International Journal of Control and Automation
Vol.8, No.5 (2015), pp.199-208
http://dx.doi.org/10.14257/ijca.2015.8.5.19

Optimization Control of ATO-S Based on Implicit Generalized Predictive of Chaotic Particle Swarm Algorithm

Lu Xiaojuan, Ma Baofeng* and Dong Haiying

School of automation and electrical engineering, Lanzhou Jiaotong University Lanzhou, Gansu, 730070, China
*mabf_lzjtu@163.com**

Abstract

With the continuous development of the train control system, some requirements like safety, smoothness, punctuality and accurate parking in controlling of the automatic train operation system has improved. An implicit generalized predictive control method based on chaos particle swarm optimization algorithm, which can advance the speed and the accuracy in optimum searching of the controller is proposed, therefore to achieve an real-time like control of the automatic train operation system. Simulation and analysis are carried out in the constraint of input amplitude of the automatic train operation system, the results show that the tracking effect is enhanced, the control input is smooth and stable, in comparison with that typical implicit generalized predictive controller used.

Keywords: *Automatic train operation; Implicit generalized predictive control; Chaotic particle swarm optimization algorithm; simulation*

1. Introduction

The automatic train operation system (ATO-S) is a nonlinear system with characteristics like complex process, large time-delay, difficult to modeling and etc.. When working in different operating conditions, environment and route, it needs the ATO-Ss to realize auto acceleration and deceleration and ensures the stability, safety, energy saving, punctuality and accurate parking of the train as well as the traveling comfort.

In China, most ATO-S still use the PID controller [1, 2] for its algorithm is simple in principle and implement is easy. The offset parameter of the formulas, however, should be given at first, which cannot achieve an ideal control effect together with its lacking of flexibility. Latterly, the fuzzy neural network (FNN) algorithm is applied in ATO-S by some scholars [3-5], but the network convergence speed of such FNN algorithm is slow and the global optimum is difficult to obtain, generally. The implicit generalized predictive control (IGPC) algorithm is a development of computer science algorithm on the basis of generalized predictive control (GPC) in recent years [6], which has advances like strong self-adaption, self-correction, robustness and not rely on accurate model completely, and it can be applied to control systems with time-delay and nonlinear. Therefore, it is suitable to apply IGPC algorithm to the ATO-S on the assumption that the system is linear and unconstrained [7-9] by using the gradient search method [10] to obtain the optimal predictive control increment. In actual process, however, a system with no restriction is almost non-existent. So, it needs to find an optimization of IGPC algorithm that can overcome the constraint.

The chaotic particle swarm optimization (CPSO) algorithm not only has the advantages of particle swarm optimization (PSO) algorithm, but also has the ability to overcome the defects of PSO which is easy to fall into local optimum, thus, to achieving a global optimization. This paper designs an implicit generalized predictive controller based on CPSO hybrid optimization and applies to a local ATO-S for trail, which makes the train

ISSN: 2005-4297 IJCA

more energy-saving, stable and more accurate in operation by follow the target operation curve, therefore, fulfills a high-efficiency control.

2. Implicit Generalized Predictive Control Algorithms

2.1. Generalized Predictive Control Algorithm

2.1.1 CARIMA Prediction Model: In GPC algorithm, an controlled auto regressive integrated moving average (CARIMA) model is used to describe the controlled process with random interference [11].

$$A\left(z^{-1}\right)y\left(k\right)=B\left(z^{-1}\right)u\left(k-1\right)+C\left(z^{-1}\right)\xi\left(k\right) \tag{1}$$

where, $A(z^{-1})$, $B(z^{-1})$ and $C(z^{-1})$ are polynomials containing a backward shift operator z^{-1}, $C(z^{-1})=1$, generally. $\Delta=1-z^{-1}$ is the difference operator, $y(k)$, $u(k)$ and $\xi(k)$ represent for white noise sequences of the output, the input and the mean value of the defined model.

According to the forecast theory, in order to get an output with j steps ahead, it necessary to solve the CARIMA model and the Diophantine equation [12] to get an optimal predictive output:

$$\hat{\boldsymbol{Y}}=\boldsymbol{G}\Delta\boldsymbol{U}+\boldsymbol{f} \tag{2}$$

where,

$$\hat{\boldsymbol{Y}}=\left[\hat{y}\left(k+1\right),\hat{y}\left(k+2\right),...,\hat{y}\left(k+n\right)\right]^{\mathrm{T}},$$

$$\Delta\boldsymbol{U}=\left[\Delta u\left(k\right),\Delta u\left(k+1\right),...,\Delta u\left(k+n-1\right)\right]^{\mathrm{T}},$$

$$\boldsymbol{f}=\boldsymbol{H}\Delta\boldsymbol{U}\left(k\right)+\boldsymbol{F}y\left(k\right)=\left[f\left(k+1\right),...,f\left(k+n\right)\right]^{\mathrm{T}},$$

$$\boldsymbol{G}=\begin{bmatrix} g_0 & & & \\ g_1 & g_0 & & \\ \vdots & \vdots & \ddots & \\ g_{n-1} & g_{n-2} & \cdots & g_0 \end{bmatrix},\boldsymbol{F}=\begin{bmatrix} F_1\left(z^{-1}\right) \\ F_2\left(z^{-1}\right) \\ \vdots \\ F_n\left(z^{-1}\right) \end{bmatrix}$$

$$\boldsymbol{H}=\begin{bmatrix} G_1-g_0 \\ z\left(G_2-z^{-1}g_1-g_0\right) \\ \vdots \\ z^{n-1}\left(G_n-z^{-n+1}g_{n-1}-...-z^{-1}g_1-g_0\right) \end{bmatrix}.$$

Note also that n is the prediction length, m is the time control step with $m\leq n$.

2.1.2 Rolling Optimization: In GPC algorithm, the optimization performance index must contain the output error and the weighting terms to minimum the value of the objective function.

$$\min J\left(k\right)=\sum_{j=1}^{n}\left[y\left(k+j\right)-w(k+j)\right]^2+\sum_{j=1}^{m}\lambda\left(j\right)\left[\Delta u\left(k+j-1\right)\right]^2\left(j=1,2,...,n\right) \tag{3}$$

where,

$$w(k+j)=\alpha^{j}y\left(k\right)+\left(1-\alpha^{j}\right)y_{r}\left(k\right)\quad (0\leq\alpha<1)$$

and $\lambda(j)$ is the control weighing coefficient. While y_r is the set value and $\Delta u(k)$ is the input increment. Therefore, an optimum solution can be obtained, with the control input increment unconstrained to minimize the performance index as is shown in (3), via gradient optimization method:

$$\Delta \boldsymbol{U}=\left(\boldsymbol{G}^{\mathrm{T}}\boldsymbol{G}+\lambda\boldsymbol{I}\right)^{-1}\boldsymbol{G}^{\mathrm{T}}\left(\boldsymbol{W}-\boldsymbol{f}\right) \tag{4}$$

and the optimal control variable is

$$u\left(k\right)=u\left(k-1\right)+\Delta u \tag{5}$$

where, Δu is the first row vector of $\Delta \boldsymbol{U}$.

2.2. Implicit Generalized Predictive Self-correcting Control Algorithm

In this paper, the IGCP algorithm is used in the ATO-S rather than GPC algorithm with without solving the Diophantine equation and identifying the model parameter, and identifies G and f directly according to the predictive equation with the input and output data used. In equation (4), the control parameter weighting factor λ and the diffusion set vector W is given, thus, we can get n controllers in parallel form equation (2):

$$\begin{cases} y(k+1)=g_0\Delta u(k)+f(k+1)+E_1\xi(k+1) \\ y(k+2)=g_1\Delta u(k)+g_0\Delta u(k+1)+f(k+2)+E_2\xi(k+2) \\ \quad\vdots \qquad\qquad\qquad\qquad \vdots \\ y(k+n)=g_{n-1}\Delta u(k)+\cdots+g_0\Delta u(k+n-1)+f(k+n)+E_n\xi(k+n) \end{cases} \tag{6}$$

Additionally, both control strategy and parameter estimation are adopted with a combination, and replaced the auxiliary input estimated value ŷ=k/(k-n) with output estimation y=k/(k-n), combines with the nth equation in (6), we can get:

$$y(k)=\hat{X}(k-n)\theta(k)+\xi\left(k\right) \tag{7}$$

where,

$$\hat{X}=[\Delta u(k),\Delta u(k+1),\cdots,\Delta u(k+n-1),1],$$
$$\theta(k)=[g_{n-1},g_{n-2},\cdots,g_0,f(k+n)]^{\mathrm{T}}$$

where $\theta(k)$ can be estimated by using the recursive least square method [13]. Then, the elements g_0, $g_1,\ldots,$ g_{n-1} and $f(k+n)$ in matrix $\boldsymbol{G}$ can be figured out once $\theta(k)$ and its estimation is obtained.

$$f=\begin{bmatrix} f(k+1) \\ f(k+2) \\ \vdots \\ f(k+n) \end{bmatrix}=\begin{bmatrix} \hat{y}(k+2/k) \\ \hat{y}(k+3/k) \\ \vdots \\ \hat{y}(k+n+1/k) \end{bmatrix}+\begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}e(k+1) \tag{8}$$

where, $e(k+1)=y(k+1)-\hat{y}(k+1/k)$ is estimation error, $\hat{y}(k+n/k)=\hat{X}(K)\hat{\theta}(K)$ is the n step estimation value at time k. Hence, we can use formula (4) to calculate the control parameter after $\boldsymbol{G}$ and $\boldsymbol{f}$ obtained. In every step, m control sequence form the current step to n steps thereafter can be determined.

3. Summary of CPSO algorithm

PSO algorithm [14-15] is a stochastic optimization method which obtains the optimal solution by iteration with a group of random particle used for initializing. In each iteration, the particles update themselves by tracing two extreme values, one is the optimum solution, another is the best found solution that the extreme value searched in the whole population. The particles update themselves according to the formula shown blew.

$$v(t+1)=wv(t)+c_1r_1\left(pbest(t)-x(t)\right)+c_2r_2\left(gbest(t)-x(t)\right) \quad (9)$$

$$x(t+1)=x(t)+v(t+1) \quad (10)$$

where, both c_1 and c_2 represent for the acceleration factor, r_1 and r_2 are random number varies from [0,1], w is the inertia weighting, $x(t)$ and $v(t)$ stand for the location and the velocity of the particle at time t, respectively. $pbest(t)$ is the extreme value of a single particle, $gbest(t)$ is the globe extreme value of the whole population at time t.

As we can known from the optimum search properties, the PSO algorithm is easy to fall into the local optimal despite its calculation is easy and parameters used is in a small number. The CPSO algorithm, in comparison, can jump out of local optimal solution easily owe to its chaotic randomness and ergodic property [16,17]. Therefore, CPSO is a commendable search mechanism as it has the advantages of PSO and overcomes the shortcomings of PSO algorithm, as well. The optimization searching and extreme solution updating in this paper is realized via the typical chaotic Logistic map [18], thus to optimize the control input increment $\Delta\boldsymbol{U}$ of the train.

4. Hybrid Optimization of IGPC based on CPSO

It can make up a multimode hybrid optimization by inducing CPSO algorithm to the rolling optimization in IGPC to coordinate with the gradient optimization search in unconstrained conditions. When the controlled object is unconstrained, optimal input parameters can be obtained by using gradient optimization search. In another case, namely, the control object is constrained, the gradient optimization search coordinates with the CPSO and then obtains the optimal control input quickly and accurately. The control structure of CPSO-IGPC is shown in Figure 1.

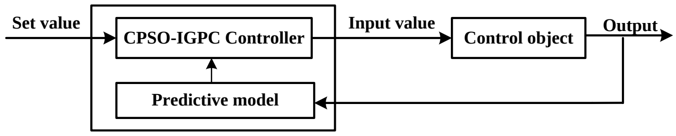

Figure 1. Control Structure of CPSO-IGPC

4.1. Optimization Strategy of IGPC based on CPSO

In the process of hybrid optimization in this paper, the particles are separated into two groups, one is the PSO group (this paper calls this group **P**), and the other is the CPSO group (this paper calls this group **C**). If $Ppbest(t)$ and $Pgbest(t)$ stand for the historical extreme value at time t and the global extreme value, $Cgbest(t)$ represents the global extreme value of group **C** a time t, $gbest(t)$ stands for the global historical extreme value at time t. Then, the optimal sequence ΔU can be obtained by making the target function minimized in the initialization process of CPSO under the unconstraint condition. After that, ΔU is assigned to the θ% seeds as the initial value while the remain seeds are assigned with random initial values, which increases the speed of the algorithm and the diversity of the population, meanwhile, makes the population contains part of high quality seed.

In order to make the input increment ΔU smaller and improve the robustness of the system, during optimization control process, this paper adopts (4) as the adjusting function of CPSO algorithm with the optimization target value set as zero. The CPSO optimization search method introduces two parameters, one is the maximum stable continuous iteration b and the other is stable accuracy δ, to identify whether an optimum value is obtained or not, in the actual situation, once one of the two parameters achieved, the optimization process stops. This way saves the optimization time and also makes it real time. The optimization search procedures can be seen in Figure 2 as is shown in below.

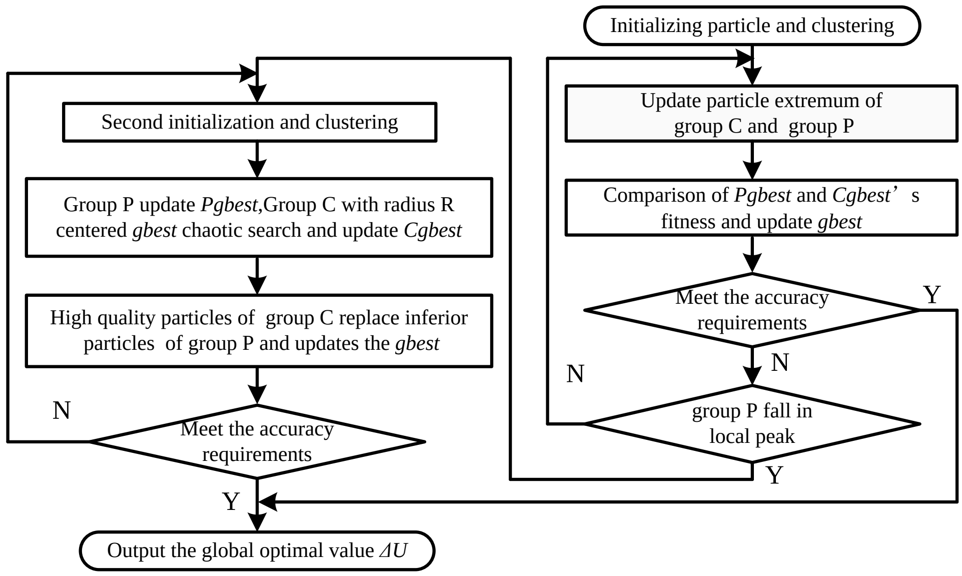

Figure 2. Optimization Search Procedures of CPSO-IGPC Algorithm

4.2. Speed Controller Design based on CPSO-IGPC

The speed controller designed according to CPSO-IGPC algorithm and characteristics of the train is shown in Figure 3. In the speed controller, the target curve is used as an input of the controller after a soften regulation, which makes it easy for limiting the overshoot in the follow of the target curve. The actual operation speed is fed back to the input side to compare with the set value, thus a control parameter is determined for correction the prediction of the coming time. So, the speed controller designed like this is of strong robustness and practical sense for the systems which requires a imprecise actual and prediction model, especially to the ATO-S, for which model can not be precisely set.

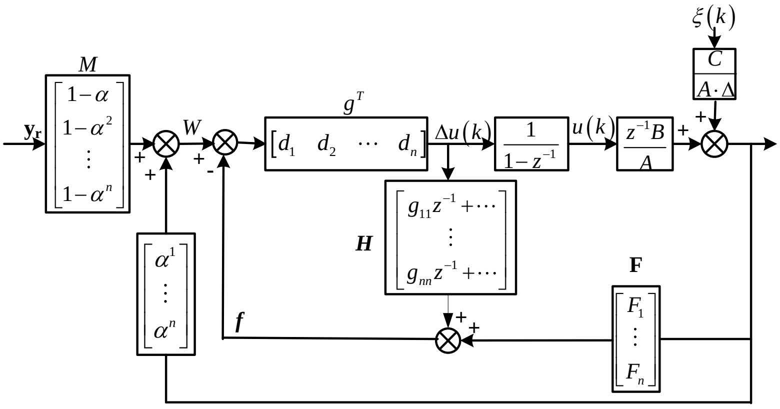

Figure 3. Speed Controller Design based on CPSO-IGPC Algorithm

5. Simulation Results Analysis

5.1. Determination of the Train Model

In this paper we use a generation bullet train CRH2-300 operated on the Beijing-Shanghai high speed railway as the object for simulation study, the parameters of that is shown in Table 1.

Table 1. Parameters of Bullet Train CRH2-300

Labels	Values
Total train weight (t)	890
Maximum operating speed (km/h)	380
Continuous operation speed (km/h)	350
Unit basic resistance (N/kN)	w_0=0.53+0.0039v+0.000114v^2

This paper uses the actual output and the model which best matches the prediction model as the sub-model of the controlled object to predict the control input [19], and the multi-model system obtained is shown in (11).

$$\begin{cases} y(k)-0.9998y(k-1)=0.0318u(k-1) \\ y(k)-0.9995y(k-1)=0.0318u(k-1) \\ y(k)-0.9989y(k-1)=0.0318u(k-1) \\ y(k)-0.9984y(k-1)=0.0318u(k-1) \\ y(k)-0.9979y(k-1)=0.0318u(k-1) \\ y(k)-0.9976y(k-1)=0.0318u(k-1) \end{cases} \tag{11}$$

where, $y(k)$ stands for the train speed at time t and $u(k)$ is the unit traction.

5.2. Target Curve Generation of the Train

The target curve of the train is an optimal curve that the controller must follow in order to achieve the optimum operation. The simulation taken in this paper is under a high-speed circumstance with the train speed set to 350km/h. As is known, an energy-saving target can be fulfilled in two situations, one is in the inert operation and the other is in the uniform operation, so, in this paper, we adopts the energy-saving operation strategy in generation the curve of the train, that is, traction-uniform operation- inert operation-uniform operation- traction- uniform operation- inert operation-braking. The simulated target curve is shown in Figure 8, where the maximum operation speed with interval speed limits should be lower than the limited speed of automatic protection system for trains.

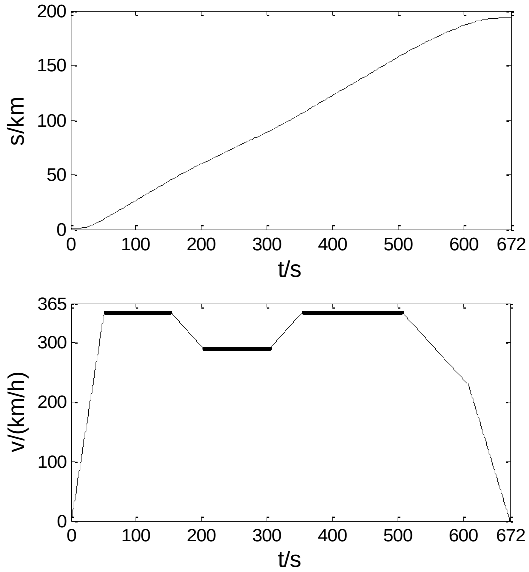

Figure 4. Target Operation Curve of the Train

5.3. Simulation Test of the Designed Speed Controller

In this paper, we use MATLAB 7.0 as the simulation environment, and uses the controller based on IGPC and CPSO-IGPC algorithm, separately, to follow the target curve with the generated target curve as input. Besides, we set $\Delta \boldsymbol{U} \in [-8,8]$, n=8, m=2, λ=0.95, α=0.9, c_1=2, c_2=2 and w=0.6, b=100 and θ=10.

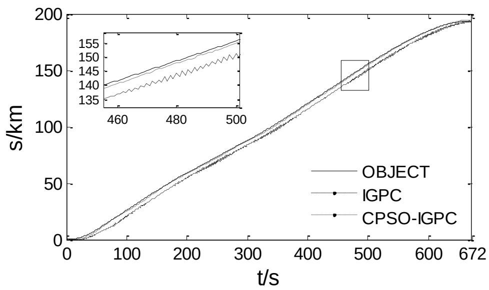

Figure 5. S-T Curves of Different Control Algorithm

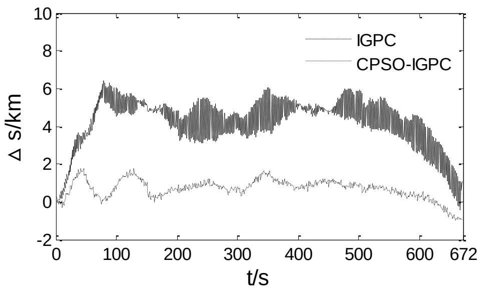

Figure 6. Error Analysis of S-T Curves

From Figure 5 we can see that the controller based on IGPC is of a better follow and adaptive capability after optimizing with CPSO, and is clearly shown in Figure 6, where the control error of CPSO-IGPC is few than that with only IGPC used, furthermore, the oscillation amplitude is smaller and the robustness is stronger.

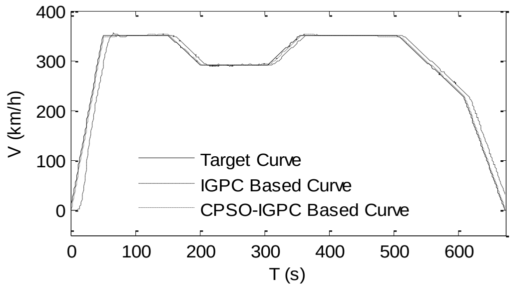

Figure 7. V-T curves of Different Control Strategy

Compared the target curve with the speed follow curves based on IGPC and CPSO-IGPC in Figure 7, it is easy to find that the follow of CPSO-IGPC is smoother and the overshooting is smaller than that IGPC used under different operation conditions, hence more energy is saved.

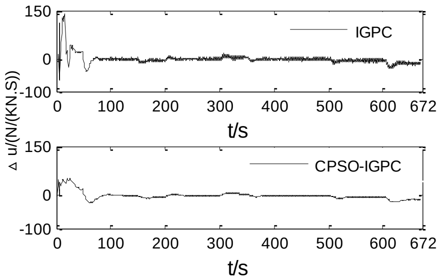

Figure 8. ΔU-T Fluctuation Analysis Curves

Figure 7 shows the Δ*u*-*t* curve with CPSO-IGPC algorithm and IGPC algorithm used, the unit is N/(KN.S), it can be seen that the control increment is higher in the staring period of the train, while the remaining time is smooth. Comparatively speaking, the increment and amplitude with CPSO-IGPC is smaller that IGPC simply used. So, the designed speed controller can fulfill the requirements for the comfort of passengers.

6. Conclusions

Once the train speed reaches to a certain extent, the ATO-S must be used for the train control. This paper studied the ATO-S development as well as the advantages and the disadvantages of both PSO and CPSO algorithm, firstly. A speed controller based on CPSO-IGPC algorithm is designed for ATO-S and simulations about the automatic train control process with constrains is conducted, simulation results show that the speed controller based on CPSO-IGPC is more superior compared with the one based on IGPC in optimization speed, accuracy, energy-saving and etc.. An applying of the CPSO-IGPC to ATO-S, therefore, can achieve a better control effect.

Acknowledgement

This research was financially supported by the Gansu Provincial Natural Science Foundation of China granted by 1208RJZA180.

References

[1] T. Tang and L. J. Huang, "A survey of control algorithm for automatic train operation", Journal of the China Railway Society, vol. 25, no. 2, (**2003**).
[2] S. Y. Wang, Y. Shi and Z. X. Feng, "A Method for Controlling a Loading System Based on a Fuzzy PID Controller", Mechanical Science and Technology for Aerospace Engineering, vol. 30, no. 1, (**2011**).
[3] J. Yu, Q. Q. Qian and Z. Y. He, "Research on Application of Two-degree Fuzzy Neural Net work in ATO of High Speed Train", Journal of the China Railway Society, vol. 30, no. 5, (**2008**).
[4] L. Yao, "A Novel Parameter Optimization Algorithm for Mamadani Fuzzy Neural Networks Based on PSO", Journal of Guangdong University of Technology, vol. 3, no. 1, (**2014**).
[5] Y. J. Xue, Y. M. Hu, S. G. Liu J. F. Yang Q. C. Chang and S. T. Bao, "Improving Land Resource Evaluation Using Fuzzy Neural Network Ensembles", Pedosphere, vol. 17, no. 4, (**2007**).
[6] F. Yang and C. H. Yuan, "Implicit Generalized Predictive Control for Electrical Heating Box with Multi-loop Control", Proceedings of the 32nd Chinese Control Conference, Xi'an, China, (**2013**) July 26-28.
[7] W. Wang, Generalized Predictive Control Theory and its Application, Scientific Publishers, Beijing (**1998**).
[8] Y. Y. Jin, "New generalized predictive control with input constraints", Control and Decision, vol. 17, no. 4, (**2002**).
[9] J. L. Guzman, M. Berenguel and S. Dormido, "Interactive teaching of constrained generalized predictive control", Control Systems Magazine, vol. 25, no. 2, (**2005**).
[10] Y. H. Hu and X. L. Jia, "Summarization of generalized predictive control", Information and control, vol. 29, no. 3, (**2000**).
[11] Q. A. Li and P. Li, "Constrained multivariable adaptive generalized predictive control for diagonal CARIMA model", Control and Decision, vol. 24, no. 3, (**2009**).
[12] R. Z. Tong, "Solving a Diophantine equation", Journal of Shenyang Normal University, vol. 23, no. 2, (**2012**).
[13] L. Q. Zou, "The least square method and its simple application", Science and Technology Information, no. 23, (**2010**).
[14] T. A. Huang, J. G. Sheng and H. Y. Xu, "Improved Simplified Particle Swarm Optimization", Computer Simulation, vol. 30, no. 2 (**2013**).
[15] Y. H. Shi and R. C. Eberhart, "A Modified Particle Swarm Optimizer", IEEE International Conference of Evolutionary Computation, Alaska, America, (**1998**) May 4-9.
[16] S. Gao and J. Y. Yang, "Research on Chaos Particle Swarm Optimization Algorithm", Pattern Recognition and artificial Intelligence, vol. 19, no. 2, (**2006**).
[17] J. C. Wang, G. L. Shan, D. P. Liu and Z. N. Zhao, "Improved mutative scale chaos particle swarm optimization and its application", Computer Engineering and Design, vol. 33, no. 5, (**2012**)
[18] R. Q. Chen and J. S. Yu, "Study and Application of Chaos-Particke Swarm Optimization-based Hybred Optimization Algorithm", Journal of System Simulation, vol. 20, no. 3, **(2008).**
[19] S. G. Zhang, Study on the design method of high speed train, China Railway Publishing House, Beijing (**2009**).