

The Improved Particle Swarm Optimization Applied in Auto Carrier Landing System

Liu Baoning, Zhang Weiguo

College of Automation
Northwestern Polytechnical University
Xi'an, China
lbn1987113@163.com

Li Guangwen, Liu Xiaoxiong

College of Automation
Northwestern Polytechnical University
Xi'an, China

Abstract—To solve the problem that standard Particle Swarm Optimization can get into local optimum prematurely, an improved Particle Swarm Optimization algorithm is proposed. The improved algorithm can be used to optimize every parameters of Auto carrier landing system. It improves the population diversity and the searching efficiency of particle swarm by introducing mutation operator based on orthogonal design area and modifying particles speed. The effectiveness of this algorithm is verified by using simulation experiments.

Keywords- Auto carrier landing system, Particle Swarm Optimization, local optimum, optimize parameters

I. INTRODUCTION

Auto carrier landing system (ACLS) is a complicated control system, that mainly consist of flight control system and power compensate system. Owing to the restriction of landing field, incoming landing demands rigid control track precision, which makes the parameter optimization of landing control system essential. Particle Swarm Optimization (PSO) [1] algorithm has few adjustable parameters and is easy to realize through programs. Also in some engineering work, flight control system applies PSO to achieve automatic parameter tuning. However, the standard PSO algorithm can get into local optimum prematurely since it depends on iterative formula solely.

Therefore, the article proposes an improved PSO algorithm to solve this problem. It escapes local optimum by means of introducing mutation operator based on orthogonal design area. The effectiveness of this algorithm is verified using simulation experiments on auto carrier landing system

II. STANDARD PARTICAL SWARM ALGORITHM

Particle swarm algorithm is initially proposed by Doctor Eberhant and Kennedy at 1995. PSO algorithm works through cooperation and competition among individuals, to realize optimum solution during complex space. Every particle in the particle swarm is a potential solution to the required optimization problem [2]. The quality of the particle is determined by a fitness function set in advance.

If there are m particles in a D dimensional searching space. We represent the location of particle i as $X_i = (x_{i1}, x_{i2}, x_{i3}, \dots, x_{iD})$, the speed as $V_i = (v_{i1}, v_{i2}, v_{i3}, \dots, v_{iD})$. By putting X_i into the fitness

function we get fitness value, which can estimate the stand or fall of its position. $P_i = (p_{i1}, p_{i2}, p_{i3}, \dots, p_{iD})$ represents the best location of particle i , $P_g = (p_{g1}, p_{g2}, p_{g3}, \dots, p_{gD})$ represents the best location of the whole particle swarm. In the standard PSO algorithm, the formula to update particle's speed and location is shown as follows:

$$v_{id}^{t+1} = \omega \times v_{id}^t + c_1 \times r_1 \times (p_{id} - x_{id}^t) + c_2 \times r_2 \times (p_{gd} - x_{id}^t) \quad (1)$$

$$x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1} \quad (2)$$

$\omega = \omega_{\max} - \frac{\omega_{\max} - \omega_{\min}}{T} \times t$, where ω_{\max} is the maximum weight and ω_{\min} is the minimum weight, $i = 1, 2, 3 \dots m$, $d = 1, 2, 3 \dots D$. Learning factor c_1 and c_2 are non-negatives (default value is 2); weight r_1 and r_2 are random values between $[0, 1]$. Speed value v is limited by maximum and minimum speed, such as $v \in [v_{\min}, v_{\max}]$, because the speed of particles affects the parameter optimization results.

The initial location and speed of standard particle swarm are generated randomly, which iterate according to iterative formula (1)(2). Finally we get the adjustable value and optimized locations for both individual and unity.

III. AN IMPROVED PARTICLE SWARM ALGORITHM

The process of optimizing standard particle swarm is the process to achieve the optimum location of all the particles. As a result of choosing improper algorithm parameter, the particle diversity dies away during calculative procedure. The prematurity of the algorithm could end PSO up with local optimum. This makes it awfully difficult to achieve global optimum.

When the particle swarms are getting into local optimum, we import a mutation operator based on orthogonal design area. The orthogonal design discretizes the searching area of PSO and calculates the effectual value along with extreme value analysis. It can change the current optimum location of particle swarm and make them fall into the effective area choose by orthogonal test. The process ensures validity of current optimum mutation, changes the direction of whole

particle swarm searching, enlarges the clearance between particles, and makes it easier to escape local optimum.

Orthogonal test is a design to study multiple factor multiple level problems. It chooses typical samples from the whole test according to their orthogonality. These samples have qualities like ‘even decentralization and neat comparison’. The orthogonal test is a high efficiency、high speed 、economical method and is realized via normally used orthogonal test table $L_n(q^s)$. Where n is experiment frequency, s is factor (variable) number, q is each factor’s statement.

The whole particle swarm searching area is decentralized into n horizontal ones and we choose corresponding area to construct an orthogonal test table, works as current optimum mutation area. The fitness function value demands optimizing is the effective value for orthogonal design. First, calculate the effective value of every area sample, and then carry out variance analysis. When the effective value is better, we get bigger weight value. Meanwhile the ratio of optimum location falling into the object area during mutation is larger.

We choose linear depression strategy to

compute ω ,
$$\omega = \omega_{\max} - \frac{\omega_{\max} - \omega_{\min}}{T} \times t$$
. Such strategy guarantees effective global search during the early stage; defines outline of certain area and ensures precise optimum searching in the later stage. In order to search new area effectively after particle swarm’s mutation, the particle speed should be changed along with current global optimum location mutation. Assume that, the global optimum location mutate after iterating for about n times, meanwhile the inertia weight ω grows smaller and particle swarm searching ability drops off. With the purpose to improve searching ability of particle swarm, we regard mutated particle swarm as a new particle swarm and deassign new speed value v_i to all the particles. If the new optimum location is x_1 , and the initial optimum location is x_0 , so we get $v_i = |x_0 - x_1|$ as a result. The pseudo-code of improved algorithm is shown as follows:

```

choose population parameter based on equality design
initialize the population Pgbest and Pibest
discretize the object searching area and implement
orthogonal test
For l=1:T
    If meet mutation condition
        Using probability calculation to evaluate the area of
        current optimum location mutation
        Global optimum location mutate
        and
         $v_i = |x_0 - x_1|$ 
    End
    Update speed value according to particle speed formula
    Update location according to particle location formula
    Calculate adaptable value J of each particle
    Calculate the optimum swarm location: Pgbest
    Calculate the optimum individual location: Pibest

```

End

IV. BASIC PRINCIPLES FOR AUTO LANDING SYSTEM CONTROL LAW OPTIMUM DESIGN BASED ON IMPROVED PSO

Optimum parameter tuning for auto landing longitudinal control system[7] based on improved PSO is shown at figure1:

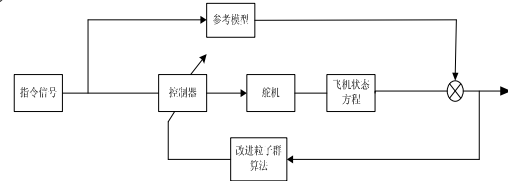


Figure 1. block diagram of longitudinal control system

The basic principle is: the algorithm realizes auto optimization of parameter via continuous computation on optimize target. We use Simulink to bring out simulation and the program is implementing improved PSO to achieve parameter optimization.

We choose the square measure between curves of reference model and practical output as the optimize target,

that is $J = \int |e| dt$. The size of square measure (the smaller, the better) indicates the error between practical output and reference model. Once overshoot appears in the response, we should increase value of adjective function immediately to lessen the overshoot frequency.

Parameter optimization of auto carrier landing system based on improved PSO can be realized as follows:

Step1 Initialize the parameters. We define the number of parameter requires optimization as dimension of particles;

set changing range for D gains; set $[v_{\min}, v_{\max}]$, c_1 , c_2 and ω according to equality design principle. After that, we calculate effective value and put up regression analysis. Finally, initialize particle swarm.

Step2 Use fitness function to calculate fitness value of each particle. We choose $J = \int |e| dt$ as the fitness function and set initial location of each particle as their initial best location Pibest. After that, we set the minimum number during Pibest as global optimum location Pgbest.

Step3 Iterate each particle before calculate their fitness value again; meanwhile keep comparing Pibest ,Pgbest and switch them.

Step4 Calculate mutation probability p if global optimum location of particles remains the same after continuous iteration. If $\text{rand} < p$, mutate current global optimum location. At the same time, we compute speed of particle again according to orthogonal test areas that the probability falls into.

Step5 Go back to step3 unless the iterate number reaches the maximum value or meets certain requirement, then exit.

V. SIMULATION SAMPLE

We use auto landing longitudinal movement of certain type of flight to realize landing control law design.

Longitudinal state formula of the flight is shown as follows:

$$\begin{cases} \dot{X} = AX + BU \\ Y = CX + DU \end{cases}$$

Where the statement variable is : $X = [v \ \alpha \ q \ \theta \ h]^T$,

and the input value is: $U = [\delta_p \ \delta_e]^T$; θ is the pitch angle and α is the attack angle, v is the speed, q is pitch angle rate; δ_e is the angle of elevator deflection, δ_p is the throttle load measuring.

The control law used by the elevator is shown as follows:

$$\delta_e = \left(K_1 \frac{1}{s} + K_2 \right) (\text{Hdot} - \gamma) + K_3 \theta + K_4 q$$

Throttle control law:

$$\delta_p = \left(K_5 + K_6 \frac{1}{s} \right) \alpha + G \cdot \delta_e$$

Where $K_1, K_2, K_3, K_4, K_5, K_6$ are parameters which is need to optimize; G is the cross-linked coefficient.

Simulation diagram chart is as follows:

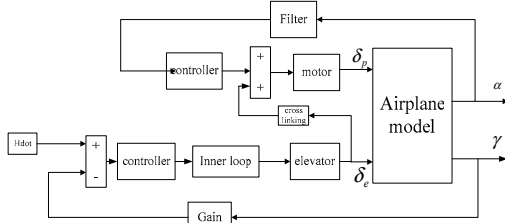


Figure 2. Simulation diagram chart

We apply the improved PSO proposed by this article to realize parameter optimization for this auto carrier landing system. First, choose population parameter based on equality design principle; then decentralize the parameter region K_1, K_2, K_3, K_4, K_5 and K_6 into 5 horizontal regions like $[0, 2], [2, 4], [4, 6], [6, 8], [8, 10]$. We choose intermediate point of each region as the representative point to construct $L_{25}(5^6)$:

TABLE I. $L_{25}(5^6)$ ORTHOGONAL TEST TABLE

Factor trail number	x1	x2	x3	x4	x5	x6
1	1	1	1	1	1	1
2	1	2	2	2	2	2
3	1	3	3	3	3	3
4	1	4	4	4	4	4
5	1	5	5	5	5	5
6	2	1	2	3	4	5
7	2	2	3	4	5	1
8	2	3	4	5	1	2
9	2	4	5	1	2	3
10	2	5	1	2	3	4
11	3	1	3	5	2	4

$J = \int |e| dt$ is the fitness function to compute 25 group of effective values. Implement regression analysis according to reference [6] and choose optimum area. The chance to have optimum value after mutation lied in the selected area is extremely large.

We apply the improved PSO proposed by this article to realize auto optimization for longitudinal flight control system using pitch reference model mentioned before. After 50 times of iteration, we get the optimum result $K_1 = 0.014, K_2 = 0.035, K_3 = 2, K_4 = 0.85, K_5 = 1.1, K_6 = 0.26$. The simulation result of longitudinal Auto carrier landing system is shown in figure 3.

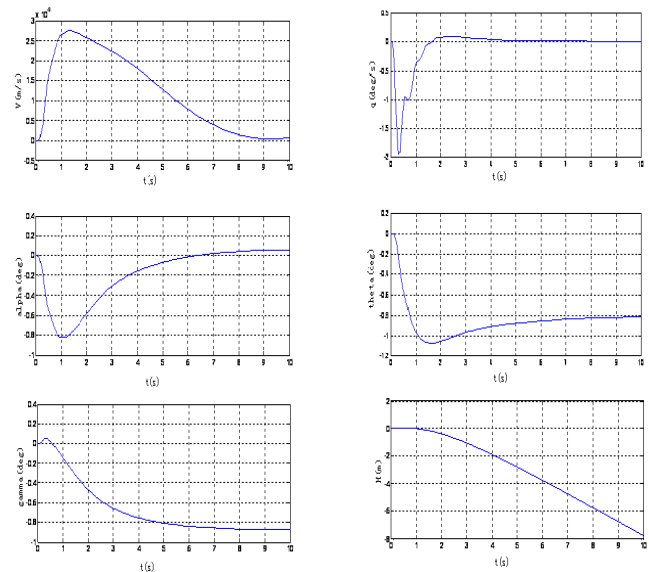


Figure 3. System response after improving optimum parameters of global optimum particle swarm

According to figure 3, the attack angle and pitch angle achieves great result using our improved PSO algorithm. The track angle reaches a constant value rapidly. The speed remains a stabled state as well and the precision changes with height H. These satisfy requirement.

VI. CONCLUSIONS

This article proposes an improved PSO algorithm to realize parameter optimization for Auto carrier landing system. This method finds optimum solution much easier than current methods and the experiment results validate its effectiveness.

REFERENCES

- [1] Kennedy J, Eberhart R C. Particle swarm optimization[C]. Proc of the IEEE International Conference on Neural Networks. Piscataway, NJ: IEEE Service Center, 1995: 1942-1948
- [2] Liu Jin-kun. Advanced PID control and matlab simulation. [M] Beijing : PUBLISHING HOUSE OF ELECTRONICS INDUSTRY, 2006

- [3] Liu Jingjing. Modification and Application of Particle Swarm Optimization Algorithm. [D].Wuhan : Wuhan University of technology, 2007
- [4] Lin Lu; Qi Luo; Jun-yong Liu; Chuan Long. An Improved Particle Swarm Optimization Algorithm [C]. Proc of IEEE International Conference on Granular computing, NJ: IEEE Service Center,2008:486-490
- [5] Li Zhi-jie; Liu Xiang-dong; Duan Xiao-dong; Wang Cun-rui; An Improved Particle Swarm Algorithm for Search Optimization[C]. Proc of IEEE International Conference on Intelligent Systems, NJ: IEEE Service Center,2009:154-158
- [6] Probability and Statistics Group, Peking University, Department of Mathematics and Mechanics Group.Orthogonal design[M] , PETROLEUM INDUSTRY PRESS, 1994
- [7] Wu Sen-tang, Fei Yu-hua. Flight control system [M]Beijing: BEIHANG UNIVERSITY PRESS,2006