DCDIS Series B, Vol. 14 (S2), Copyright © 2007 Watam Press
Complex Systems and Applications—Modeling, Control and Simulations, (2007) 603-607

Parameter Optimization of PID Controller Based on Quantum-behaved Particle Swarm Optimization Algorithm

Maolong Xi[1], Jun Sun[2] and Wenbo Xu[3]

Abstract —**The conventional parameter optimisation of PID controller is easy to produce surge and big overshoot, and therefore heuristics such as genetic algorithm (GA), particle swarm optimisation (PSO) are employed to enhance the capability of traditional techniques. But the major problem of these algorithms is that they may be trapped in the local optima of the objective and lead to poor performance. In this paper, a quantum-behaved particle swarm optimisation (QPSO) for the parameter optimisation of PID controller is proposed from sub-optimal perspective. This method is very advantageous for practical control systems. Three examples are given to illustrate the design procedure and exhibit the effectiveness of the proposed method via a comparison study with an existing Z-N, GA and PSO approaches**

Index Terms—**QPSO, PID controller, parameter optimisation**

Ⅰ INTRODUCTION

Research of modern control theory is prompted recently and many methods have been proposed for design of controllers. Though the conventional proportional-integral-derivative (PID) controller is still widely used in process industries because its simplicity and robustness, thus it is very important which methods to choose for tuning the parameter of PID controller. Among the existing tuning methods, the Ziegler–Nichols formula [1] may be the most well known technique. For a wide range of practical processes, this tuning approach works quite well. However, sometimes it dose not provide good tuning and is easy to produce surge and big overshoot, particularly for processes with serious non-linearity. Therefore, this method usually needs retuning before being used to control industrial process. To enhance the capabilities of traditional PID parameter tuning techniques, several intelligent approaches have been suggested to improve PID tuning, such as the neural networks [2~3], the fuzzy methods [4~5], the genetic algorithms (GA)[6~7], the particle swarm optimisation (PSO)[8~9]. As intelligent algorithms, genetic algorithm and particle swarm optimization have great superiority in tuning the parameters of PID controllers.

[1]Maolong Xi is with the Centre of Intelligent and High Performance Computing, School of Information Technology Southern Yangtze University, Wuxi, China, 214122. He is with Wuxi Institute of Technology, Wuxi, China, 214121(e-mail:wx_xml@hotmail.com)

[2]Jun Sun is with the Centre of Intelligent and High Performance Computing, School of Information Technology Southern Yangtze University, Wuxi, China, 214122.

[3]Wenbo Xu is with the Centre of Intelligent and High Performance Computing, School of Information Technology Southern Yangtze University, Wuxi, China, 214122

But the major problem of PSO is that they are not global searching algorithm and may be trap in the local optima of the objective [10].

In this paper, quantum-behaved particle optimisation which having global searching ability is introduced to tuning the parameters of PID controller. The rest part of the paper is organized as follows. In Section 2, the problem formulation of PID controller is given. In Section 3, we introduce the QPSO and give the implement progress of the algorithm. The experiment results of three different plants of PID controller with different algorithms are given in Section 4. Finally, the paper is concluded in Section 5.

Ⅱ PROBLEM FORMULATION

PID controller consists of Proportional Action, Integral Action and Derivative Action. The control-loop system is illustrated in Fig. 1 where r, e, y are respectively the reference, error and controlled variables.

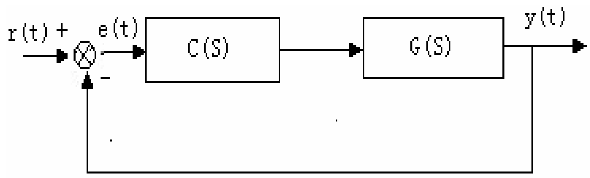

Fig.1 The PID control system diagram

G(s) is the plant transfer function and C(s) is the controller transfer function. The *Laplace* transform of the C(s) is

$$C(S) = Kp + Ki * \frac{1}{S} + Kd * S \qquad (1)$$

Where Kp , Ki , Kd are respectively the proportional, integral, derivative parameters of the PID controllers.

Formally, a "performance index" is defined as a quantitative measure to depict the system performance. Using this technique an "optimum system" can often be designed and a set of parameters in the system can be adjusted to meet the required specification optimally .To a PID control system, there are often four indices to depict the system performance: ISE, IAE, ITAE and ITSE. They are defined as follows:

$$ISE = \int_0^T e(t)^2 dt$$

$$IAE = \int_0^T |e(t)| dt$$

$$ITAE = \int_0^T t|e(t)| dt$$

$$ITSE = \int_0^T te(t)^2 dt$$

In this paper, in order to obtain the good overshoot and short settling time (Ts), settling time is adopted as the main performance criterion and ITAE index is adopted as the assistant performance criterion. Composing them together, the objective function of PID controller parameter problem is depict as:

$$objectivevalue = \alpha T_s + (1-\alpha)\int_0^T t|e(t)| dt \quad (2)$$

With the assistant performance criterion ITAE, we choose the weight factor α to be in [0.6,1], with the different value of α, they represent different importance to $objectivevalue$

Ⅲ QUANTUN-BEHAVED PARTICLE SWARM PTIMIZATION

Quantum-behaved particle swarm optimisation is stochastic optimisation algorithm that was originally motivated by the thinking model of an individual of the social organism. In [11], Jun Sun et al consider a social organism is a system far more complex than that formulated by particle swarm optimisation (PSO)[12], and a linear evolvement equation is not sufficient to depict it at all. In practice, the evolution of man's thinking is uncertain to a great extent somewhat like a particle having quantum behaviour and they introduce quantum theory into PSO and propose a Quantum-behaved PSO algorithm. The experiment results indicate that the QPSO works better than standard PSO on several benchmark functions in [13~14]. In practice applications of designing IIR digital filters [15] and training RBF neural networks [16], the algorithm has shown better performance and it is a promising algorithm.

In QPSO, at every iteration, each particle records its *pbest* (best information of individual) and compares its *pbest* with those of all other particles in its neighbourhood, and population gets the *gbest*. (best information of the population). Then its Learning Inclination Point p can be given by Equation (3) after random numbers φ_1 and φ_2 which are generated between 0 and 1.

$$p = (\varphi_1 * pbest + \varphi_2 * gbest)/(\varphi_1 + \varphi_2) \quad (3)$$

A mainstream thought point is employed to evaluate the creativity of a particle and the point or Mean Best Position (*mbest*) is defined as the centre-of –gravity *gbest* position of the particle swarm which is formulated as equation (4)

$$mbest = \sum_{i=1}^{M} pbest_i / M = (\sum_{i=1}^{M} pbest_{i1}/M, \sum_{i=1}^{M} pbest_{i2}/M, \ldots \sum_{i=1}^{M} pbest_{id}/M) \quad (4)$$

Where M is the population size.

Therefore, QPSO has iterative equation of following form:

$$x(t+1) = p \pm \beta * |mbest - x(t)| * \ln(1/u) \quad (5)$$

Where β is called Creativity Coefficient, u is a random number between 0 and 1. ± is decided by a random number between 0 and 1 in every iteration, when the number is bigger than 0.5, **-** is used otherwise + is used.

The process for implementing the global version of QPSO is as follows:

The QPSO algorithm is described as follows.

```
----------------------------------------------------------
Initialize population: random xi
do
find out mbest using equation (4)
for i=1 to population size M
  If f (xi)<f(Pi) then pi=xi
       pg=min (Pi)
  for d=1 to dimension D
       fi1=rand (0,1), fi2=rand (0,1)
       P=(fi1*pid+fi2*pgd)/(fi1+fi2)
       L=beta*abs(mbestd-xid)
       u=rand (0,1)
       if rand (0,1)>0.5
             xid=P-L*ln(1/u)
       else
             xid=P+L*ln(1/u)
Until termination criterion is met
----------------------------------------------------
```

As a result, QPSO has simple form and only one parameter β works on individual particle's convergence speed; therefore it is very suitable to treat our optimisation problem.

Ⅳ EXPERIMENT RESULTS

In this section, three examples with different plants are given to illustrate the proposed design procedures and a comparison study with existing algorithm (Z-N, GA, PSO) is carried out to illustrate the effectiveness of the proposed approach.

In order to utilize the results of Z-N algorithm and reducing the searching space, considering the value of Z-N algorithm as the centre and extending them, then we can determine the searching space of problems for the three algorithms by the following equalities

$$(1-\lambda)Kp' \le Kp \le (1+\lambda)Kp' \quad (6)$$

$$(1-\lambda)Ki' \le Ki \le (1+\lambda)Ki' \quad (7)$$

$$(1-\lambda)Kd' \le Kd \le (1+\lambda)Kd' \quad (8)$$

Where $Kp^{'}$, $Ki^{'}$, $Kd^{'}$ are the tuning value with Z-N algorithm and Kp , Ki , Kd are parameters of PID controller.

In the experiments, the fitness value is set as formula (2) value where α is 0.95. We set the population size 40, max Generation 100 and λ in searching space 0.9. In order to investigate the scalability of the algorithms, we had 10 trial runs for every instance and recorded mean values of best fitness, settling time and overshoot.

Here, β in QPSO is declined linearly from 1 to 0. 5

Table1 The Best Parameters and Performance Index of 10 Trial Runs for Four Algorithms

Algorithm	The best parameters of 10 trial runs								
	G1 (S)			G2 (S)			G3 (S)		
	Kp	Ki	Kd	Kp	Ki	Kd	Kp	Ki	Kd
Z-N	18	12.8114	6.3223	12.4533	2.2811	0.1096	2.813	1.719	1.151
GA	32.3450	2.6240	12.0703	17.4218	2.6976	0.1739	2.1203	1.2091	2.0697
PSO	12.5950	0.6406	12.3285	16.7134	2.6484	0.1657	2.0522	1.1355	1.3629
QPSO	12.6681	0.6406	12.3285	18.0820	2.4901	0.2137	2.2256	1.2759	1.6080
	Ts	Ov.	Obj	Ts	Ov.	Obj.	Ts	Ov.	Obj
Z-N	6.2969	0.6229	6.1968	0.9107	0.3187	0.9379	2.5432	0.3215	2.3935
GA	1.7156	0.3646	1.6101	0.7758	0.3847	0.8010	0.9864	0.0695	1.0012
PSO	0.6846	0.0483	0.7536	0.7722	0.3814	0.7994	0.9636	0.0277	0.9204
QPSO	0.6793	0.0500	0.7479	0.6952	0.3628	0.7468	0.6737	0.0495	0.6699

Ov. :overshoot Obj.: objective value

Table2 The Mean Best Value and Standard Variance of Four Algorithm for G1 (S)

Algorithm	G1 (S) (10 trial runs)					
	Mean Best Value			Standard Variance		
	Ts	Overshoot	Objective value	Ts	Overshoot	Objective value
Z-N	6.2969	0.6229	6.1968			
GA	1.8467	0.32716	1.73251	0.01017488	0.008434	0.0088474
PSO	0.9634	0.10477	1.02986	0.13200059	0.022008	0.0944449
QPSO	0.7178	0.05	0.757714	0.00726694	1.5419E-18	0.0004152

Table3 The Mean Best Value and Standard Variance of Four Algorithm for G2 (S)

Algorithm	G2 (S) (10 trial runs)					
	Mean Best Value			Standard Variance		
	Ts	Overshoot	Objective value	Ts	Overshoot	Objective value
Z-N	0.9107	0.3187	0.9379			
GA	0.8906	0.3531	0.90441	0.00702832	0.0004244	0.00313852
PSO	0.86466	0.35726	0.87227	0.00644614	0.0004035	0.00274499
QPSO	0.71692	0.37323	0.75174	0.00047648	0.0001204	1.64671E-5

Table4 The Mean Best Value and Standard Variance of Four Algorithm for G3 (S)

Algorithm	G3 (S) (10 trial runs)					
	Mean Best Value			Standard Variance		
	Ts	Overshoot	Objective value	Ts	Overshoot	Objective value
Z-N	2.5432	0.3215	2.3935			
GA	1.50522	1.2714	1.51345	0.59046490	0.00316088	0.42005622
PSO	1.06639	0.1403	1.12283	0.02125771	0.01300166	0.052056271
QPSO	0.72512	0.04921	0.71601	0.00727884	2.1365E-06	0.007417961

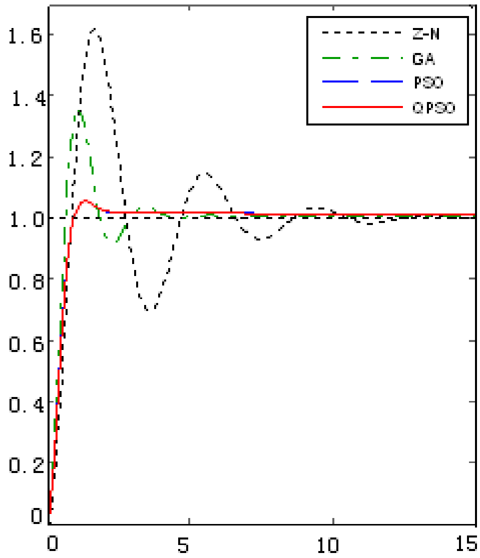

Fig.2 step response of G1 (S) for PID with for algorithm

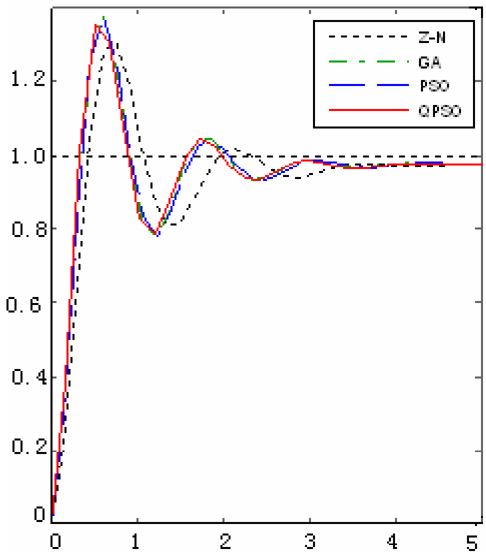

Fig.3step response of G2 (S) for PID with for algorithm

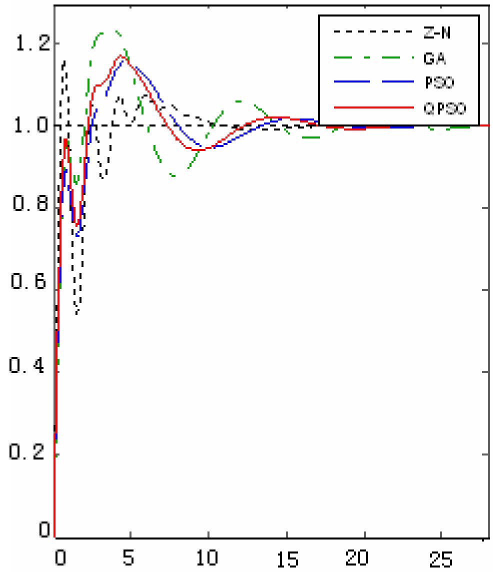

Fig.3step response of G2 (S) for PID with for algorithm

The numerical results in table 1 show that the QPSO could hit the optimal solution with high precision and faster convergence speed and has better experiment values of settling time, overshoot and objective value than other three algorithms. The experiment results of mean best performance values and standard variances of the four algorithms are given in Tables 2 to table 4, and the QPSO has the smaller values. They show that QPSO works better than other three algorithms in three different plants and has superior stability and global searching ability. Fig2 to Fig4 give the comparison of step response for the three plants using four algorithms, the curves show that the QPSO approach has better performance than other three algorithms with smaller settling time and overshoot. From the results above, we can conclude that the calculation method of QPSO can make the convergence speed for PID parameters optimisation problems faster with good global searching ability.

Ⅴ CONCLUSIONS

In this paper, we have described the Quantum-behaved Particle Swarm Optimisation and three PID controllers design via quantum-behaved particle swarm optimisation has been proposed. In this work, simulation results has showed by the comparison study with Z-N, GA, PSO indicate that QPSO can offer an effective and simple method to design PID controllers. Furthermore, it can also give very good tracking response with only one parameter. Actually, the proposed algorithm can also be applied to other optimal control designs..

REFERENCES

[1] Ziegler, J. G. and Nichols, N. B. 1942. Optimum setting for automatic controllers, *Trans. ASME 64*, Page(s): 759–768.

[2] Ching-Hung Lee, Yi-Hsiung Lee, Ching-Cheng Teng, 2002."A novel robust PID controllers design by fuzzy neural network", *American Control Conference*, Page(s): 1561 – 1566,

[3] Shan Li, Jinhua Li, 2006. "IMC-PID Control of Ultra-Sonic Motor Servo System Based on Neural Network", Intelligent Control and Automation, WCICA. Page(s): 8275 – 8279.

[4] Qiao, W. and Mizumoto, M, 1996. " PID type fuzzy controller and parameters adaptive method", *fuzzy sets and system*. Page(s): 23–35.

[5] Yu Yang, Wen-Ge Wang, De-Jie Yu, Ge Ding, 2002. "A fuzzy parameters adaptive PID controller design of digital positional servo system", *Machine Learning and Cybernetics*, ICMLC, Page(s): 310 - 314.

[6] Huang Yourui, Chen Xiuqiao, Hu Yihua, 2005. "Optimisation for Parameter of PID Based on DNA Genetic Algorithm", *Neural Networks and Brain*, ICNN&B, Page(s): 859 – 861.

[7] Hu Yunan, Bin Qu, 2006 ."Application of Iterative Learning Genetic Algorithms for PID Parameters Auto-Optimization of Missile controller", *Intelligent Control and Automation*, WCICA, Page(s): 3435 – 3439.

[8] P.B. de Moura Oliveira, J. Boaventura Cunha, and J. Paulo Coelho (Portugal), 2002. "Design of PID Controllers using the Particle Swarm Algorithm". *The IASTED Conference on Modelling, Identification, and Control* , Page(s): 350-359.

[9] Yijian Liu, Jianming Zhang, Shuqing Wang, 2004."Optimisation design based on PSO algorithm for PID controller", *Intelligent Control and Automation*, WCICA. Page(s): 2419 – 2422,.

[10] F. Van den Bergh, 2001. "An Analysis of Particle Swarm Optimizers," PhD Thesis, University of Pretoria.

[11] J. Sun, B. Feng and W. Xu, 2004. "Particle Swarm Optimization with Particles Having Quantum Behavior," *IEEE Proceedings of Congress on Evolutionary Computation*. Page(s): 325-331.

[12] J.Kennedy and R.Eberhart. 1995. "Particle Swarm Optimization", *proceedings of IEEE international Conference On Neural Network,* Page(s): 1942~1948.

[13] J.Sun, W.Xu, B.Feng, 2004. "A Global Search Strategy of Quantum-Behaved Particle Swarm Optimization", *Cybernetics and Intelligent Systems Proceedings of the 2004 IEEE Conference,* Page(s): 111-116.

[14] Wenbo Xu and Jun Sun, 2005. "Adaptive Parameter Selection of Quantum-Behaved Particle Swarm Optimization on Global Level", *International Conference on Intelligent Computing,* ICIC (1), Page(s): 420-428.

[15] Wei Fang, Jun Sun, Wenbo Xu, 2006. " Design IIR Digital Filters Using Quantum-Behaved Particle Swarm Optimization", *International Conference on Natural Computation,* ICNC (2), Page(s): 637-640.

[16] Jun Sun, Wenbo Xu, Jing Liu, 2006. "Training RBF Neural Network Via Quantum-Behaved Particle Swarm Optimisation", *International Conference on Neural Information Processing*, ICONIP (2), Page(s): 1156-1163,