GA Tuning PID Controller Based on Second-order Time-delay Industrial System

Zhi Wang[1*], *Xiangtao* Ran[1], *Bin* Zhao[1], *Jie* Zhao[1]

[1]Institute of Oceanographic Instrumentation, Qilu University of Technology (Shandong Academy of Sciences), Shandong Provincial Key Laboratory of Marine monitoring instrument equipment technology National Engineering and Technological Research Center of Marine Monitoring Equipment, Qingdao, China

Abstract. For a typical second-order time-delay system model, an intelligent genetic algorithm is used to initially optimize the initial parameters of the PID controller, and a step response curve of the system is obtained, and the performance index is compared with performance indexes obtained by other optimization methods. The results show that the GA optimization method is more robust.

1 Introduction

The second-order time-delay system is a typical system model in industrial systems, and its practical application is extensive. Because of its simple structure and easy implementation, PID control is widely used in today's industrial process control. Therefore, the parameter optimization design of PID controller has been extensively studied and has certain social and economic benefits.

Genetic algorithm (GA) is a widely applied and efficient random search and optimization method based on the principle of biological evolution theory. It is a global optimization algorithm derived from the idea of biological evolution in nature. It can directly manipulate structural objects, evaluate multiple solutions in the search space, reduce the risk of falling into local optimal solutions, and the algorithm itself is easy to implement.

This paper adopts intelligent genetic algorithm to optimize the three parameters of proportional, integral and differential of PID controller. The method is practical and feasible, reducing the time to adjust the initial parameters of PID controller and improving the operating efficiency in industrial production!

2 Genetic Algorithm Tuning PID Parameters

2.1 The Genetic Algorithm (GA)

As a new global optimization search method, genetic algorithm (GA) is robust, simple and universal [1-2].

Genetic algorithm belongs to the aspect of evolutionary computation. It is an iterative adaptive probabilistic search algorithm based on the mechanism of natural selection and natural genetics. It embodies the competition mechanism of “survival competition, survival of the fittest, survival of the fittest”. The algorithm was first proposed by Professor Holland in the United States in 1962 to solve various search and optimization problems encountered in scientific research and engineering practice. In 1975, the publication of Professor Holland's monograph <Adaptation in Natural and Artificial systems> marked the establishment of genetic algorithm, as a new global optimization search method, it has the characteristics of strong robustness, simple and universal, adaptable to dynamic changes, and has the ability of self-optimization.

Although no mathematical theory that can completely clarify the evolutionary characteristics of genetic algorithms has been found yet, in order to be able to explain to some extent the evolutionary characteristics of genetic algorithms, only a few methods for describing such evolutionary characteristics are used as their The basic theory, in which the pattern theorem is the most important one, was originally proposed by Professor Holland, and has been developed into the most mature and important theory of genetic algorithms.

The model theorem clarifies the operating mechanism of the genetic algorithm, that is, only the model with short definition, low order and mode average fitness function value above the population average fitness function value, the number of which will increase continuously during the execution of the genetic algorithm. The model is evolved toward the direction of optimal search, and finally the global optimal solution is obtained, which reflects the survival of the fittest in the process of biological evolution.

The basic flow of a genetic algorithm (GA) is shown in Fig. 1.

* Corresponding author: sduwangzhi@163.com

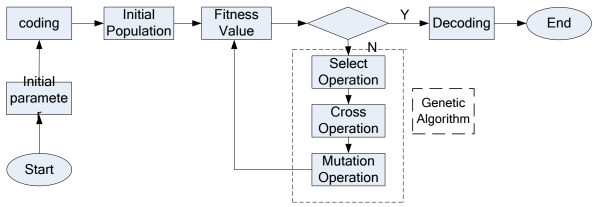

Fig. 1 GA basic flow chart.

The first step: the determination and representation of the parameters

Firstly, the range of parameters is determined. It is determined by the user according to the actual situation, and then the parameters are encoded. There are various encoding methods, such as binary coding, decimal coding, dynamic coding and structural coding. Strings are used to represent parameters, and correspondences between parameters are established. Finally, strings representing various parameters are concatenated to form a "chromosome" that can be genetically manipulated.

The second step: generate the initial population.

The initial population is randomly generated. For binary coding, the size of the population is first determined according to the actual situation, and then the corresponding random function can be conveniently used to generate the initial population.

The third step: calculate the adaptation value.

In the optimal solution search process, the genetic algorithm does not need external information, and only relies on the fitness function as the sole basis, and searches for the optimal solution according to the individual adaptation values.

The selection of the fitness function should meet the following conditions:

i. Single value, continuous, non-negative, maximized;
ii. Reasonable and consistent, can reflect the pros and cons of the corresponding solution;
iii. The amount of calculation is small, which can reduce the complexity of the calculation process;
iv. It is highly versatile. For a similar problem, use the same fitness function as much as possible.

The fourth step: genetic manipulation

The selection operation is performed according to the calculated adaptation value, and chromosomes with good quality are selected, and then the chromosomes are cross-operated according to the crossover probability, and finally the mutation operation of the chromosomes according to the mutation probability is performed.

Through the above process, a new generation of populations is obtained, and the newly generated generations are decoded, and then the individual adaptation function values are calculated to determine whether the constraints can be satisfied. If they are satisfied, the optimization ends; if not, the third and fourth steps are performed cyclically until the requirements are met [3-4].

2.2 The PID controller

PID control is one of the earliest developments of control strategies. It is based on the linear factors of the systematic factors (proportional), past factors (integration) and future factors (differential) to determine the control amount.

PID controller has the advantages of simple structure, easy implementation and strong robustness. It is the most classic and widely used control strategy in the field of industrial control.

The basic idea of the PID controller is to perform Proportional, Integral, and Derivative operations on the current error of the system, and then use the weighted combination of the results of the three operations as a control variable to control the controlled object [5]. The loop formed by the controlled object and the PID controller is called a PID control system. The principle of the PID controller is shown in Fig.2:

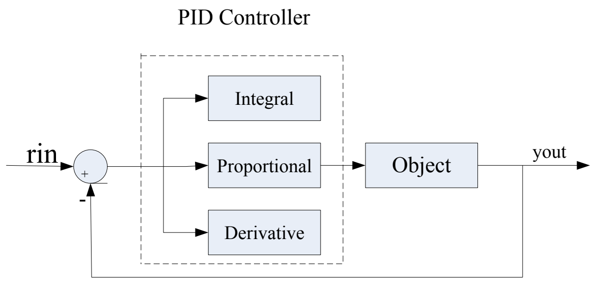

Fig. 2 PID controller schematic.

The proportional action is proportional to the deviation and it determines the response speed. The smaller the proportional coefficient, the weaker the adjustment effect, this will inevitably lead to a slow change of the process curve, a long oscillation period and a large attenuation.

The integral action is to make the system stable without static difference, and the strength of integral action depends on the size of the integral coefficient. The larger the K_i, the stronger the integral action, the faster the system static elimination is eliminated. However, if Ki is too large, the integral saturation phenomenon will occur in the initial stage of the response process and during the transition process, which will cause a large overshoot in the response process, resulting in poor dynamic performance and unstable operation. The smaller the K_i, the weaker the integral action, but if Ki is too small, it will make the static difference difficult to eliminate and make the transition process time longer.

The differential action is to improve the dynamic characteristics of the system. It only affects the rate of change of the system deviation. Its role is mainly to suppress the change of the deviation in any direction during the response process, to brake the deviation in advance, reduce overshoot and increase the stability of the system. The strength of the differential action mainly depends on the magnitude of the differential time kd. The larger the kd is, the stronger the differential action is, and the stronger the regulation action is. However, if the differential action is too large, the response process will be excessively braked and the oscillation will be excessive. And too small is not obvious enough. Therefore, how to adjust the proportional band, integration time and differential time is the key of PID regulation [6-7].

The three parameters of the PID controller must be set according to the specific conditions of the engineering control. In the field of engineering control, the controlled process is required to be stable, the response is rapid and accurate, and the overshoot is small. Under different disturbances, the output of the system should be stable at a given value, and the fluctuation of the controlled amount should be kept within a certain range and not too large. And the system control should remain stable after system and environmental parameters change.

The differential equation of the PID controller in the time domain can be expressed as:

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) \mathrm{d}\tau + K_d \frac{\mathrm{d}e(t)}{\mathrm{d}t} \quad (1)$$

Where e(t) and u(t) represent the error signal and the output signal of the controller, respectively. kp, ki, and kd are controller parameters, which are called proportional coefficient, integral coefficient, and differential coefficient.

Using the Laplace transform, the PID controller's transfer can be expressed as:

$$G_c(s) = K_p + K_i/s + K_d s \quad (2)$$

The performance of the PID controller is closely related to the value of its parameters. To achieve satisfactory dynamic performance, the three parameters of the PID must be set properly. The traditional PID controller parameter setting methods include Z-N tuning method, C-N tuning method, critical scaling method, etc. However, these methods often have disadvantage of complicated processes, poor setting effect, poor adaptability to operating conditions, and easy to appear large overshoot and shock.

2.3 Tuning PID controller with GA

The principle of using intelligent genetic algorithm to tune the control parameters of PID controller is shown in Figure 3 [8]:

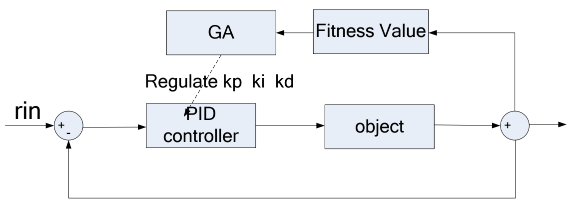

Fig. 3 GA tuning PID controller schematic.

The initial parameters of the initial parameters kp, ki, and kd of the PID controller are all set to [0, 10].

In this paper, the initial population number is set to 30, the encoding method is real-coded, and the roulette selection method is combined with the optimal individual conservation method. The crossover probability pc is chosen as 0.9 and the evolutionary algebra is taken as 300.

In the selection stage, the most classic roulette selection method and the optimal individual preservation method are used. The greater the individual fitness value and the proportional value of the whole population fitness value, the greater the probability of being selected. But in theory, the optimal individual does not necessarily enter the next generation, so in order to ensure that the best individual is necessarily selected into the next generation, we have increased the use of the optimal individual preservation method, proposed the optimal individual and then selected the entire population, the selection operating procedure is as follows:

```
pi=fi/fi_sum;
q(1)=pi(1,1);
for i=2:1:size
  q(i)=pi(1,1);
end
pp=rand(1,size)
for i=1:1:size
  if pp(1,i)<q(1)
    tempe(1,:)=kpid(1,:);
  end
  for j=2:1:size
    if (pp(1,i)<=q(j))&&(pp(1,i)>=q(j-1))
      tempe(i,:)=kpid(j,:);
    end
  end
end
```

The crossover stage uses the linear combination crossover commonly used for real coding, and the crossover probability is Pc=0.9. The crossover procedure is as follows:

```
pc=0.9;
for i=1:2:(size-1)
   temp=rand;
   if pc>temp
     alfa=rand;
     tempe(i,:)=alfa*kpid(i+1,:)+(1-alfa)*kpid(i,:);
     tempe(i+1,:)=alfa*kpid(i,:)+(1-
alfa)*kpid(i+1,:);
   end
end
```

According to the design, the mutation procedure is as follows:

```
pm=0.10-[1:1:size]*(0.01)/size;
pm_rand=rand(size,codel);
mean=(maxx+minx)/2;
dif=(maxx-minx);
for i=1:1:size
   for j=1:1:codel
      if pm(i)>pm_rand(i,j)
         tempe(i,j)=mean(j)+dif(j)*(rand-0.5);
      end
   end
end
```

At this point, the second generation is generated and then iterated until it reaches Convergence condition.

3 Typical second-order delay function application

The controlled object is a second-order typical delay function, and the transfer function is as follows in Eq. 3. In this paper, τ=1:

$$f(s)=\frac{e^{-\tau s}}{(1+s)^2} \quad (3)$$

The fitness function is directly related to the tuning result of the optimization parameters and the control quality of the regulation system. Using different performance indicators, the final parameter tuning results will be different, and accordingly, the system's control quality will be different. The performance index must be able to comprehensively reflect the control quality of the system and be convenient for analysis and calculation. In this work, the ITAE index (integration of the time and the absolute value of the error on the time integral, see the following Eq. 4. This index has better practicability and selectivity (the greater the change of the system parameters causes the change of the index, the better the selectivity), which is the objective function describing the performance of the system comprehensively. It is also considered as the best performance indicators for single in-out control systems and adaptive control systems by many documents. At the same time, the criterion is easy to implement on a computer, and the optimal parameter combination scheme obtained by it can make the system have good stability, fast response time and small overshoot. [8].

$$J=\int t\left|e(t)\right|dt \quad (4)$$

According to the determined fitness function, set the input to a step response with an amplitude of 1 and write the m file in MATLAB and run it independently for 10 times. The optimal PID controller parameters are as follows: kp=0.9358, ki=0.5967, kd= 0.4758, The final step response curve of the second-order time-delay function is shown in Fig. 4. The evolution curve of the performance index is shown in Fig. 5. Generally, the actual output is within ±2% of the error range, and the system is stable, according to the obtained response curve. The system output is stable after 4.75s, and the maximum overshoot is 2.7%.

In literature [9], the optimal system step response obtained by using the Z-N setting method, IPSO, FPA (basic flower pollination algorithm) and FPAQOX (orthogonal flower pollination algorithm) to set a typical second-order delay model is given in Table1.

In literature [9], the optimal system step response obtained by using the Z-N setting method, IPSO, FPA (basic flower pollination algorithm) and FPAQOX (orthogonal flower pollination algorithm) to set a typical second-order delay model is given in Tab.1.

Table 1. The index of step response.

Algorithm	kp	ki	kd	σ%	ts	ITAE
FPA	1.2737	0.4476	0.8624	2.6525	6.2953	3.0646
IPSO	1.2876	0.4497	0.8685	2.8000	6.3500	3.0419
Z-N	1.6200	0.6707	0.9782	2.7600	8.7500	4.4524
FPAQOX	0.7994	0.0010	0.9031	7.3896	7.4618	6.3229

Among the four intelligent algorithms, when using the IPSO algorithm, the overshoot of system is 2.8%, and the corresponding responding time is 2.8s. When using the Z-N tuning method, the overshoot of system is 2.76%, and the corresponding responding time is 8.75s. When using the FPAQOX algorithm. At the time, the overshoot of system is 7.39%, and the corresponding responding time is 7.46s, when the FPA algorithm is used, the system has the smallest overshoot, which is 2.65%, but the corresponding response time is 6.29s. Compared with the GA algorithm used in this paper, the two systems produce system overshoots. The difference is not large, but the use of GA algorithm to achieve a stable response time increased by 1.54s; visible, based on the typical second-order time-delay model, using PID control, the use of GA algorithm has a clear advantage.

4 Conclusions

Based on the PID control of a typical second-order time-delay model commonly used in industry, GA (genetic algorithm) is used to optimize the initial parameters of the PID controller, and the optimization results are compared with the results of flower pollination algorithm optimization in [9]. In contrast, the final simulation results show that the genetic algorithm is used to optimize the PID parameters, the system's overshoot is not much different, but the system response time has been greatly improved.

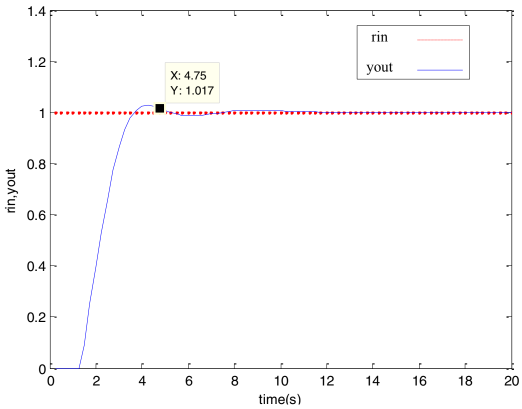

Fig. 4. Step response curve.

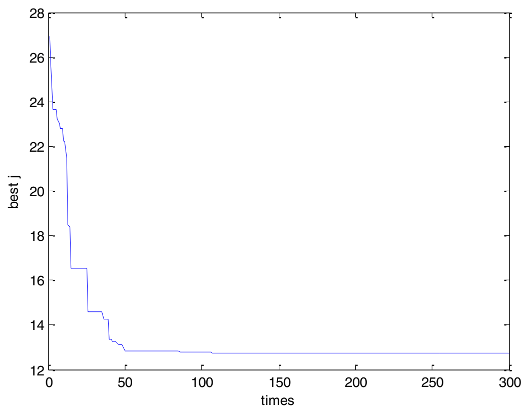

Fig. 5 Performance index evolution curve.

5 Acknowledgement

This work is supported by Shandong Provincial Natural Science Foundation (ZR2018LD008).

References

1. Shiyong Li, *Fuzzy control, neural control and intelligent control theory*. (2004)
2. Chuncheng Wang, Design and Simulation Based on Improved Genetic Algorithm for Gas Mixture Control System, *Chinese Hydrauulics&Pneumatics*, **11**,81(2017)
3. Yao Liu, PID Control System of Temperature Chamber of Biochip Reaction Instrument Based on Genetic Algorithm, *Computer Measurement & Control*, **20**,2410-2412, (2012)
4. Xuebin Li, PID Controller Parameters Setting in Hydraulic Turbine Based on Multi-objective Optimization and Multi-attribute Decision, *Electric Power Science and Engineering*, **25**,45-48,(2009)
5. Yujiao Lin, *Research on PID parameters tuning based on improved particle swarm*. (2015)
6. Qian Zhang, PID Controller Parameters Optimization Method Based on Genetic Algorithm, *Electric Power Science and Engineering*, **27**, 53-57,(2011)
7. Jianqiu Cao, Compound Control Strategy of Fuzzy+Variable Universe Self-adaptive Fuzzy-PID Based on Genetic Algorithm Optimization, *information and control*, 40, 44-49, (2011)
8. Jinbiao Liu, *Advanced PID Control and Its MATLAB Simulation*., (2003)
9. Shuhui. Xu, *Principle and engineering application of improved flower pollination optimization algorithm.*, (2018)