Optimization of PID parameter tuning for gravity stabilized platform based on improved differential evolutionary algorithm

Xin Chen*, Fangneng Li, Hongyang He and Miao Wu

College of Electrical Engineering, Naval Univ. of Engineering, Wuhan, Hubei 430033, China
E-mail: duan@whut.edu.cn

Abstract. To address the difficulty of the PID controller parameters tuning of gravity stabilized platform, an improved differential evolutionary (IDE) algorithm containing two currently better variational strategies is proposed. In the process of algorithm iteration, the mutation factor and crossover probability factor of the algorithm are gradually changed to improve the optimization performance of the algorithm. On this basis, the improved differential evolution algorithm is applied to the PID controller parameter tuning problem of gravity stabilized platform. Simulation results show that the algorithm is effective and can be used to optimize the parameters of PID controller of gravity stabilization platform.

1. Introduction

Earth gravity field observation is an important part of the study of geospatial information and undersea navigation based on gravity information[1,2]. Meanwhile, it also has important application value in geoscience research, seabed mineral development and seismic activity monitoring. The function of gravity measurement stabilization platform is to ensure the absolute vertical state of gravimeter for gravity measurement and ensure that the measured gravity direction is consistent with the actual gravity direction. Therefore, gravity measurement stabilization platform plays a very important role in gravity measurement. Among the control methods of stable platform, PID control is most used [3], which has the advantages of simple structure, easy implementation and strong robustness.

2. Differential evolutionary algorithm

Differential evolution (DE) algorithm was proposed by Rainer storn and Kenneth price in 1995 [4]. It is a random heuristic search algorithm formed by simulating the evolutionary principle of biological population in nature - "survival of the fittest and survival of the fittest". It is a stochastic heuristic search algorithm that simulates the evolutionary principle of "survival of the fittest" in natural biological populations. Differential evolutionary algorithm is an intelligent optimization algorithm based on population theory. It performs optimization search through competition and cooperation among individuals in the group. It is a global search strategy based on population theory. It adopts real number coding, variation operation based on difference and one-to-one competitive survival strategy, which reduces the complexity of genetic operation. At the same time, it has certain memory ability. It can adjust the search strategy in real time according to the current search situation, and has strong global search ability and strong robustness. The main advantages of differential evolution algorithm are less undetermined parameters, not easy to fall into local optimization and fast convergence speed. Compared
with other traditional optimization algorithms, differential evolution algorithm has the following characteristics [5].

- It has the ability to deal with non-differentiable, nonlinear and multimodal functions;
- The possibility of parallel processing of intensive cost functions.
- Ease of use, which can be minimized using few, robust and easily selectable control variables.
- Good convergence to a global minimum in successive independent experiments.

The basic flow of the DE algorithm consists of four basic processes: generation of the initial population, mutation operation, crossover operation, and selection operation.

2.1. Generate initial groups
Differential evolution algorithm is a parallel direct search method, which generates a population individual \( x_{ij}(0) \) of the j-dimensional component of the i-th individual in an n-dimensional space, and searches the largest range in the specified maximum search boundary \( U_{ij} \) and minimum search boundary \( L_{ij} \) as far as possible. The generated component formula is as follows.

\[
x_{ij}(0) = rand_{ij}(0,1)(U_{ij} - L_{ij}) + L_{ij}
\]

where \( rand_{ij}(0,1) \) is a random fractional number between [0, 1].

2.2. Mutation operation
In the initial population, several individuals different from the current individual are randomly selected, and then operate according to different mutation strategies to generate new individuals. Five variation strategies commonly used by de are as follows[6].

- DE/rand/1:
  \[
h_{ij}(t + 1) = x_{ij}(t) + F(x_{r1}(t) - x_{r2}(t))
  \]

- DE/best/1:
  \[
h_{ij}(t + 1) = x_{best}(t) + F(x_{i1}(t) - x_{i2}(t))
  \]

- DE/target-to-best/1:
  \[
h_{ij}(t + 1) = x_{i}(t) + F(x_{best}(t) - x_{i}(t)) + F(x_{r1}(t) - x_{r2}(t))
  \]

- DE/rand/2:
  \[
h_{ij}(t + 1) = x_{ij}(t) + F(x_{r1}(t) - x_{r2}(t)) + F(x_{r3}(t) - x_{r4}(t))
  \]

- DE/best/2:
  \[
h_{ij}(t + 1) = x_{best}(t) + F(x_{r1}(t) - x_{r2}(t)) + F(x_{r4}(t) - x_{r5}(t))
  \]

2.3. Crossover operation
Crossover operations are performed to increase the diversity of the population as follows.

\[
v_{ij}(t + 1) = \begin{cases} h_{ij}(t + 1), & rand_{ij}(0,1) \leq CR \\ x_{ij}(t), & rand_{ij}(0,1) > CR \end{cases}
\]

2.4. Select Operation
To determine whether the target individual \( x_{i}(t) \) can become a member of the next generation, the experimental individual \( v_{i}(t + 1) \) and the target individual \( x_{i}(t) \) are compared with the fitness function according to the following formula.

\[
x_{i}(t + 1) = \begin{cases} v_{i}(t + 1), & f(v_{i}(t + 1)) < f(x_{i}(t)) \\ x_{i}(t), & f(v_{i}(t + 1)) \geq f(x_{i}(t)) \end{cases}
\]
3. Improved differential evolution algorithm

3.1. Mutation Strategies

There are many forms of improvement of differential evolution algorithms, and almost all of them try to achieve a balance between global search ability and local exploitation ability. Comprehensive information of current literature[7,8], all the variation strategies of DE algorithm, among which DE/rand/1 is the most widely used and beneficial to maintain the population diversity; while DE/best/2 has the optimal solution[9,10], which is more beneficial to solve some technical problems of the algorithm and accelerate the convergence speed of the algorithm. However, the variational strategy with optimal information is easy to fall into local optimum, and all design an improved differential evolutionary algorithm containing two variational strategies. Based on the above, this paper proposes an evolutionary algorithm containing the above two variational strategies, which will be combined by a certain ratio, and the specific computational procedure is as follows.

\[
\begin{align*}
    \mathbf{h}_1(t+1) & = x_{r_1}(t) + F(x_{r_2}(t) - x_{r_3}(t)) \\
    \mathbf{h}_2(t+1) & = x_{\text{best}}(t) + F(x_{r_1}(t) - x_{r_2}(t)) + F(x_{r_4}(t) - x_{r_5}(t)) \\
    \mathbf{h}(t+1) & = \text{lamda} \times \mathbf{h}_1(t+1) + (1 - \text{lamda}) \times \mathbf{h}_2(t+1)
\end{align*}
\]

Where \( \text{lamda} \) in Eq. (11) is the proportion of DE/rand/1 in the variation strategy. For different problems, adjust the value of \( \text{lamda} \), and then the proportion of the two variation strategies in the final variation strategy is adjusted, so as to better balance the global search ability and convergence speed.

3.2. Variance factor F

Variation factor F mainly controls the search step of differential evolution algorithm, and then affects the diversity and convergence of algorithm population. In the process of population evolution, with the decrease of mutation factor F, the population diversity decreases, and the algorithm is prone to premature convergence; With the increase of mutation factor F, the population diversity increases, and the algorithm is easy to jump out of the extreme value, but it will affect the convergence speed. In the standard differential evolution algorithm, the variation factor f generally takes a fixed value between \([0,2]\), which can not make full use of the characteristics of each stage of algorithm evolution. Therefore, this paper uses a variation factor that dynamically adjusts with the number of iterations as follows.

\[
\begin{align*}
    x(G) & = e^{\frac{1-G_{\text{max}}}{G_{\text{max}}-G}} \\
    F(G) & = F_{\text{min}} + x(G)(F_{\text{max}} + F_{\text{min}})
\end{align*}
\]

In Eq. (12), \( G \) represents the current number of iterations and \( G_{\text{max}} \) represents the maximum number of iterations; In Eq. (13), \( F(G) \) represents the value of variation factor of current iteration times, \( F_{\text{max}} \) and \( F_{\text{min}} \) represent the maximum and minimum value of variation factor respectively.

3.3. Crossover probability factor CR

The crossover probability factor CR can control the participation of each randomly selected mutation vector in the crossover, and can balance the ability of local search and global search. The crossover probability factor Cr is generally selected between \([0,1]\). If it is too small, the population diversity will be reduced and it is easy to converge prematurely; If the selection is too large, the convergence speed will be too slow, and the influence of individual disturbance will be greater. In conventional differential
evolution, the crossover probability factor CR will choose a fixed value, which ignores the changes of population in the process of iterative evolution. A more reasonable setting is that with the increase of the number of iterations, the crossover probability factor CR gradually increases and the variation factor f gradually decreases, which will gradually accelerate the convergence speed of the population. The formula for adjusting the crossover probability factor Cr is as follows.

\[
CR = CR_{\text{min}} + \left(\frac{CR_{\text{max}} - CR_{\text{min}}}{G_{\text{max}}}\right)G
\]  

3.4. Improved differential evolution algorithm flow

The flow chart of IDE algorithm as shown figure 1.

4. Optimization of PID parameter tuning based on improved differential evolutionary algorithm

At present, most of the controllers used in industrial production are PID or PID variant controllers. The PID controller has simple structure and convenient use. The control quantity is composed of deviation proportion (P), integral (I) and differential (d) through a certain linear combination, so as to realize the control of the controlled object. The PID control system is composed of PID controller and controlled object. The control law of PID controller is written in the form of transfer function as follows.

\[
u(t) = k_p e(t) + k_i \int_0^t e(t)\,dt + k_d \frac{de(t)}{dt}
\]  

The performance of PID controller largely depends on the value of \(K_p\), \(K_i\) and \(K_d\). The improved search evolutionary algorithm is to find the best value. In order to obtain the best transition dynamic process, the absolute deviation integral performance index is selected as the minimum objective function of parameter selection. At the same time, in order to prevent excessive control energy, the square term of control input is added to the objective function. In order to avoid overshoot, penalty measures are
added. Once overshoot, the overshoot is taken as a part of the objective function. At this point, the objective function as follows.

\[
J = \begin{cases} 
\int_0^\infty (w_1 |e(t)| + w_2 u^2(t)) dt & e(t) \geq 0 \\
\int_0^\infty (w_1 |e(t)| + w_2 u^2(t) + w_3 |e(t)|) dt & e(t) < 0 
\end{cases}
\]

where \( J \) is the optimal indicator, and \( w_1, w_2 \) and \( w_3 \) are the weights.

5. Simulation experiments

The controlled object chosen in this paper is the gravity stabilized platform, and its model as follows [11].

\[
G(s) = \frac{7.9008}{1.185s + 0.00004}
\]

Construct the simulation model according to equation (3.9), and the set differential evolution simulation parameters are: in order to obtain the balance between calculation time and optimal solution, the dimension of population number is 3, and the population size \( NP = 20 \); The maximum value of the variance factor \( F_{\text{max}} = 1.8 \) and the minimum value \( F_{\text{min}} = 0.4 \); the maximum value of the crossover probability factor \( CR_{\text{max}} = 0.8 \) and the minimum value \( CR_{\text{min}} = 0.2 \); the maximum number of iterations \( G_{m} = 500 \); the objective function weights \( w_1 = 0.999, w_2 = 0.001 \) and \( w_3 = 10 \). After 500 generations of evolution, the rectified results are \( \text{Best } J = 22.62, k_p = 126.45, k_d = 0.1023 \) and \( k_i = 6.13 \). The variation curve of the optimal index in the iterative process is shown in Figure 2.

![Figure 2. Variation curve of optimal index](image)

![Figure 3. System step response curves.](image)

It can be seen from Figure 2 that the optimal index Best \( J \) decreases continuously in the iteration. It can be seen that the convergence speed of IDE is fast, the algorithm has a large optimization range, and the found optimal parameters make the PID controller have better control performance. In order to verify that the parameters optimized by ide algorithm have better control performance in PID, it is compared with other algorithms. The comparison of step response curve is shown in Figure 3.

Figure 3 shows that the PID controller with optimal parameters of IDE algorithm has better control performance, and has obvious control advantages in terms of overshoot and regulation time, which verifies the effectiveness of IDE algorithm.
6. Conclusions
In this paper, an improved differential evolution algorithm is proposed for the tuning of control parameters of gravity stabilized platform. The algorithm has the combination of two excellent mutation strategies, and can adjust the proportion of the combination according to the actual problem to make it more meet the needs of solving the problem. At the same time, the mutation factor and crossover probability factor varying with the number of iterations are also used in the algorithm, which makes up for the deficiency that it is a fixed value in the conventional differential evolution algorithm. The IDE algorithm proposed in this paper is used to adjust the control parameters of gravity stabilized platform. The simulation results show that compared with the traditional PID algorithm and conventional differential evolution algorithm, it has the advantages of small overshoot, short adjustment time and fast response speed, and has better dynamic and steady-state performance.

References
[1] Huang MT, Liu M, Wu TQ, Lu XP, Deng KL, Zhai GJ, OuYang YZ, Chen X. (2018) Research and Evaluation on Key Technological Target System for Marine and Airborne Gravity Surveys. Acta Geodaetica et Cartographica Sinica, 47(11):1537-1548.
[2] Wu MP, Cai SK, Yu RH, Cao JL. (2020) Research progress on Strapdown Gravimetry Technology. Navigation and Control, 19(Z1):161-169.
[3] Zhang YQ, Zhang Y, You K, Zhao MW, Li YL, Chen GY, Yan XL. (2020)Research on Improved Auto-Tuning of a PID Controller Based on Phase Angle Margin. Nuclear Power Engineering, 41(S2):108-113.
[4] Storn R, Price K. (1996). Minimizing the real functions of the ICEC'96 contest by differential evolution. In Proceedings of IEEE international conference on evolutionary computation, Nagoya. pp. 842-844.
[5] Storn R, Price K. (1997) Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. Journal of Global Optimization, 11(4):341-359.
[6] Qin AK, Huang VL, Suganthan PN. (2008)Differential evolution algorithm with strategy adaptation for global numerical optimization. IEEE transactions on Evolutionary Computation, 13(2):398-417.
[7] Price KV. (2013)Differential Evolution. In: Zelinka I, Snášel V, Abraham A (eds) Handbook of Optimization. Intelligent Systems Reference Library, vol 38. Springer, Berlin, Heidelberg. pp. 187-214.
[8] Das S, Suganthan PN. (2010)Differential evolution: A survey of the state-of-the-art. IEEE transactions on evolutionary computation, 15(1):4-31.
[9] Zhang J, Sanderson AC. (2009)JADE: adaptive differential evolution with optional external archive. IEEE Transactions on evolutionary computation, 13(5):945-958.
[10] Pahner U, Hameyer K. (2000)Adaptive coupling of differential evolution and multiquadrics approximation for the tuning of the optimization process. IEEE Transactions on magnetics, 36(4):1047-1051.
[11] Li A, Li N, Gao Y, Guo B. (2016)Research on Control System of Stable Platform Based on Genetic Algorithm. Ship Electronic Engineering, 36(03):42-46.