t) = [v(t) \ z(t)]^T$, $A = \begin{bmatrix} \omega & -c \\ \omega & 1-c \end{bmatrix}$.

The equilibrium in Lyapunov sense can be obtained by the following equations:

$$(I - A)Y_e = 0. \tag{26}$$

If $\det(I - A) = c \neq 0$, there is only one equilibrium. The characteristic equation of autonomous system can be expressed as follows:

$$\lambda^2 - (1 - c + \omega)\lambda + \omega = 0. \tag{27}$$

Characteristic root is the two following equations:

$$\begin{aligned}
\lambda_1 &= \frac{\omega + 1 - c + \sqrt{(\omega + 1 - c)^2 - 4\omega}}{2} \\
\lambda_2 &= \frac{\omega + 1 - c - \sqrt{(\omega + 1 - c)^2 - 4\omega}}{2}. \tag{28}
\end{aligned}$$

Theorem 1. *The sufficient and necessary condition of autonomous system (25) converging to the equilibrium state is as follows:*

$$\max\{|\lambda_1|, |\lambda_2|\} < 1. \tag{29}$$

Therefore the sufficient and necessary condition can be obtained as follows:

$$\begin{aligned}
|\omega| &< 1 \\
0 &< c < 4. \tag{30}
\end{aligned}$$

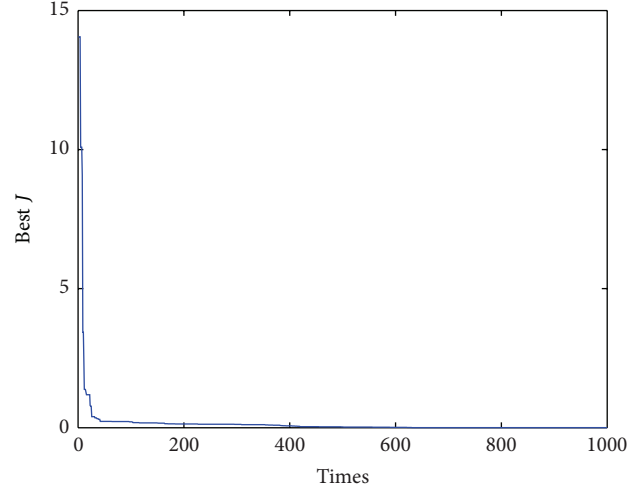


FIGURE 1: The results of nonlinear function needing to be optimized versus Times.

5. Numerical Simulation

5.1. Experiment One. Consider the following third-order continuous system with time delay. The input signal is random signal sequence with amplitude of 4:

$$g(s) = \frac{4s^2 + 6s}{4s^3 + 15s^2 + 17.5s + 1} e^{-0.8s}. \tag{31}$$

There are six parameters needed to be identified. They can be expressed as the following vector:

$$\theta = [b_2 \ b_1 \ a_3 \ a_2 \ a_1 \ L]. \tag{32}$$

The settings of identification method based on AEPSo algorithm are as follows. The identification window s is set to 1000. The number of particles $N = 200$. The number of iteration generations $Times = 1000$. The local learning factor $c_1 = 1.3$. The global learning factor $c_2 = 1.7$. The inertia weight ω decreases from 0.9 to 0.1 with the iteration. The search range of parameters mentioned above is $[0, 20]$.

The results of the nonlinear function needing to be optimized are shown in Figure 1.

The error δ of parameters estimation can be calculated by the following equation:

$$\delta = \frac{\|\hat{\theta} - \theta\|}{\|\theta\|} \times 100\%. \tag{33}$$

The error δ of parameters estimation is shown in Figure 2.

The parameters estimation is shown in Table 1. From Figures 1-2, we can see that the results of the nonlinear optimized function and the error of parameters estimation δ are becoming smaller as $Times$ increases. From Table 1, the parameters estimation is becoming closer to the true values with iteration. The identification method based on AEPSo algorithm is effective for continuous system with time delay according to the results of experiment one.

Now we will use MATLAB system identification toolbox to identify that system, because the MATLAB identification

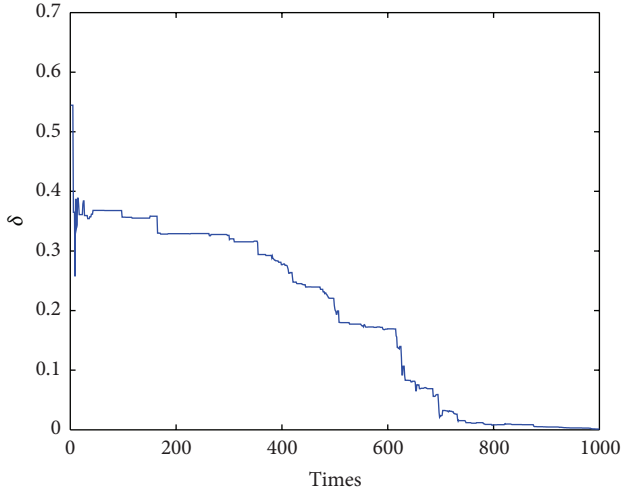


FIGURE 2: The error δ of parameters estimation versus *Times*.

TABLE 1: Estimation of parameters.

Times	b_2	b_1	a_3	a_2	a_1	L	δ (%)
1	4.3001	3.6775	4.7041	15.8040	4.4133	0.8062	54.46
100	3.9306	3.0649	3.8284	12.0910	9.8059	0.8043	35.67
200	3.6863	3.2186	3.6900	11.7114	10.7010	0.8015	32.91
300	3.6091	3.3414	3.6158	11.6022	10.7973	0.8000	32.62
400	2.9136	4.1961	2.9262	10.7083	12.8079	0.8003	27.70
500	2.5226	5.4856	2.5382	10.9166	15.6515	0.7986	20.28
600	2.6645	5.7444	2.6757	11.5040	16.3083	0.7994	16.95
700	3.7783	6.0790	3.7766	14.5121	17.5467	0.7996	2.40
800	3.9280	6.0538	3.9294	14.8759	17.6169	0.8000	0.84
900	3.9867	6.0446	3.9872	15.0075	17.6055	0.8000	0.47
1000	3.9997	6.0106	4.0000	15.0090	17.5290	0.8000	0.14
True value	4	6	4	15	17.5	0.8	

toolbox identifies process system by fitting the transfer function of process system. But due to the limitation of MATLAB identification toolbox, it can identify the process system model as follows:

$$\frac{K(1 + Tz * s)}{(1 + (2\zeta * Tw)s + (Tw)^2 s^2)(1 + Tp * s)} e^{-Td * s}. \quad (34)$$

The process system transfer function obtained by MATLAB system identification toolbox can be shown as follows:

$$\frac{5.78 \times 10^4 s + 0.005793}{2.949s^3 + 9.687s^2 + 16.66s + 1} e^{-0.00182s}. \quad (35)$$

It can be obviously seen that the transfer function obtained by MATLAB system identification toolbox is completely wrong. The parameters which we need cannot be identified efficiently by the identification toolbox. So we can draw the conclusion that identification algorithm mentioned in this paper is more effective than traditional identification toolbox on identification of nonlinear system.

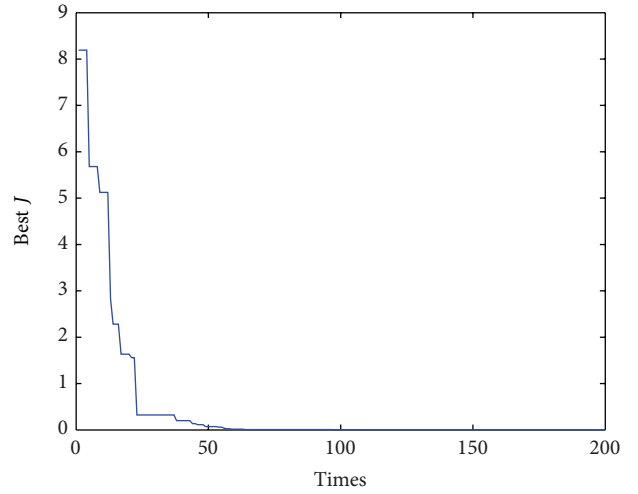


FIGURE 3: The results of nonlinear function needing to be optimized versus *Times*.

5.2. *Experiment Two.* Identify a second-order system which is similar to the heat-transfer system of frequency induction furnace. The second-order continuous system with time delay is as follows. The input signal is unit step signal

$$G(s) = \frac{6}{(6s + 1)(20s + 1)} e^{-4s}. \quad (36)$$

There are four parameters needed to be identified. They can be expressed as the following vector:

$$\theta = [K \ T_1 \ T_2 \ L]. \quad (37)$$

The settings of identification method based on AEPSO algorithm are as follows. The identification window s is set to 1000. The number of particles $N = 100$. The number of iteration generations $Times = 200$. The local learning factor $c_1 = 1.3$. The global learning factor $c_2 = 1.7$. The inertia weight ω decreases from 0.9 to 0.1 with the iteration. The search range of parameters mentioned above is $[0, 30]$.

The results of the nonlinear optimized function and the error δ of parameters estimation are shown in Figures 3 and 4, respectively.

The parameters estimation of the second-order system with iteration is shown in Table 2.

From Table 2, we can see that the step response of continuous system can be used to identify the system. The proposed identification method in this paper has no restrictions on input signal.

Now we will identify that system by MATLAB system identification toolbox. Fit the following transfer function model:

$$\frac{K}{(1 + Tp_1 * s)(1 + Tp_2 * s)} e^{-Td * s}. \quad (38)$$

The transfer function obtained by MATLAB system identification toolbox can be shown as follows:

$$\frac{5.9902}{(1 + 19.9651s)(1 + 5.9793s)} e^{-4.9955s}. \quad (39)$$

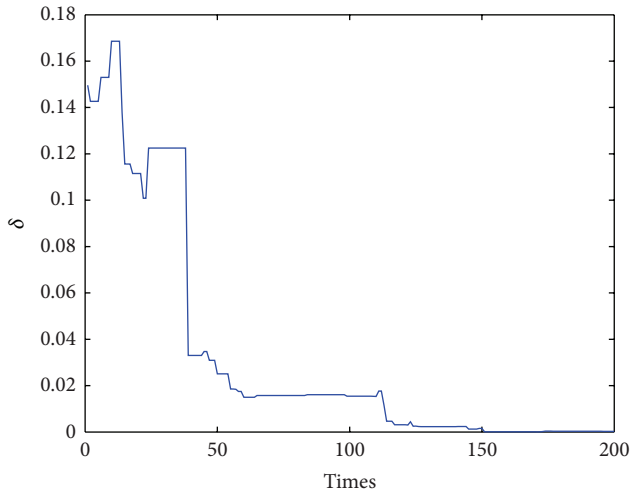


FIGURE 4: The error δ of parameters estimation versus Times.

TABLE 2: Estimation of parameters.

Times	K	T_1	T_2	L	δ (%)
1	6.0794	5.1238	17.8664	6.3655	14.96
20	5.9288	6.1143	17.8270	5.1525	11.15
40	6.0121	6.5824	20.1386	3.5830	3.30
60	6.0044	6.2609	19.8547	3.8584	1.50
80	5.9985	6.2603	19.8246	3.8509	1.57
100	5.9962	6.2567	19.8341	3.8470	1.55
120	6.0000	6.0510	19.9795	3.9592	0.31
140	6.0000	6.0381	19.9858	3.9694	0.23
160	5.9998	5.9995	19.9996	3.9979	0.01
180	5.9999	6.0057	19.9969	3.9968	0.03
200	6.0000	6.0040	19.9973	3.9978	0.02
True value	6	6	20	4	

The error δ of parameters estimated by the identification method mentioned by this paper is 0.02%. The error δ_1 of parameters obtained by MATLAB identification toolbox is 4.51%. Because of $\delta < \delta_1$, the method mentioned in this paper is obviously superior to the system identification toolbox. In order to observe this kind of superiority more directly, the step response of the system transfer function identified by the mentioned method in this paper and MATLAB identification toolbox can be painted in Figure 5.

The step responses of transfer function obtained by different methods are painted in Figure 5. The blue circles represent step response of system identified by MATLAB identification toolbox. The green solid line represents step response of system identified by method mentioned in this paper. The red rectangles represent step response of the original system. The system transfer function obtained by system identification toolbox in MATLAB is very accurate and bestfits has reached 95.22%. But by comparison of the error of parameter estimated and step response in Figure 5, we can draw the conclusion that the identification method

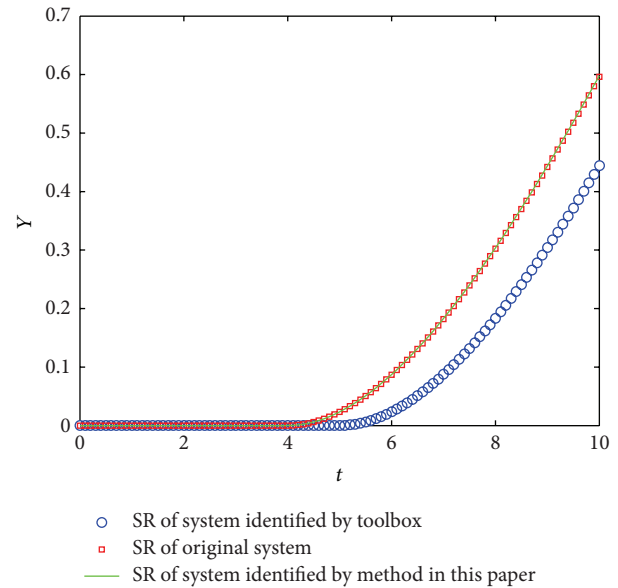


FIGURE 5: The step response of the system transfer function identified.

mentioned in this paper is more effective than traditional methods.

6. Conclusion

The paper discusses how to identify the heat-transfer process system of frequency induction furnace. An identification method of general continuous system with time delay based on AEPSO algorithm has been proposed. Experiment one demonstrates the proposed identification method's effectiveness for general continuous system. Experiment two shows that the heat-transfer process system of frequency induction system can be identified by the proposed method. The proposed method has advantage over the method in literature [3–6]. There are no special restrictions on the input signal. Through the comparison experiment with traditional MATLAB identification toolbox, we can see that the proposed method is more effective.

Conflict of Interests

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

References

- [1] X. Meisheng, J. Zhou, D. Sun, and Y. Liu, "Simulation research of temperature control system for frequency induction furnace," *Control and Instruments in Chemical Industry*, vol. 40, no. 8, 2013 (Chinese).
- [2] Z. Wang, K. Cao, and J. He, *Induction Furnace*, Chemical Industry Press, Beijing, China, 2007.
- [3] Y. B. Quan, W. D. Zhang, and X. M. Xu, "Step response identification method for 2-order with time-delay system," *Control Theory & Applications*, vol. 19, no. 6, pp. 954–956, 2002 (Chinese).

- [4] X. Wang, H. Yue, and D. Gao, "Direct identification of continuous second order plus dead-time model," *Acta Automatica Sinica*, vol. 27, no. 5, pp. 728–731, 2001 (Chinese).
- [5] W. Weihe and P. Wang, "Identification method and application of second order plus time delay model," *Control and Instruments in Chemical Industry*, vol. 37, no. 9, 2010 (Chinese).
- [6] Q. Meng and H. Ge, "System identification of second order plus time delay system based on unit impulse response," *Machinery*, vol. 50, no. 2, 2012.
- [7] Y. Dai, X. Zhao, and L. Liu, "Nonlinear system identification based on evolution particle swarm optimization," *Computer Simulation*, vol. 27, article 10, 2010.
- [8] M. Chen, C. Zhang, and C. Luo, "Adaptive evolutionary particle swarm algorithm for multi-objective optimization," *Journal of System Simulation*, vol. 21, no. 22, 2009.
- [9] Q.-B. Jin, Z. Liang, and L. Quan, "Identification of continuous-time systems with unsteady initial conditions," *Control Theory and Applications*, vol. 28, no. 1, pp. 125–130, 2011.
- [10] F. Wang, K. Xing, and X. Xu, "A system identification method using particle swarm optimization," *Journal of Xi'an Jiaotong University*, vol. 43, no. 2, pp. 116–120, 2009.
- [11] W. Lin, H. Zhang, S. Liu, and J. Qian, "The hammerstein model identification based on PSO," *Chinese Journal of Scientific Instrument*, vol. 27, no. 1, pp. 75–79, 2006.
- [12] X. Xu, B. Sun, and J. Fu, "Identification method of structure system based on particle swarm optimization," *Bulletin of Science, Technology & Society*, vol. 23, no. 6, 2007.
- [13] W. An and W. Li, "An improved particle swarm optimization algorithm and its application to engineering," *Mechanical Science and Technology*, vol. 24, no. 4, 2005.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

