Temperature Control Optimization for Heat Pipe Based on Particle Swarm Optimization

Zhu Xi, Du Chunlin, Feng Yao, Wang Jing, Zhu Lin
Beijing Institute of Spacecraft Environment Engineering, Beijing, China
E-mail: 525891876@qq.com

Abstract—In order to optimize the parameters of heat pipe temperature control during the thermal vacuum test, the particle swarm optimization (PSO) algorithm is used to tune the PID control parameters. According to the heat response data of heat pipe, the temperature control system model of heat pipe is established. The time integral of the absolute value of the control error is selected as the objective function. The parameters of the particle swarm optimization algorithm are compared with those of the attenuation curve method, and compares the control result based on parameter from PSO and the attenuation curve method. The simulation result shows that the optimization algorithm can accelerate the convergence rate. The setting time of system is cut down from 10h to 3.7h, and the maximum overshoot decreases from 16.7% to 1.4%. The dynamic performance of system is effectively improved.

1. INTRODUCTION

These are heat pipe is one of the main means of spacecraft thermal control, especially the high-capacity communication satellites. Because of its large thermal power consumption and high power density of transponders, a large number of heat pipes are used on the satellites to form a heat pipe network system to solve the problem of temperature control[1]. In thermal vacuum test, the satellites are divided into zones for temperature control, which are divided by the location of critical equipment. Generally, the temperature control of the key equipment is realized by controlling the temperature of the heat pipe.

The temperature control of heat pipe is usually controlled by PID controller, and the PID control effect is closely related to its proportional, integral and differential three parameters. In order to improve the regulation ability of the traditional PID controller, the intelligent tuning method has become a hot research field in recent years. In [2-3], neural network is used to adjust the parameters of PID controller, and the system performance index is improved remarkably. In [4-5], a fuzzy adaptive PID controller is designed to improve the dynamic performance and robust stability of the system. In [6-7], a PID controller tuning method based on genetic algorithm is proposed. The genetic algorithm is used to optimize the PID parameters in the control process, which improves the control performance and the adaptive ability of PID controller. However, there are still some shortcomings of the above algorithms, such as premature convergence, large computational complexity and strong correlation of the optimized parameters.

In recent years, with the rapid development of computer technology and optimization algorithms, Kennedy and Eberhart creatively proposed Particle Swarm Optimization (PSO) [8]. This algorithm has been extensively studied and applied in recent years because of its easy implementation, fast
convergence speed and small error of optimization results[9,10]. In this paper, the optimization objective function is established by the heat pipe control model. The PID parameters of the objective function are set based on PSO. The simulation result shows that the tuning method is an efficient PID parameter tuning method.

2. SYSTEM CONTROL MODEL
Heat pipe temperature control system consists of computer, DC power, infrared lamp array, heat pipe, thermocouple, data acquisition instrument, the block diagram shown in Fig. 1. In Fig. 1, the controlled object is the heat pipe, the system control quantity is the DC power output current, the output value is the heat pipe temperature.

![Fig.1 Block diagram of temperature control system for heat pipe](image)

According to Fig. 1, the control model of the heat pipe temperature control system is shown in Fig. 2.

![Fig.2 System control model](image)

In Fig. 2, r is the set value, y is the output value, e is the difference between the set value and the output value, u is the controlled variable. \( G_c(s) \) is the PID controller:

\[
G_c(s) = K_p + \frac{K_i}{s} + K_d \cdot s
\]  

(1)

Here in (1), \( K_p \), \( K_i \), \( K_d \) are proportion coefficient, integral coefficient, differential coefficient respectively.

\( G_p(s) \) is the transfer function of the heat pipe, this paper through the analysis of experimental data acquisition to identify the object. Fig. 3 is a satellite heat pipe in the thermal vacuum test of the excitation and response data curve. According to the trend of the open-loop response curve, the transfer function of the heat pipe can be approximated by the first-order inertia link. There are two key parameters, which are the open loop gain and time constant.

![Fig.3 Excitation and Response Data Curve of a Satellite Heat](image)

Fig. 3 corresponds to the heat pipe open-loop gain K is 13.88, the time constant T is 3.7. The incentive is 2.5A, corresponding to the steady-state value of 34.7℃. The heat pipe of the spacecraft is similar to the heat pipe of the other controlled area in the thermal test, except that the parameters of each heat pipe can be different and can be expressed by formula (2).
\[ G_p(s) = \frac{K}{Ts + 1} \]  

(2)

\( G_p(s) \)is the open-loop transfer function of the controlled object, \( K \) is the open-loop gain, \( T \) is the open-loop time constant.

According to formula (1) and formula (2) can be obtained as follows:

\[ \frac{Y(s)}{E(s)} = G_p(s) \cdot G_r(s) = \left( K_p + \frac{K_i}{s} + K_d \cdot s \right) \cdot \frac{K}{Ts + 1} \]  

(3)

3. PID Parameter Tuning Based on PSO

3.1 PSO algorithm description

PSO is an iterative algorithm with global search capability\(^{[11]}\). First, a group of particles is randomly initialized in the feasible solution space. Each particle is a feasible solution of the optimization problem. The degree of the solution is determined by a pre-set objective function. Then, each particle will move in the feasible solution space and its direction and position will be determined by a velocity variable. Finally, the particle follows the current optimal particle, and the optimal solution is obtained by successive search. The mathematical description is as follows\(^{[12]}\):

A population consisting of \( m \) particles flying at a certain velocity in the \( D \)-dimensional space, assuming that \( X = (x_1, x_2, \cdots, x_D) \) is the position vector of the \( i \)-th particle, \( V = (v_1, v_2, \cdots, v_D) \) is the velocity vector of the \( i \)-th particle, \( P = (p_1, p_2, \cdots, p_d) \) is the position of the \( i \)-th particle so far, \( P_o = (p_{o1}, p_{o2}, \cdots, p_{od}) \) is the optimal position of the whole particle swarm to date. In each iteration, the particle is updated according to the formula (5) for velocity and position:

\[
\begin{align*}
v_{d+1}^{i} &= w \cdot v_{d}^{i} + c_1 \cdot \text{rand}(0,1) \cdot (p_{d} - x_{d}^{i}) + \\
& \quad c_2 \cdot \text{rand}(0,1) \cdot (p_{d} - x_{d}^{i}) \\
x_{d+1}^{i} &= x_{d}^{i} + v_{d+1}^{i}
\end{align*}
\]  

(4)

Here in (4): \( i = 1, 2, \ldots, m; d = 1, 2, \ldots, D; k = 1,2, \ldots, N; N \) is the number of iterations; \( w \) is the inertia weight\(^{[13]}\); \( c_1, c_2 \) is the learning factor; \( \text{rand}(0,1) \) is a random number distributed between \([0,1]\).

In addition, the particles are also limited by the maximum flight velocity \( V_{\text{max}} \). The velocity range of the particles is \([-V_{\text{max}}, V_{\text{max}}]\). The flying velocity is shown in formula (5):

\[
\begin{align*}
& \text{if} \quad v_{d} > V_{\text{max}} \quad \text{then} \quad v_{d} = V_{\text{max}} \\
& \text{if} \quad -V_{\text{max}} \leq v_{d} \leq V_{\text{max}} \quad \text{then} \quad v_{d} = v_{d} \\
& \text{if} \quad v_{d} > -V_{\text{max}} \quad \text{then} \quad v_{d} = -V_{\text{max}}
\end{align*}
\]  

(5)

3.2 Establish objective function

The number of particles in the population \( P \) is \( m \), and the position vector of each particle consists of three control parameters of the PID controller, that is, the dimension vector of the particle position vector \( D = 3 \), the population can be represented by a \( m \times D \) matrix:

\[
P(m, D) = \begin{bmatrix}
K_p^1 & K_i^1 & K_d^1 \\
K_p^2 & K_i^2 & K_d^2 \\
\vdots & \vdots & \vdots \\
K_p^m & K_i^m & K_d^m 
\end{bmatrix}
\]

In order to obtain the satisfactory dynamic characteristics of the transient process, the time integral of the absolute value of the control error is selected as the objective function. In order to prevent the overshoot of the control, the square of the control input is added to the objective function, and the penalty control is adopted, that is, once overshoot occurs, the overshoot is taken as the optimal
index. The following formula is used as the optimization objective function, which is to make the following objective function to a minimum:

$$F = \int_{0}^{\infty} \left( w_1 |e(t)| + w_2 |u^2(t)| + w_3 |e(t)| \right) dt$$  \hspace{1cm} (6)

Here in (4): $e(t)$ is the control error, $u(t)$ is the controller output; $w_1$, $w_2$ and $w_3$ is the weight coefficient.

3.3 PSO algorithm flow

The parameters to be identified in the objective function are $K_p$, $K_i$, $K_d$:

$$X = (K_p, K_i, K_d) = (x_1, x_2, x_3)$$  \hspace{1cm} (7)

$X$ in formula (7) is a feasible solution to the problem of solving the control parameter described by formula (4). The matrix $P = (X_1', X_2', \ldots, X_n')$ composed of $X$ is the particle swarm in PSO, and each feasible solution $X$ is called particle in particle swarm.

Control parameters based on PSO are solved as follows. Firstly, the particle swarm consisting of $m$ particles is initialized in Matlab, the initial position matrix $K$ and the initial velocity matrix $Q$ are randomly generated:

$$K = \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ \vdots & \vdots & \vdots \\ x_{m1} & x_{m2} & x_{m3} \end{bmatrix} \quad Q = \begin{bmatrix} v_{11} & v_{12} & v_{13} \\ v_{21} & v_{22} & v_{23} \\ \vdots & \vdots & \vdots \\ v_{m1} & v_{m2} & v_{m3} \end{bmatrix}$$

Then, the particles in the particle swarm are assigned to $x_1$, $x_2$, $x_3$, and the parameter transfer process is completed. The PID control system model in Simulink environment is simulated and the corresponding performance index of the group parameters is obtained.

Finally, according to equation (5) to determine whether the fitness value of the particles content the set conditions, if not content the set conditions using (4) for speed and location updates, and then the next iteration. The iteration terminates when the maximum number of iterations is reached or the objective function is less than the set threshold. The calculation flow based on PSO is shown in Fig. 4.
4. SYSTEM SIMULATION

The parameters of PSO are as follows: the maximum number of iterations is \( N = 200 \), the size of the particle swarm is \( m = 50 \), the parameters to be identified are \( K_p, K_i, K_d \), so the search space dimension \( D = 3 \); \( c_1 \) and \( c_2 \) is a non-negative constant, usually 1.49\(^\text{[14]}\). The inertia weight \( w \) is a very important parameter to influence the search ability and search speed. To solve this problem, the adaptive adjustment method can be used to calculate \( w \):

\[
w = w_{\text{max}} - (w_{\text{max}} - w_{\text{min}}) \frac{n}{N}
\]

(8)

Where \( n \) and \( N \) represent the number of iterations and the maximum number of iterations, \( w_{\text{min}} \) and \( w_{\text{max}} \) represent the minimum and maximum values of inertia weight. In this paper, \( w_{\text{min}} = 0.4 \), \( w_{\text{max}} = 0.9 \).

According to the above parameters and the calculation flow in Fig.4, the PID parameter optimization program based on PSO is programmed by Matlab / Simulink. The curves of \( K_p, K_i, K_d \) and the objective function \( F \) with iteration times are shown in Fig.5. The final result is: \( K_p=16.83253 \), \( K_i=0.01594 \), \( K_d=1.43779 \), \( F=6.34386 \).

![Optimization curve of \( K_p \)](image1)

(a) Optimization curve of \( K_p \)

![Optimization curve of \( K_i \)](image2)

(b) Optimization curve of \( K_i \)

![Optimization curve of \( K_d \)](image3)

(c) Optimization curve of \( K_d \)

![Optimization curve of \( F \)](image4)

(d) Optimization curve of \( F \)

Fig.5 Optimization process of \( K_p, K_i, K_d, F \)

According to the above parameters, the target temperature control simulation is carried out, and the target temperature is set to 34.7°C. Using the attenuation curve method and PSO tuning PID control step response is shown in Fig. 6.
Note: In Fig.6, curve 1 is the PID control step response curve, which is based on the attenuation curve method. Curve 2 is the PID control step response curve, which is based on the PSO.

It can be seen from Fig. 6 that the PID parameters set by the attenuation curve method can achieve a maximum overshoot of 16.7% in the control process, and a long time to reach the steady state, approximately 10 hours. With the PID parameters obtained by PSO tuning, the maximum overshoot in the control process is approximately 1.4%, and the steady-state time is approximately 3.7 hours.

5. CONCLUSION
According to the requirement of the temperature control of the satellite heat pipe in the thermal vacuum test, the PID parameters of the heat pipe temperature control are optimized by PSO. The simulation result shows that the overall performance of the PID controller optimized by PSO is improved obviously, the overshoot of the system is reduced obviously, the adjustment time is shortened and the dynamic transient performance of the system is greatly improved. Therefore, it is feasible and effective to apply the PSO to the optimization of the heat pipe temperature control parameters.

References
[1] Hou Zengqi, Hu Jingang. Spacecraft thermal control technology - principles and applications [M]. Beijing: China Science and Technology Press, 2007:443-448.
[2] M. O. Efe. Neural network assisted computationally simple PID control of a UAV [J]. Industrial Informatics, IEEE Transactions on, 2011, 7(2): 354-361.
[3] P. P. Joel, P. P. Jose. Trajectory tracking error using PID control law for two-link robot manipulator via adaptive neural networks[J]. Procedia Technology, 2012, 3(2): 139-146.
[4] Lv Bo. Wu Yun-jie. Servo System Control Based on Fuzzy Adaptive PID Method [J]. Journal of System Simulation, 2009,21(21): 6794-6797.
[5] Y. S. Qi, Q. J. Meng. The application of fuzzy PID control in pitch wind turbine [J]. Energy Procedia,2012, 16(c): 1635-1641.
[6] M. K. Hassan, N. A. Azubir. Optimal design of electric power assisted steering system using GA-PID method[J]. Procedia Engineering, 2012, 41(2): 614-621.
[7] ZHAN Zicai. Design of Intelligent PID Controller Based on Genetic Algorithm[J].Journal of East China Jiaotong University, 2009,26(1): 58-61.
[8] Kennedy J, Eberhart R C. Particle Swarm Optimization [C]. Proceedings of the IEEE International Conference on Neural Networks, 1995 :1942-1948.
[9] Wang Y, Li B, Weise T, et al. Self-adaptive learning based particle swarm optimization[J]. Information Sciences,2011, 181(20):4515-4538.
[10] Lin Q, Loxton R, Teo K L. The control parameterization method for nonlinear optimal control: a Survey[J]. Journal of Industrial and Management Optimization. 2014,10(1):275-309.
[11] Gutierrez A L. Comparison of different PSO initialization techniques for high dimensional search space problems: a test with FSS and antenna arrays[C]. Proceedings of the 5th European Conference on Antennas and Propagation (EUCAP), 2011:965 -969.

[12] Chatterjee A, Siarry P. Nonlinear inertia weight variation for dynamic adaptation in particle swarm optimization[J]. Computers and Operations Research, 2006,33(3): 859-871.

[13] Catalao J, Pousinho H. Hybrid wavelet-PSO-ANFIS approach for short-term electricity prices forecasting [J]. Transactions on Power Systems, 2011,26(1): 137 -144.

[14] Liang J J, Qin A K, Suganthan P N, et a1. Comprehensive learning particle swarm optimizer for global optimization of multimodal functions[J]. IEEE Transactions on Evolutionary Computation, 2006,10(3): 28 1-295.