Study on Model Identification for Predictive Control of Boiler Superheated Steam Temperature Model based on Improved Gray Wolf Optimizer

Jianjun Zhu, Binghong Wei

School of information and Control Engineering, Jilin Institute of Chemical Technology, Jilin 132022, China

Abstract

For the boiler superheated steam temperature control system there is a large inertia, large time lag caused by the traditional control method of control accuracy is not high, put forward a fusion of multi-strategy to improve the Gray wolf optimizer to identify the temperature model of the model predictive controller, MATLAB simulation verification, to build a semi-physical simulation platform, through the OPC communication connection between MATLAB and PCS7, and then on the SMPT-1000 Boiler superheated steam temperature control system for experimental verification, the experimental results show that the fusion of multi-strategy to improve the Gray wolf optimizer to identify the temperature model model than other algorithms accuracy method is higher, faster, and the model predictive control of the control indexes are better than the PID control.

Keywords

Temperature; Gray Wolf Optimizer; Particle Swarm Algorithm; Model Identification; Model Predictive Control.

1. Introduction

Boiler is a kind of equipment that can heat water into steam or hot water. It is widely used in various industries, and in some industrial production requires a specific steam temperature. By reasonably controlling the superheated steam temperature, it can ensure that the steam temperature output from the boiler is stabilized within a certain range, avoiding the waste of energy caused by too low a temperature, and the loss of energy caused by too high a temperature, which will have a serious impact on the boiler's life and safety. Nowadays, scholars at home and abroad for regulating the boiler superheated steam temperature control methods have been studied and achieved certain control effects. The more common treatment is to introduce some advanced optimization control methods combined with traditional PID control to form an improved PID control method. Literature [1] Samir Ladjouzi et al. proposed the use of a single memory neuron to find the optimal parameters of the PID controller, and the effectiveness of this method in terms of the change of the set point, good tracking accuracy, and overshooting cancellation has been verified. Literature [2] Wu Tingyun et al. By adding fuzzy logic to the PID controller, the parameters of the PID controller can be adjusted in real time to make the system more robust.

However, the system is often due to large inertia, hysteresis, time-varying uncertainty, external uncertainty brought about by the disturbing factors such as the traditional PID control is difficult to obtain satisfactory control results, so the selection of advanced control strategies for its control is the current hot spot of research [3-4]. Literature [5] Shymaa Darwish used a combination of bimodal model predictive control- mpc (optimal model predictive control- ompc) and generalized predictive

ISSN: 2414-1895 DOI: 10.6919/ICJE.202412_10(12).0017

control (GPC) algorithms, which is highly adaptive and has high control accuracy and improves the efficiency and response of the system. Literature [6] Song Dongran et al. Song Dongran et al. proposed a nonlinear model predictive control method for maximum wind energy extraction from wind turbines based on the yin-yang gray wolf optimization algorithm, efficient convergence and global optimization are achieved. At present, how to identify and obtain the accurate mathematical model of the superheated steam temperature system has become the focus of current research[7].

2. Boiler Superheated Steam Temperature Control System Overview

In this paper, the boiler unit of the SMPT-1000 system is taken as the controlled object, and its process flow diagram is shown in Fig. 1. The saturated steam separated from the steam bag V1102 enters the furnace chamber F1101 for vapor-phase heating, heated to superheated steam. The superheated steam out of the furnace chamber into the desuperheater E1101, for temperature fine-tuning, and finally to the process required superheated steam pressure, superheated steam temperature delivery to the next production unit (such as evaporator).

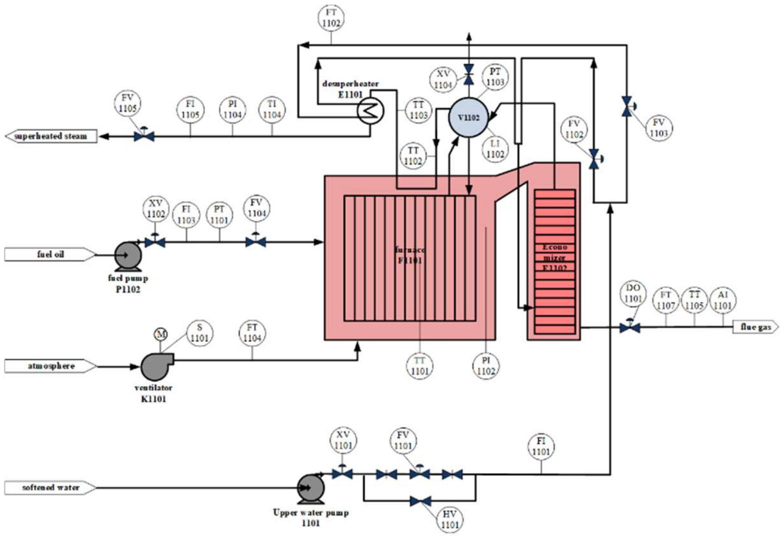

Fig. 1 Sketch of boiler control system

The boiler superheated steam temperature control system is a large inertia and time lag system, the passive object is the superheated steam temperature, and the control variable is the opening of the desuperheated water valve, through reviewing the literature, the superheated steam temperature control system model can be approximated as a second-order inertia plus hysteresis link, as shown in equation (1).

$$G(S) = \frac{K}{(T_1 S+1)(T_2 S+1)} e^{-\tau s} \tag{1}$$

3. Improved Gray Wolf Optimizer for Identifying Predictive Models for Model Predictive Controllers

3.1 Model Predictive Control

Model predictive control is an intelligent control algorithm widely used in industrial process control, mainly including dynamic matrix control (DMC), model algorithmic control (MAC), generalized predictive control (GPC) and generalized predictive pole configuration control (GPP), this paper adopts the dynamic matrix control, but no matter which control method has three elements: predictive model, rolling optimization and feedback correction[8]. The schematic diagram of model predictive control is shown in Fig. 2.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202412_10(12).0017

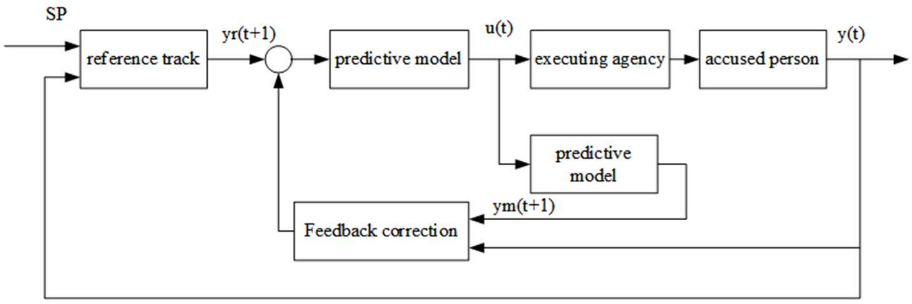

Fig. 2 Schematic diagram of model predictive control

3.1.1 Predictive Model

Starting from the step response of the controlled object, the value $a_i = a(iT), i = 1,2,3,\cdots N$, at the sampling moment, where T is the sampling period, the superheated steam temperature tends to stabilize after a certain moment $t = NT$, and can be regarded as the steady-state value of a_N when $t \to \infty$, and at this time, the dynamic information of the system can be approximately described by $a = \{a_1, a_2, a_2, \dots, a_N\}$, the vector a is called the model vector, and N is the modeling time domain. a to be described, the vector a is called the model vector, and N is the modeling time domain.

If the input $u(k-i)$ is input at some $k+1(k \geq i)$ moment, $\Delta u(k-i)$ acts on the output $y(k)$ as shown in equation (2).

$$y(k) = \begin{cases} a_i \Delta u(k-i), \ 1 \ll i < N \\ a_N \Delta u(k-i), \ i \gg N \end{cases} \tag{2}$$

The $p(p < N)$ step prediction of $y(k+j)$ is obtained according to the superposition principle as shown in equation (3).

$$\hat{y}(k+j) = \sum_{i=1}^{N-1} a_i \Delta u(k+j-i) + a_N \Delta u(k+j-N), (j = 1,2,\cdots,p) \tag{3}$$

The past control inputs are known and can be separated from the past-to-future outputs in the above equation as shown in equation (4).

$$\hat{y}(k+j) = \sum_{i=1}^{j} a_i \Delta u(k+j-i) + \sum_{i=j+1}^{N-1} a_i \Delta u(k+j-i) + a_N \Delta u(k+j-N), (j = 1,2,\cdots,p) \tag{4}$$

The last two terms at the right end of the above equation are the past p-step prediction of the output, denoted $y_0(k+i)$. To increase the dynamic stability of the system and the realizability of the control inputs, as well as to reduce the amount of computation, the vector consisting of Δu is reduced to $m(m < p)$ dimensions. The equation is written in matrix form as shown in equation (5).

$$\hat{Y} = A\Delta U + Y_0 \tag{5}$$

3.1.2 Rolling Optimization

The incremental temperature control equation of this algorithm is obtained by optimizing the minimum performance index and the optimization criterion is shown in equation (6).

$$J = \sum_{j=1}^{p} [y(k+j) - w(k+j)]^2 + \sum_{j=1}^{m} \lambda [\Delta u(k+j-1)]^2 \tag{6}$$

ISSN: 2414-1895 | DOI: 10.6919/ICJE.202412_10(12).0017

Where, λ is the control weighting coefficient, such that $W = [w(k+1), w(k+2), \cdots, w(k+p)]^T$, replace Y with the optimal predicted value of $\hat{Y}$and let $\frac{\partial J}{\partial \Delta U} = 0$,to obtain the control increment of the algorithm in the form of open-loop control as shown in equation (7).

$$\Delta U = (A^T A + \lambda I)^{-1} A^T (W - Y_0) \tag{7}$$

The open-loop form of control does not track the desired value closely, so a closed-loop form of control is used, i.e., the first of the m calculated control increments is put into effect, as shown in equation (8).

$$\Delta u(k) = c^T (A^T A + \lambda I)^{-1} A^T (W - Y_0) = d^T (W - Y) \tag{8}$$

Where, $c^T = [1\ 0\ \cdots\ 0]$, $d^T = c^T (A^T A + \lambda I)^{-1} A^T$.

3.1.3 Feedback correction

Due to the uncertainty of the object and the environment, after the implementation of the control action at the moment k, the actual output $y(k+1)$ and the predicted output at the moment $k+1$ are not necessarily equal, which is required to constitute the prediction error as shown in equation (9).

$$e(k+1) = y(k+1) - \hat{y}(k+1) \tag{9}$$

The prediction for other future moments is corrected with this error weighted as shown in equation (10).

$$\tilde{Y} = \hat{Y} + he(k+1) \tag{10}$$

Where, $\tilde{Y} = [\tilde{y}(k+1), \tilde{y}(k+2), \cdots, \tilde{y}(k+N)]^T$ is the output of the system at the moment $t = (k+1)T$ predicted by the feedback correction, and $h = [h_1, h_2, \cdots, h_N]^T$ is called the error correction vector.

The corrected $\tilde{Y}$ is used as the predicted initial value of the next moment, and since the predicted initial value at the moment of $t = (k+1)T$ is used to predict the output value at the moment of $t = (k+2)T, \cdots, (k+1+\mathrm{N})T$, it is made that $y_0(k+1) = \tilde{y}(k+\mathrm{i}+1)$ $(i = 1,2, \cdots, N-1)$. The predicted initial value of the next moment is obtained as shown in equation (11).

$$y_0(k+\mathrm{i}) = \tilde{y}(k+\mathrm{i}+1) + h_{\mathrm{i}+1} e(k+1) \tag{11}$$

3.2 Gray Wolf Optimizer

The Gray wolf optimizer simulates the leadership hierarchy and hunting system of gray wolves in nature, which has a pyramid-like hierarchy [9-10]. The hunting process of gray wolves is divided into four tier populations:

Tier 1: α wolves. Responsible for leading the whole wolf pack to hunt the prey, the optimal solution in the optimization algorithm;

Tier 2: β wolves. Responsible for assisting the α wolf pack, the sub-optimal solution in the optimization algorithm;

Tier 3: δ wolves. Follows the orders and decisions of α and β, and is responsible for scouting, sentrying, Poorly adapted α and β are demoted to δ;

Tier 4: ω wolves. Follows the actions of the three tier wolves above.

The position of any gray wolf in the pack can be represented by $X_i = \left[x_i^1, x_i^2, \cdots, x_i^d\right]$, where $i = 1,2,\cdots,n$, denotes the number of the gray wolf in the pack, n denotes the size of the gray wolf population, and d denotes the dimension of the space where the gray wolf is located. The process of positional movement of the gray wolf pack during hunting can be expressed by the following equation:

$$\begin{cases} D_{i\alpha}(t) = |C_1(t)X_\alpha(t) - X_i(t)| \\ D_{i\beta}(t) = |C_2(t)X_\beta(t) - X_i(t)| \\ D_{i\delta}(t) = |C_3(t)X_\delta(t) - X_i(t)| \end{cases} \begin{cases} X_{i\alpha}(t) = X_\alpha(t) - A_1(t)D_{i\alpha}(t) \\ X_{i\beta}(t) = X_\beta(t) - A_2(t)D_{i\beta}(t) \\ X_{i\delta}(t) = X_\delta(t) - A_3(t)D_{i\delta}(t) \end{cases} \tag{12}$$

$$X_i(t+1) = \frac{X_{i\alpha}(t) + X_{i\beta}(t) + X_{i\delta}(t)}{3} \tag{13}$$

Where, $X_\alpha(t)$, $X_\beta(t)$, $X_\delta(t)$, represent the position of the three elite wolves in the current iteration; $D_{i\alpha}(t), D_{i\beta}(t), D_{i\delta}(t)$ represent the random positive distance vector between the gray Wolf numbered i and the three elite wolves in the current iteration, that is, the absolute value of the random distance between any Wolf and the elite Wolf in each dimension; $X_{i\alpha}(t)$, $X_{i\beta}(t)$, $X_{i\delta}(t)$, represent the updated positions of the gray wolves numbered i under the guidance of the three elite wolves respectively in the current iteration; $X_i(t+1)$ represents the position that the gray Wolf numbered i will reach at the next iteration time. The random vectors A and C can be obtained as follows:

$$\begin{cases} A_k = a(t)(2r_1 - 1) \\ C_k = 2r_2 \end{cases} \tag{14}$$

Where, A_k, C_k denote the random vectors A and C corresponding to the random variables of the kth level elite wolves, respectively, and r_1, r_2 denote the random numbers of kth level elite wolves in the jth dimensional position taking the values of 0 to 1, respectively. a is a convergence factor that decreases linearly with the increase of iteration, $a(t) = 2 - 2\frac{t}{t_{max}}$, with t denoting the current number of iterations; and t_{max} the maximum number of iterations.

3.3 Particle Swarm Algorithm

The basic particle swarm algorithm is an evolutionary computational algorithm based on group intelligence proposed by James Kennedy and Russell Eberhart in 1995 when simulating the migration and flocking behavior of bird flocks during foraging [11]. Assuming that there are M particles in the D-dimensional search space, $X_i = (X_{i1}, X_{i2}, \dots, X_{iD})$ and $v_i = (v_{i1}, v_{i2}, \dots, v_{iD})$ denote the iterative change position and velocity of the ith particle, respectively. The individual optimal solution $Pbest_i = (P_{i1}, P_{i2}, \dots, P_{iD})$ and the population optimal solution $Gbest_i = (G_1, G_2, \dots, G_i)$ are obtained after the particles pass through the D iterations, and the computational formulas of the conventional particle swarm algorithm are shown in the following equations:

$$v_{i+1} = v_i + c_1 * r_1[Pbest_i - X_i] + c_2 * r_2[Gbest_i - X_i] \tag{15}$$

$$X_{i+1} = X_i + v_{i+1} \tag{16}$$

Where, v_i is the velocity of the ith particle, c_1, c_2 are the individual learning factor and social learning factor, respectively, X_i is the position of the ith particle, $Pbest$ denotes the current particle optimal value, $Gbest$ denotes the population optimal value, and r_1 and r_2 are (0,1) random numbers.

3.4 Improving the Gray Wolf Optimizer

3.4.1 Chaotic Initialization Populations

The positional distribution of the initial gray wolf population in the solution space of the standard GWO algorithm is randomly distributed, and the positional layout of the initial gray wolf population in the solution space has a key influence on finding the optimal path, so the standard GWO algorithm is prone to uneven distribution, resulting in a reduction in the convergence speed. If the initial population can be evenly distributed in the solution space, the accuracy and speed of the algorithm to find the optimal solution will be improved.

In this paper, chaotic mapping is introduced to initialize the location distribution of the gray wolf population, however, a single chaotic mapping often leads to uneven distribution in the solution space due to its own shortcomings, Tent chaotic mapping has the characteristics of fast iteration speed and strong autocorrelation, and logistic chaotic mapping has the characteristics of complex dynamics, thus this paper integrates tent chaotic mapping and logistic chaotic mapping. So that the system can generate more and uniform random sequences in a short time compared with a single chaotic system [12]. Its mathematical formula is:

$$X_{i+1} = \begin{cases} (rX_i(1-X_i) + (4-r)X_i)/2mod1 & X_i < 0.5 \\ (rX_i(1-X_i) + (4-r)(1-X_i)\,)/2mod1 & X_i > 0.5 \end{cases} \tag{17}$$

Where, X_{i+1} is the newly generated conforming chaotic sequence, $X_i \in [0,1]$, r is a control parameter, $r \in [0,4]$, and mod is a modulo operation.

3.4.2 PSO Algorithm Combined with GWO Algorithm

The PSO algorithm has better global exploitation ability and the GWO algorithm has better local mining ability, this paper tries to combine the PSO algorithm with the GWO algorithm in order to improve the algorithm's global exploitation ability and local mining ability, and replaces the gray wolf's position updating formula by the particle position updating formula, so that the gray wolf has the ability to memorize the optimal position in the process of its own evolution. Its velocity and position update formula is:

$$v_{i+1} = \omega\left[v_i + c_1 * r_1(X_{i\alpha}(t) - X_i) + c_2 * r_2\left(X_{i\beta}(t) - X_i\right) + c_3 * r_3(X_{i\delta}(t) - X_i)\right] \tag{18}$$

$$X_{i+1} = X_i + v_{i+1} \tag{19}$$

Where, r_1, r_2, r_3 are uniformly distributed random variables in the interval [0,1], c_1, c_2, c_3 are learning factors with the value of 0.5, and ω is the inertia weight with the value range [0.4,0.9].

3.4.3 Improvement Strategies for Nonlinear Convergence Factor $\boldsymbol{a}$

The search of GWO algorithm is divided into two processes, search exploration process: when the gray wolves just start to hunt, i.e., when the convergence factor satisfies $|a|$>1, at this time, the random coefficient is in the state of $|A|$>1 with high probability, and the wolves will expand the search range, and the population will be more inclined to the global exploration; convergence process: when the convergence factor increases with the number of iterations to satisfy $|a|\leqslant 1$, at this time, the random coefficient must satisfy $|A|\leq 1$, the position of the gray wolf must be close to the prey, the wolf pack

ISSN: 2414-1895 DOI: 10.6919/ICJE.202412_10(12).0017

will narrow the search range, and the population is more inclined to local exploitation. Therefore, the conversion of global exploration and local exploitation ability of the GWO algorithm depends on the change of the control parameter a. The value of the convergence factor a decreases linearly with the number of iterations from 2 to 0. In this paper, we set the nonlinear convergence factor of the sigmoid function [13], which is given by:

$$a(t) = 2 - \frac{2}{1 + e^{-1 * (\frac{t}{t_{max}} - \frac{1}{2})}} \tag{20}$$

3.4.4 Improvement Strategies for Inertia Weights ω

The inertia weight ω of the PSO algorithm is set as a constant, which may lead to the PSO algorithm in the later stages of the optimization will not be able to search for the optimal solution and other problems, where the inertia weight of the typical first decreasing strategy is introduced, so that it decreases with the increase in the number of iterations from 0.9 to 0.4, which is given by the formula:

$$\omega = 0.4 + 0.5 \frac{t_{max} - t}{t_{max}} \tag{21}$$

3.5 Improving the Gray Wolf Optimizer Process

Need to establish a measure of the accuracy of superheated steam temperature model of the fitness function, commonly used error integration criterion include: squared error integration criterion (ISE), absolute error integration criterion (IAE), time multiplied by the squared error integration criterion (ITSE), time multiplied by the absolute error integration criterion (ITAE), etc., this paper uses the ISE criterion as a recognition of the model parameters of the fitness function. It is shown in the following equation:

$$J_{ISE} = \int_0^{\infty} [y(t) - \hat{y}(t)]^2 dt \tag{22}$$

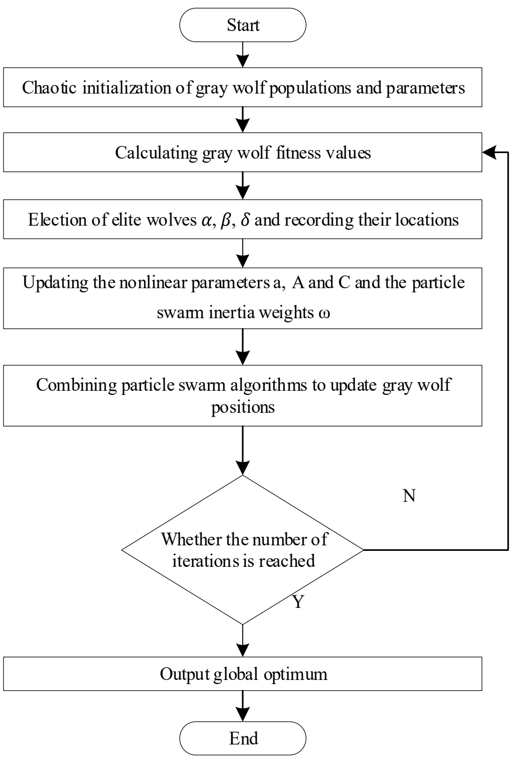

Fig. 3 Optimization flowchart of improved Gray wolf optimizer

ISSN: 2414-1895 DOI: 10.6919/ICJE.202412_10(12).0017

Where, J_{ISE} is the adaptation value; $y(t)$ denotes the true value at time t; and $\hat{y}(t)$ denotes the predicted value of the model at time t. The smaller the value of this criterion, the closer the predicted value of the model is to the actual observed value and the better the performance of the model. The specific steps to improve the Gray wolf optimizer to identify the model parameters are shown below.

4. Experimental Results and Analysis

4.1 Data Acquisition

The operating data of superheated steam temperature is obtained from the boiler unit of Siemens SMPT-1000 equipment with a superheated steam flow load of 16Kg/s, a sampling interval of 1s, and a sampling time of 394s.

4.2 Analysis of Experimental Results

In order to verify the effectiveness of the algorithm proposed in this paper, the program is written in MATLAB environment, and compared with the conventional Gray wolf optimizer, particle swarm algorithm, and the model accuracy identified by the two-point method, and the squared error integration criterion (ISE) is selected as the fitness function, and the iteration curves of the three algorithms are shown in Fig. 4. From Fig. 4, it can be seen that the improved Gray wolf optimizer is better in terms of initial fitness value and convergence speed, and reaches the optimal value after 76 iterations, while the standard Gray wolf optimizer and particle swarm algorithm need 95 and 97 iterations respectively to reach the optimal; the standard GWO algorithm has a weaker ability to develop in the later stage, and is prone to fall into the local optimum; the particle swarm algorithm has insufficient pre-development ability, and a weaker ability to develop in the later stage.

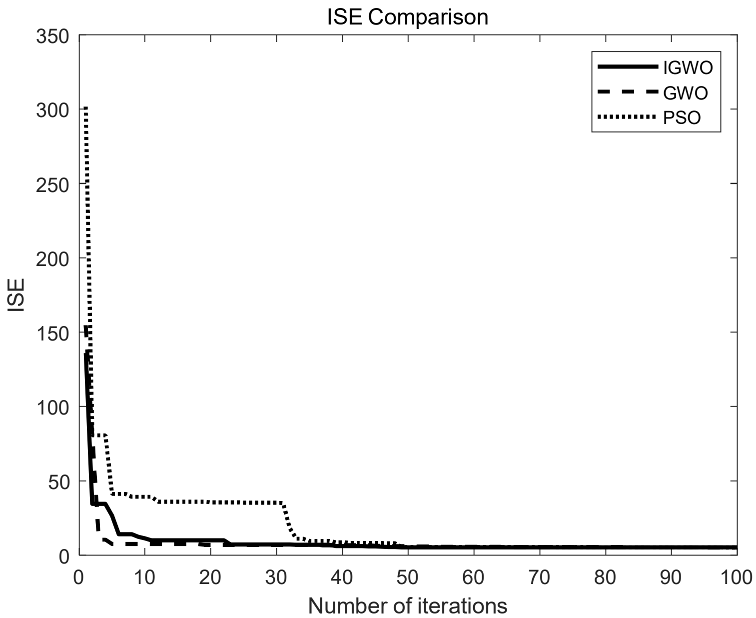

Fig. 4 Iteration curves of three algorithms for transfer function optimization

In order to reduce the randomness of the algorithms, multiple sets of discrimination experiments were conducted using each algorithm and 10 sets of data were recorded randomly and the five sets of data with the smallest values of the fitness function were averaged, as shown in Table 1.

From Table 1, it can be seen that the average value of the fitness function of the improved Gray wolf optimizer is significantly lower than that of the other three algorithms, which proves its better fit to the original data.

Table 1. Results of four algorithms for optimization of transfer function

arithmetic	transfer function model	ISE
IGWO	$G(S)=\frac{2.3674}{4.3535s^2+11.8861s+1}e^{-4.941s}$	5.2108
GWO	$G(S)=\frac{2.3674}{4.1086s^2+11.8147s+1}e^{-4.9503s}$	5.2123
PSO	$G(S)=\frac{2.3659}{5.005s^2+11.7037s+1}e^{-5.2869s}$	5.2294
two-pointed approach	$G(S)=\frac{2.35}{40.435s^2+14.815s+1}e^{-4s}$	230.1177

In order to better verify the superiority of the improved Gray wolf optimizer, the models identified by each algorithm are applied to the superheated steam temperature control system of the SMPT-1000 boiler unit respectively for comparison. According to the control scheme and control requirements of the boiler, the CFC and SFC programs of PCS7 are written. The OPC communication between MATLAB and PCS7 is realized through KEPServer software, and MATLAB is used as the controller of the boiler superheated steam temperature control system, which in turn controls the superheated steam temperature. The experimental results are shown in Fig. 5, and its various characteristic indexes are shown in Table 2. A conforming interference is added in the operation to 1000s to increase the superheated steam flow rate from 16kg/s to 18kg/s, see Fig. 6.

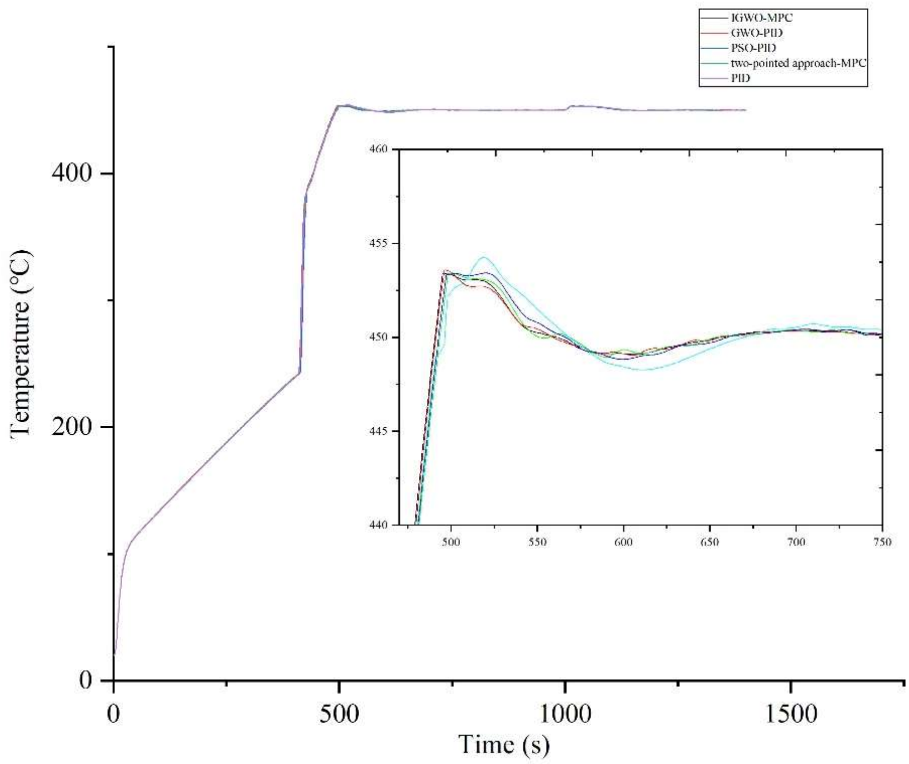

Fig. 5 Each algorithm in the SMPT-1000 system operation curve

ISSN: 2414-1895 DOI: 10.6919/ICJE.202412_10(12).0017

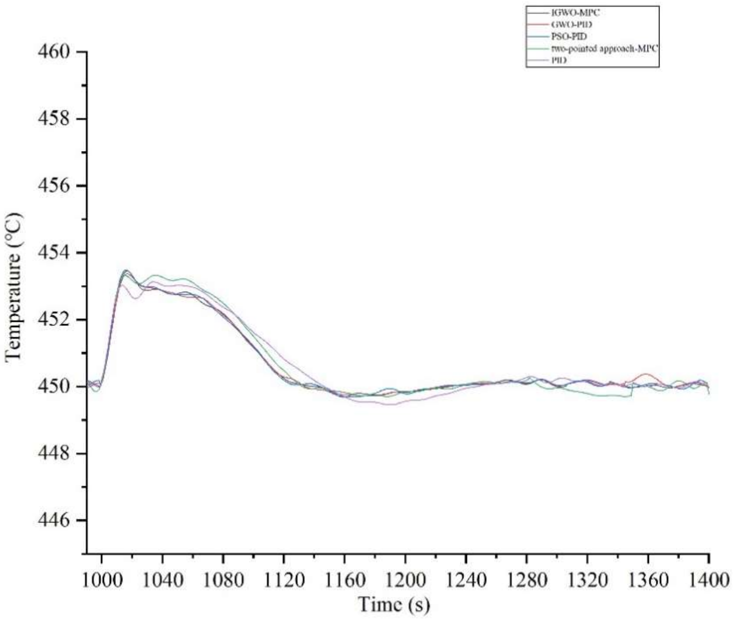

Fig. 6 Interference section curve

Table 2. Indicators of the characteristics of each algorithm

Numble	Peak/°C	Peak time/s	Amount of overshooting/%	Adjustment time/s
IGWO-MPC	453.384	495	16.92%	632.2
GWO-MPC	453.579	496	17.89%	633.2
PSO-MPC	453.4	501	17%	639
two-pointed approach-MPC	453.404	502	17.02%	635.1
PID	454.252	519	21.26%	656

As can be seen from Fig. 6 and Table 2, the peak value, peak time, overshooting amount and regulation time of the model predictive control have been optimized to a greater extent compared with the PID control, which proves that the model predictive controller applied to the boiler superheated steam temperature system control possesses application value. In the model predictive control, the application of the improved Gray wolf optimizer to identify the model of the control indexes are better than the other three algorithms, in the increase of the superheated steam outlet flow load interference, the model predictive control algorithm will regulate the temperature to return to the set value faster, the algorithm's robustness is better, but the peak value of the model predictive control will be slightly higher than the PID control.

5. Summary

Aiming at the boiler superheated steam temperature control system, which is characterized by large inertia and large time lag that cause the control accuracy of traditional control methods to be low, and the standard Gray wolf optimizer has low accuracy in model recognition, a scheme is proposed to improve the Gray wolf optimizer to recognize the boiler superheated steam temperature model by fusing multi-strategy, in the improvement of the Gray wolf optimizer, the Chaos mapping is added in the initialized population, and the Particle Swarm algorithm is fused to replace the Gray Wolf's position updating formula by the particle position updating formula. In the improved Gray wolf optimizer, chaotic mapping is added to the initialized population, the particle swarm algorithm is integrated, the particle position update formula is replaced by the gray wolf position update formula, and the updating strategies of convergence factor and inertia weight are improved. Through the experimental validation of the boiler superheated steam temperature control system, it is proved that the accuracy and speed of this method of model identification is greatly improved compared with the standard Gray wolf optimizer, particle swarm algorithm and two-point method, reducing the situation of falling into the local optimum, and providing a new method and idea for the optimization of the controlled object model in the process control industry.

Acknowledgments

Jilin province science and technology development item(YDZJ202201ZYTS555).

References

[1] Ladjouzi S, Grouni S. PID controller parameters adjustment using a single memory neuron. Journal of the Franklin Institute. 2020, Vol. 357(No. 9), p. 5143-5172.

[2] Wu T Y, Jiang Y Z, Su Y Z, et al. Using simplified swarm optimization on multiloop fuzzy PID controller tuning design for flow and temperature control system. Applied Sciences. 2020, Vol. 10(No. 23), p. 8472.

[3] Li Z, Chu M, Liu Z, et al. Furnace heat prediction and control model and its application to large blast furnace. High Temperature Materials and Processes. 2019, Vol. 38(No. 2019), p. 884-891.

[4] Zhan J, Du J, Dong S, et al. Superheated steam temperature system of thermal power control engineering based on neural network local multi-model prediction. Thermal Science. 2021, Vol. 25(No. 4 Part B), p. 2949-2956.

[5] Darwish S, Pertew A, Elhaweet W, et al. Advanced boiler control system for steam power plants using modern control techniques. IEEE 28th International Symposium on Industrial Electronics (ISIE). Vancouver, British Columbia, Canada, 2019.6.12, p. 461-466.

[6] Song D, Liu J, Yang Y, et al. Maximum wind energy extraction of large-scale wind turbines using nonlinear model predictive control via Yin-Yang grey wolf optimization algorithm. Energy. 2021, Vol. 221, p. 119866.

[7] Wang Q, Zhong Y, Zhang L, et al. Model identification of CFB boiler combustion system based on COLGWO algorithm. Control Engineering. 2023, Vol. 30(No. 07):1171-1179.

[8] Shi Y, Zhang Z, Chang H, et al. Pseudo feedforward dynamic matrix control algorithm for desired closed-loop performance in MPC systems. Journal of Process Control. 2023, Vol. 123, p. 37-49.

[9] Salgotra R, Singh U, Sharma S. On the improvement in grey wolf optimization. Neural Computing and Applications. 2020, Vol. 32, p. 3709-3748.

[10] Faris H, Aljarah I, Al-Betar M A, et al. Grey wolf optimizer: a review of recent variants and applications. Neural computing and applications. 2018, Vol. 30, p. 413-435.

[11] Taieb A, Salhi H, Chaari A. Adaptive TS fuzzy MPC based on particle swarm optimization-cuckoo search algorithm. ISA transactions. 2022, Vol. 131, p. 598-609.

[12] Huang Q, Chen HY, Liu Y, et al. Mobile robot path planning based on multi-strategy fusion Gray wolf optimizer. Journal of Air Force Engineering University. 2024, Vol. 25(No. 03), p. 112-120.