

Study on Component Characteristic Correction Algorithm of Turbofan Engine

Shuai LIU (✉ caucliushuai@163.com)

Civil Aviation University of China <https://orcid.org/0000-0002-8966-7041>

WANG Wei

Civil Aviation University of China

BAI Jie

Civil Aviation University of China

DAI ShiJie

HEBUT: Hebei University of Technology

Original Article

Keywords: Turbofan engine, Component characteristic, GA, DE, PSO

Posted Date: February 18th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-195963/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Title page

Study on Component Characteristic Correction Algorithm of Turbofan Engine

LIU Shuai, born in 1990, is currently a lecturer at *Civil Aviation University of China*. He received his doctor degree from *Hebei University of Technology, China*, in 2020. His research interests include Aeroengine Modelling and aeroengine nonlinear control.
Tel: +86-18618141563; E-mail: cacliushuai@163.com

WANG Wei, born in 1977, is currently an associate professor at *Civil Aviation University of China*. He received his doctor degree from Harbin Institute of Technology University, *China*, in 2008.

BAI Jie, born in 1963, is currently is currently a professor at *Civil Aviation University of China*.
E-mail: caucbaijie63@163.com

DAI ShiJie, born in 1966, is currently is currently a professor at *Hebei University of Technology*.

Corresponding author: Bai Jie E-mail: caucbaijie63@163.com

ORIGINAL ARTICLE

Study on Component Characteristic Correction Algorithm of Turbofan Engine

LIU Shuai^{1,2} WANG Wei^{1,2} BAI Jie¹ DAI ShiJie³

(1. Key Laboratory for Civil Airworthiness Certification Technology, Civil Aviation University of China; Tianjin 300300, China;

2. School of Airworthiness, Civil Aviation University of China; Tianjin 300300, China;

3. School of Mechanical Engineering, Hebei University of Technology, Tianjin 300130, China)

Abstract

The optimization of component characteristics is an important part of the research on model modification of turbofan engine. This paper studied three kinds of modification and optimization algorithms: genetic algorithm (GA), differential evolution algorithm (DE) and particle swarm optimization algorithm (PSO), and made a comparative analysis on the principles of the algorithms. In addition, the DGEN380 turbofan engine component model was taken as the research object to test and compare the modification performance of each algorithm. The results show that: compared with GA, DE algorithm and PSO algorithm have higher adaptability to multi-objective and multi-parameter correction, the success rate can reach 100%, the error is kept within 1%, and the convergence ability is strong. Compared with the other two algorithms, the solving speed of PSO algorithm is more influenced by the initial parameters. This conclusion can provide an important reference for selecting and improving the algorithm.

Key words: Turbofan engine; Component characteristic; GA; DE; PSO

1 Introduction

Aeroengine model is an important tool for the development of advanced aerodynamic power plants, and the accuracy of the model determines the scope of application of the model. Due to the theoretical and random errors of the components, the deviation of installation of each component and the variety in the working environment, the actual characteristics of the engine components have changed. In

order to make the output parameters of the engine model more accurate, it is necessary to study a method to further modify the characteristics of the engine components through the engine test data, and to modify the engine model to meet the accuracy requirements and match the actual engine conditions as much as possible.

In the field of aero-engine model correction. Stamatis A¹ proposed a method that can adaptively correct the characteristics of engine components according to the actual measurement data, and use the nonlinear generalized minimum residual method to obtain the guess vector and the correction factor. C Kong² proposed a method of scaling the characteristics map of components based on system identification, which can obtain high-precision simulation results. Later, he used genetic algorithm to solve the compressor component characteristic diagram with higher accuracy, and proposed a method to improve the design of fitness function, which reduced the dependence on experimental data and shortened the calculation time, and improved the correction efficiency^{3,4}. Y G Li⁵ proposed that in order to solve the problem of inaccurate calculation of the model design point, the unmeasurable parameters such as the work efficiency of the components were regarded as unknown quantities, and the Newton-Raphson algorithm was used to solve the correction coefficients, thereby obtaining the accurate true values of the unmeasurable parameters. On this basis, Y G Li⁶ put forward the concept of component characteristic correction factor function. This method can reduce the amount of calculation and is suitable for correction of multiple operating points.

Related research focused on the matching of component characteristics. Choosing an appropriate algorithm is the key to correcting component characteristics. There is no relevant research on how to select component characteristics correction algorithms and the differences in correction effects of each algorithm. Aiming at the problem of the selection of correcting algorithm, this paper takes DGEN380 engine compressor components as the correcting object. The three different types of optimization algorithms are systematically compared and analyzed in theory, and examples and model simulations are used to more objectively describe the effect and difference characteristics of model correction of each algorithm. This paper provides reliable theoretical support for the research on the components characteristics correction algorithm of turbofan engine in the future.

2 Component-level Model

The model proposed in this paper is based on the component method modeling^{7,8} using the DGEN380 engine as the object of modeling. This engine is a dual-rotor, separated exhaust turbofan engine. The core uses a single-stage centrifugal compressor, an annular reversed combustion chamber and an axial-flow high pressure turbine. Low pressure rotor is composed of high bypass ratio fan, gear box (low pressure speed/fan speed=3.32) and axial-flow low pressure turbine.

Each parameter of the model is not independent, but there is a unique solution that simultaneously satisfies the dynamic co-working equation. Therefore, an iterative process is required to find the steady-state operating point of the engine⁹. The dynamic co-working equation can be expressed as the following 4 flow balance equations and

2 power balance equations, as shown in Eq.(1) and Eq.(2).

(1) According to the continuous flow, the flow balance relationship of fan outlet, combustion chamber outlet, high pressure turbine outlet, and low pressure turbine outlet can be obtained.

$$\begin{cases} (W_{comp,in} + W_{bypass}) / W_{fan,out} - 1 = 0 \\ W_{ht,out} / W_{ht,in} - 1 = 0 \\ W_{lt,out} / W_{lt,in} - 1 = 0 \\ W_{comp,out} / W_{comp,in} - 1 = 0 \end{cases} \quad (1)$$

(2) High and low pressure rotor power balance equation

$$\begin{cases} \eta_{ht} \cdot P_{HT} / P_{COMP} - 1 = 0 \\ \eta_{lt} \cdot P_{LT} / P_{FAN} - 1 = 0 \end{cases} \quad (2)$$

In Eq.(2), η_{ht} and η_{lt} are the mechanical efficiency of the high-pressure rotor shaft and the low-pressure rotor shaft, respectively.

High pressure speed n_{ht} , low pressure speed n_{lt} , and fan, compressor, high and low pressure turbine pressure ratio π_f , π_c , π_{ht} , π_{lt} are selected as the initial guess before solving. In particular, the main function of the high and low pressure speeds n_{ht} and n_{lt} is to determine the component characteristic value on the component characteristic map. Taking the compressor component for instance, π_c is known and calculate the corrected spool speed by using n_{ht} , and then use the interpolation method to calculate the flow rate and efficiency under the corrected spool speed and pressure ratio from the component characteristic map. Therefore, the deviation of the component characteristics will cause large errors in model calculation results.

3 Component Characteristic Correction Algorithm

Because the difference in component characteristics will reduce the accuracy of the model, it is necessary to modify the component characteristics to match the actual engine state. The correction problem is essentially a multi-objective parameter optimization problem. At present, genetic algorithm (GA), differential evolution algorithm (DE) and particle swarm optimization algorithm (PSO) are widely used in component characteristics correction. This paper compares and analyzes the three algorithms in component characteristics correction application.

3.1 Component Characteristic Correction

For engines, because it is difficult to obtain accurate component characteristics, general characteristics are usually used in the process of modeling, which will directly affect the accuracy of the model. Therefore, this article uses component characteristic correction factors to correct component characteristics. Taking the compressor component for instance, let the characteristics given by its design point (π , w , η), which represent pressure ratio, flow rate and efficiency respectively. (ΔC_π , ΔC_w , ΔC_η) are selected as the correction factors of the corresponding component characteristic, and are used to calculate the new component characteristic, as shown in Eq.(3).

$$\begin{cases} \Delta C_\eta = \frac{\eta_{real}}{\eta_{sim}} & \eta_{amend} = \Delta C_\eta \cdot \eta_{sim} \\ \Delta C_w = \frac{w_{real}}{w_{sim}} & w_{amend} = \Delta C_w \cdot w_{sim} \\ \Delta C_\pi = \frac{\pi_{real} - 1}{\pi_{sim} - 1} & \pi_{amend} = \Delta C_\pi \cdot (\pi_{sim} - 1) + 1 \end{cases} \quad (3)$$

In the Eq.3, the subscript “sim” is the simulation value, “real” represents the test value, and “amend” represents the new component characteristic value.

Combined with the engine test data, the relative error between it and the output of corrected model is used as the criterion for judging the accuracy of model. The fitness function F is established,

$$F = k \left/ \sum_{i=1}^n w_i \left| \frac{p_{i,sim} - p_{i,real}}{p_{i,real}} \right| \right. \quad (4)$$

Which $k=1$, p_i represents the target parameter, w_i represents the weight coefficient of the above-mentioned target parameter error, and all w_i take 1 here.

The larger fitness function, the higher agreement between the model simulation output data and the experimental measurement data, and the smaller model error.

The specific component characteristic correction process is shown in the Fig.1.

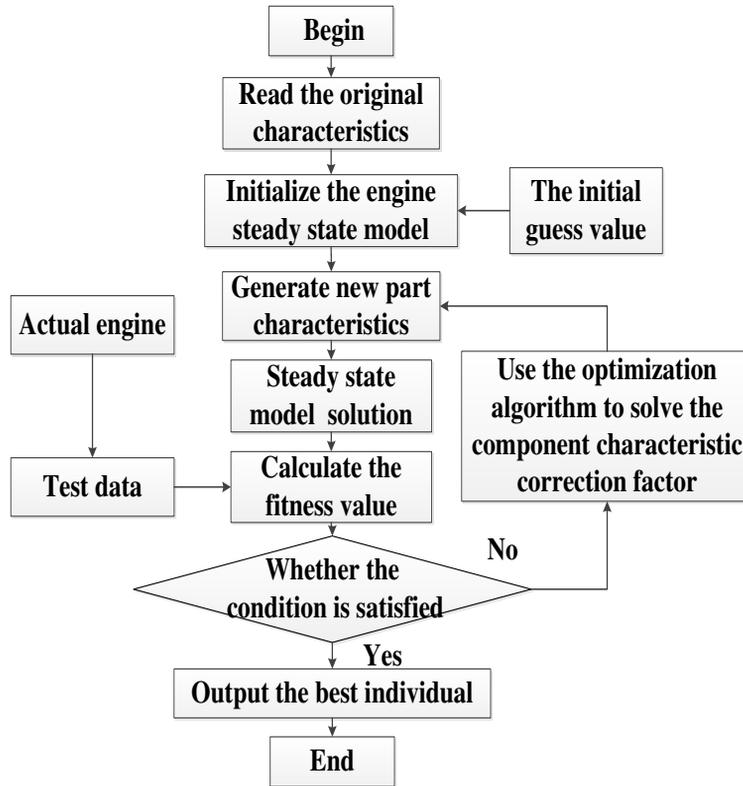


Fig.1 Component feature modification process

Choosing an appropriate algorithm is the key to correct the component characteristic. Therefore, this paper compares and analyzes three commonly used optimization algorithms of model component characteristic.

3.2 Mechanism Analysis of 3 Correction Algorithms

The process of solving the component characteristic correction coefficients by three correction algorithms (GA, DE, PSO) is shown schematically in Fig.2.

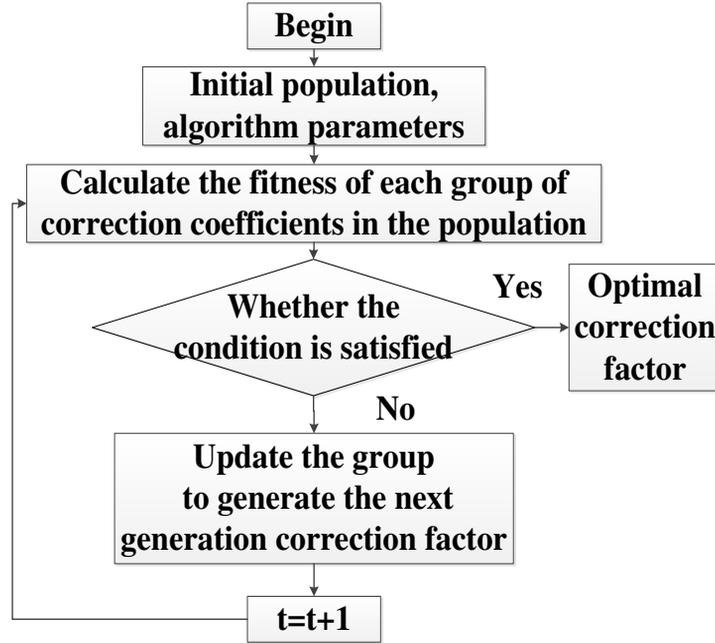


Fig.2 Optimization algorithm solution process

The basic processes of the three algorithms and the establishment of fitness functions are consistent, but there are certain differences in the implementation of specific steps. Compare and describe the differences between the three algorithms in the following two parts.

1. Initialize the population

Genetic algorithm (GA) and differential evolution algorithm (DE) randomly generate N -digit M binary individuals in the solution domain Ω to form the initial population. However, DE forms the initial population in which each individual is a real number and does not need to be binary coded. PSO does not use evolutionary operators for individuals like other evolutionary algorithms, but treats each individual as a particle with no volume and weight in the n -dimensional search space. Each particle will move in the solution space and its direction and distance will be determined by a speed. It is necessary to randomly generate the position P and velocity V of N particles in the solution domain Ω , and set the initial parameter learning factors $C1$, $C2$ and inertia weight ω .

2. Update the population

GA uses selection, crossover, and mutation operations to update the population to generate the next generation. DE makes the population evolve toward a larger population fitness value through three operations of mutation, crossover, and selection. Both algorithms use these three operations to update the population, but their order is different, and the specific implementation of each operation is also different.

Table 1 The difference of each algorithm updates the population

Algo rithm	Update the group		
	Select	Cross	Mutate
GA	<p>This paper adopts the method of "Roulette row-in-the-middle method selection". Firstly, calculate the cumulative fitness value of each individual.</p> $fitnessX_i = \sum_{j=1}^i fitness(X_j) \quad i=1,2,3 \dots N \quad (5)$ <p>Which $fitness(X_j)$ represents the fitness value of each individual.</p> <p>Then use the row-in-the-middle method for random sampling with replacement to select the next generation</p> $X_{new} = \begin{cases} X_i & r < fitnessX_i \\ X_k & fitnessX_{k-1} < r \leq fitnessX_k \end{cases} \quad (6)$ <p>Which r is a random number in the range of $[0, fitnessX_N]$, X_{new} is a new generation population.</p>	<p>In this paper, GA adopts single-point crossover. It means that only one crossover point is randomly set in the individual coding string, and then part of the chromosomes of two paired individuals are exchanged at this point.</p> $X_{i,jM} = \begin{cases} X_{k,jM} & r \leq C_p \\ X_{i,jM} & r > C_p \end{cases} \quad i,k \in 1,2,3 \dots N \quad (7)$ <p>Which r is a random number in the range of $[0,1]$, i and k are random integers in the range of $[0,M]$, j is a random integer in the range of $[0,M]$, C_p is the crossover probability.</p>	<p>In this paper, GA adopts basic bit mutation: the mutation operation is performed on the value of a certain locus randomly designated by the mutation probability in the individual code string, as shown in the following equation,</p> $X_{i,j} = \begin{cases} \overline{X_{i,j}} & r \leq M_p \\ X_{i,j} & r > M_p \end{cases} \quad i=1,2,3 \dots N \quad (8)$ <p>Which r is a random number in the range of $[0,1]$, j is a random integer in the range of $[0, M]$, $X_{i,j}$ is the j-th binary code of the i-th individual in the parent population and M_p is the mutation probability.</p>
DE	<p>DE also selects the next generation according to the individual fitness value, which is different from GA, as shown in the following equation,</p> $X_i = \begin{cases} U_i & fitnessU(i) > fitnessX(i) \\ X_i & fitnessU(i) \leq fitnessX(i) \end{cases} \quad (9)$ <p>$i=1,2,3 \dots N$</p> <p>Which U is the cross vector and X is the parent population.</p>	<p>DE generates the cross vector U_i according to the following equation,</p> $U_i = \begin{cases} V_i & k \leq C_p \\ X_i & k > C_p \end{cases} \quad (10)$ <p>Which k is a random number in the range of $[0,1]$.</p>	<p>DE generates the mutation vector V_i according to the following equation,</p> $V_i = gbest + M_p \cdot (X_{r1} - X_{r2}) \quad i=1,2,3 \dots N$ <p>Which X_{r1}, X_{r2} are 2 different individuals randomly selected in the parent population, and $r1 \neq r2 \neq i$.</p>
PSO	The particle swarm algorithm does not update the next generation through selection,		

crossover and mutation operations, but determines the next generation population by tracking two extreme values of particles. One is the optimal solution $pbest$ found by the particle itself so far, and the other is the optimal solution $gbest$ found so far for the entire population. The specific update method is as follows,

$$\text{Speed update: } V_i = \omega \cdot V_i + c1 \cdot (gbest - X_i) + c2 \cdot (pbest - X_i) \quad (12); \quad \text{Population update: } X_{new} = X + V \quad (13)$$

$$\text{Which, } V_i = \begin{cases} V_{\max} & V_i > V_{\max} \\ V_{\min} & V_i < V_{\min} \end{cases}, \text{ and } V_{\max}, V_{\min} \text{ are the maximum and minimum set speed.}$$

3.3 Example Simulation

Construct the following two nonlinear sets of equations as simulation research objects.

(1) The set of equations 1 (simplified equation of compressor)

This paper takes the compressor components as the correction object. The overall component model of the compressor is integrated and packaged into a mathematical function model. In this model, the total temperature of the gas at the compressor inlet T_{25} , total pressure at the compressor inlet P_{25} , and compressor compression ratio π_c are as input, and the total gas temperature at the compressor outlet T_3 , total pressure at the compressor outlet P_3 , actual compressor flow W_3 and compressor power N_c are as output.

$$[T_3, P_3, W_3, N_c] = f_{comp}(T_{25}, P_{25}, \pi_c) \quad (14)$$

According to the principle of compressor, it can be simplified into the following form,

$$\begin{aligned} C_1 - \frac{C_2}{f_1} &= 0 \\ C_3 - C_4 \cdot f_2 &= 0 \\ C_5 - C_6 \cdot f_1 \cdot f_2 &= 0 \\ &\Downarrow \\ \begin{cases} f_1 = a_1 x_1^2 + b_1 x_2^2 + c_1 x_1 + d_1 x_2 + C_7 = C_2 / C_1 \\ f_2 = a_2 x_1^2 + b_2 x_3^2 + c_2 x_1 + d_2 x_3 + C_7 = C_3 / C_4 \end{cases} \end{aligned} \quad (15)$$

In the equation, C , a , b , c , and d are constant terms, f_1 , f_2 are efficiency and flow interpolation functions, and x_1 , x_2 , and x_3 respectively correspond to the pressure ratio, efficiency and flow coefficient of the component characteristics.

Assignment to the above equations can get the final simplified form of the set of equations 1,

$$\begin{cases} x_1^2 + 2x_2^2 + 3x_1 + 4x_2 + 2 = 15 \\ 2x_1^2 + x_3^2 + 2x_1 + x_3 + 3 = 10 \\ x \in [0, 2] \end{cases} \quad (16)$$

(2) The set of equations 2 (multi-variable and multi-objective, theoretical solution (1,1,1))

$$\begin{cases} x_1^2 + x_2^2 + x_3^2 - 3 = 0 \\ x_1^2 + x_2^2 + x_1 \cdot x_2 + x_1 + x_2 - 5 = 0 \\ x_1 + x_2 + x_3 - 2 = 0 \\ x \in [0, 2] \end{cases} \quad (17)$$

The fitness function is the above two sets of equations. The smaller fitness value, the smaller error. Set the average fitness value to be less than 0.1 to meet the solution requirements. The test results of each algorithm running 20 times are as follows.

Table 2 The solution result of the system of equations 1 for each algorithm

Number	Algorithm	Mean optimal fitness value (error)	Times(error<0.1)	Optimal algebra
1	GA	0.1258	10	30
	DE	0	20	110
	PSO	0	20	236

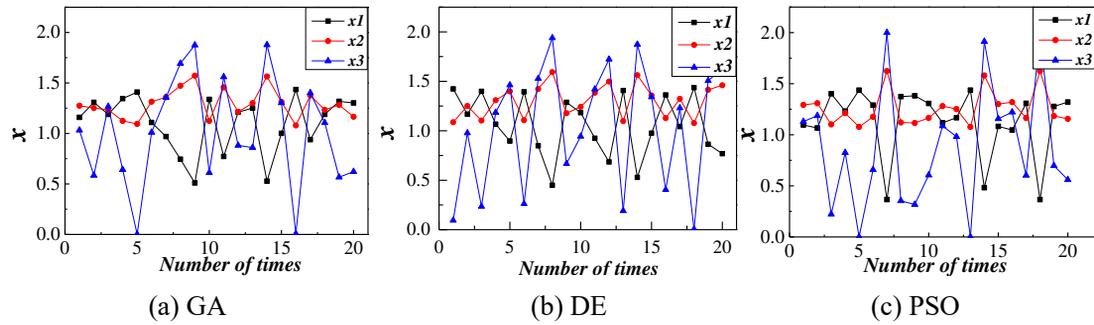


Fig.3 X value results of each algorithm running 20 times

Table 3a The solution result of the system of equations 2 for each algorithm

Number	Algorithm	Mean optimal fitness value (error)	Times(error<0.1)	Optimal algebra
2	GA	0.063	18	28
	DE	0	20	111
	PSO	0	20	7

Table 3b The solution result(X) of the system of equations 2 for each algorithm

Number	Algorithm	X1		X2		X3	
		Average value	Error/10 ⁻⁶	Average value	Error/10 ⁻⁶	Average value	Error/10 ⁻⁶
2	GA	0.98	-20185.7	1.01	11730.21	0.99	-11388.1
	DE	1	-0.00089	1	0.00089	1	0
	PSO	1	0	1	0	1	0

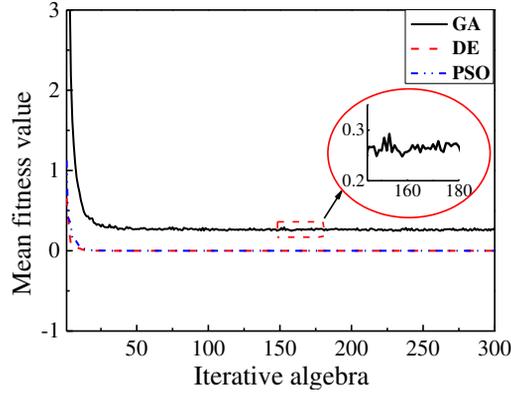


Fig.4 The mean fitness value of the system of equations 1 for each algorithm

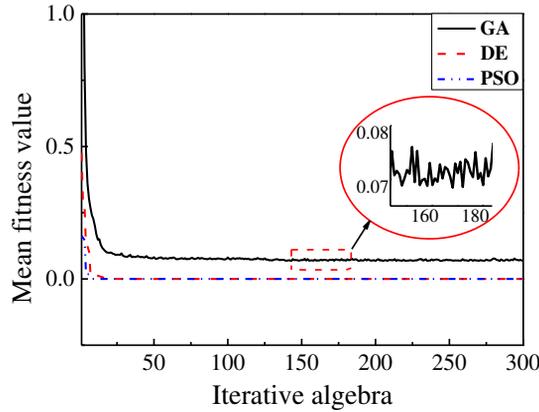


Fig.5 The mean fitness value of the system of equations 2 for each algorithm

The errors of DE and PSO are basically negligible compared to GA algorithm. The DE algorithm has little difference in the solution error and the solution speed of the two sets of equations. The PSO algorithm has a large difference in the algebraic difference of the solution of the two sets of equations. For equation 2 with only a unique solution, the optimal solution can be found within a few generations. However, for the set of equations 1 with multiple solutions, it needs to average more than 200 generations to find the optimal solution.

Combining Fig.4 and Fig.5, the optimal algebra of GA algorithm for solving the two equations is relatively small, but the average fitness value will be stable in a certain range around 50 generations, fluctuate and no longer continue to look for better solution. The optimal solution of DE and PSO algorithm is slower than that of GA algorithm, and its average fitness value can be stabilized in the ideal convergence area within a few generations, and continue to search for better solutions.

According to the simulation results of the above examples, it can be inferred that the traditional GA algorithm for multi-objective multi-parameter solving problems is slower than the DE and PSO algorithms, and the solution accuracy is not high. The PSO algorithm is slower to optimize the problem with multiple solutions.

4 Component Model Simulation Analysis

In order to verify the correction performance of each algorithm, the actual compressor component model is used as the object to compare and analyze the correction effect, and analyze the results from the principle of the algorithm.

4.1 Simulation Results of Compressor Components

DGEN380 engine compressor components is as the correction object. At the design point (maximum continuous climb point) of DGEN380 (height $H=3048\text{m}$, Mach number $Ma=0.338$, fuel flow rate $W_f=0.027782\text{kg/s}$), the optimization algorithm is tested. The section parameters of the compressor components in Table 4 are selected as the target parameters.

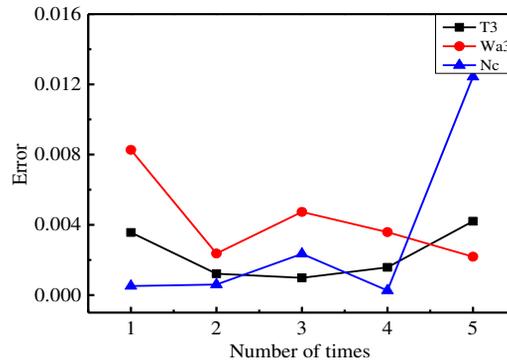
Table 4 Target parameter

Number	Symbol	Implication	Real value
1	T_3	Compressor outlet temperature	490.8726K
2	Wa_3	Compressor flow	1.33647kg/s
3	N_c	Compressor power	269.849kw

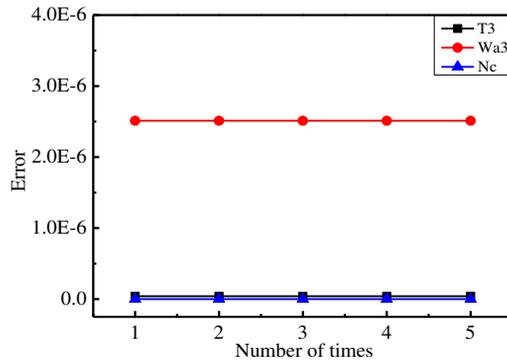
The results of each algorithm running 5 times are as follows.

Table 5 Compressor correction results of each algorithm

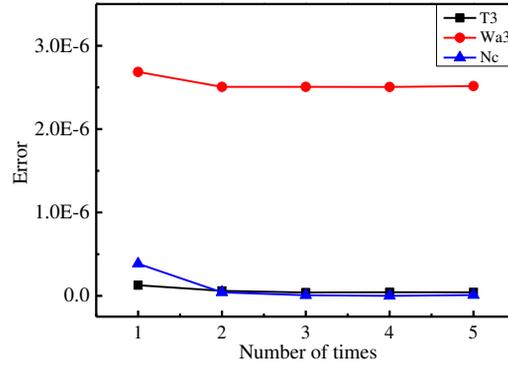
Algorithm	Mean error of target parameters			Mean optimal fitness value	Optimal algebra
	T_3	Wa_3	N_c		
GA	0.00231	0.00423	0.00323	109.7182	68
DE	4.04612e-8	2.51109e-6	7.70927e-8	5.0287e+07	97
PSO	6.20746e-8	2.54437e-6	8.86864e-8	2.2200e+07	98



(a) GA



(b) DE



(c) PSO

Fig.6 The correction error of each algorithm

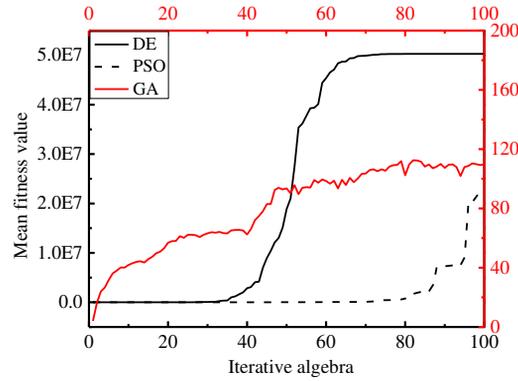


Fig.7 The average fitness of each algorithm

General engineering requires the correction accuracy to be within 1%. According to the calculation results shown in Table 5, it can be seen that the average error of a single performance parameter of the compressor obtained by the three correction algorithms meets the accuracy requirements. However, relatively, the GA algorithm correction model results have a larger deviation. From Fig.6, it can be seen that the single correction result fluctuates greatly, and there are several performance parameters exceeding 1%. For example, the N_c error of the fifth time is 1.24%. By contrast, the DE and PSO algorithm have better correction effects, the accuracy of each performance parameter can be kept within 10^{-5} , and the results of each time are basically the same. According to Fig.7, the GA algorithm tends to be stable around the 80th generation, but fluctuates greatly, the DE algorithm converges faster, and the PSO algorithm is relatively slow to converge. The result obtained above is similar to the result of solving the set of equations 1 simplified by the compressor model.

4.2 Analysis of Result

The GA algorithm selection operation is carried out according to Eq.(5) and Eq.(6). The probability of each individual being selected is proportional to the value of its fitness function. Since r is a random number in the range of $[0, fitnessX_N]$, this randomness may slow down the speed of convergence. The DE algorithm selects the next generation according to Eq.(9), and directly selects individuals with high fitness by comparing the previous generation and the population fitness value after crossover and mutation operation. The PSO algorithm selects the next generation according to

Eq.(13). V determines the direction of update for it and make the particles move closer to the optimal solution. This deterministic selection of DE and PSO algorithms accelerates the convergence speed.

The GA algorithm population individuals use binary coding. During the crossover and mutation operation, the individual coding string is first transformed according to the rules of Eq.(7) and Eq.(8). However, when the best point is at or close to the boundary of the feasible solution, the randomness of this method may prevent it from effectively continuing to optimize. For example, the theoretical solution is 1 in Eq.2. When it is close to the theoretical solution, the possible results of crossover and mutation are shown in the following table.

Table 6 Crossover and mutation results

Original code	Value	Crossed	Value	Mutated	Value
0111111111	0.9980	0100000101	0.5098	0101111111	0.7480
1000000101	1.0098	1011111111	1.4980	1010000101	1.2598

It can be seen from Table 6 that the original individual is already close to the theoretical solution, but is far away from the theoretical solution after crossover and mutation. The randomness of GA algorithm crossover-mutation can maintain the diversity of the population and avoid local optima. However, when it is close to the theoretical solution, it may make it impossible to continue to search for optimization, and the phenomenon of average fitness value will fluctuate.

The DE algorithm performs crossover-mutation operations based on the optimal solution g_{best} found so far in the entire population. The PSO algorithm performs the next-generation update based on the optimal solutions p_{best} and g_{best} found so far by the particle itself. In the early and late stages of the solution process, both two algorithms can make it continuously move closer to the optimal solution.

The PSO algorithm selects the next generation according to Eq.(12). Compared with the DE algorithm, the inertia weight ω is introduced here. Its size affects the algorithm's global and local optimization capabilities. When ω is larger, the global optimization ability will be stronger, and the local optimization ability will be weaker. According to the above example, the PSO algorithm has a slower optimization speed for problems with multiple solutions, which may cause the local optimization ability to be too weak due to the value of ω . Therefore, the simulation is performed under different values of ω , and the results are as follows,

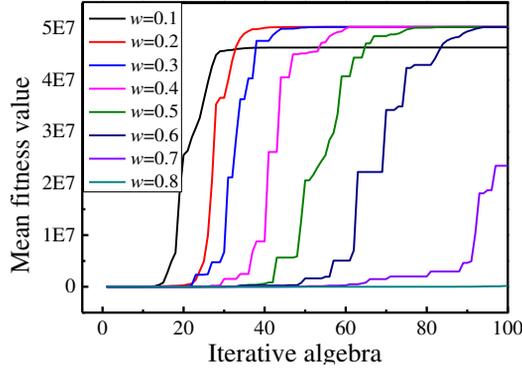


Fig.8 Mean fitness value of PSO algorithm under different ω values

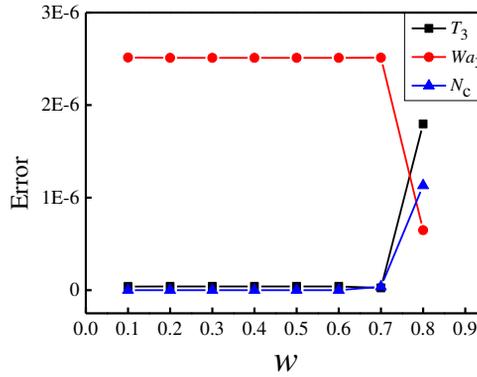


Fig.9 Relative error of performance parameters under different ω values

As shown in Fig.8 and Fig.9, with the change of ω , the correction convergence speed also changes. The larger ω , the slower convergence. The error of each performance parameter is basically the same at different values of ω , because both $\omega=0.7$ and $\omega=0.8$ do not reach optimal convergence effect, so the relative error of performance parameters has increased.

5 Conclusion

In this paper, the mechanism analysis and algorithm realization of the three main component characteristic correction optimization algorithms are carried out, and the three optimization algorithms are simulated and compared using the turbofan engine compressor component model. The algorithm performance are analyzed through the algorithm mechanism and simulation results. Specifically, the performance of each algorithm is tested in terms of correction accuracy, convergence speed and so on. It provides a reference for the selection and improvement of the component characteristic correction algorithm of turbofan engine in the future.

(1) When using the traditional GA algorithm solves multi-objective and multi-parameter problems such as model correction, due to the randomness of the algorithm, the convergence speed may be slow, the accuracy is low, and the solution is unstable. If you want to improve the performance of the correction algorithm, you need to further improve the selection, crossover and mutation operations of algorithm, and then the loss of performance caused by randomness can be offset.

(2) The traditional DE algorithm has high correction accuracy and does not need to adjust the parameters too much, but the convergence speed needs to be improved,

which can be improved from the local convergence ability.

(3) When using the traditional PSO algorithm solves multi-objective and multi-parameter problems such as model correction, the correction accuracy is not much different from the DE algorithm. The convergence speed is affected by the value of the inertia weight ω . Therefore, the algorithm parameters need to be adjusted when the PSO algorithm is used for model correction.

6 Declaration

Acknowledgements

The authors sincerely thanks to Professor Yu DaRen of Harbin Institute of Technology University for his critical discussion and reading during manuscript preparation.

Funding

Supported by the National Key Technology R&D Program of China (Grant No. 2019YFB1311104).

Availability of data and materials

The datasets supporting the conclusions of this article are included within the article.

Authors' contributions

The author' contributions are as follows: Prof. Bai Jie and Prof. Dai ShiJie were in charge of the whole trial; Dr. Liu Shuai wrote the manuscript; Dr. Wang Wei assisted with sampling and laboratory analyses.

Competing interests

The authors declare no competing financial interests.

Consent for publication

Not applicable

Ethics approval and consent to participate

Not applicable

References

- [1]Stamatis A, Mathioudakis K, Papailiou K D. Adaptive simulation of gas turbine performance. *Journal of Engineering for Gas Turbines and Power*, 1990, 112(2): 168~175.
- [2]C Kong, J Ki, M Kang. A New Scaling Method for Component Maps of Gas Turbine Using System Identification. *Journal of Engineering for Gas Turbines and Power*, 2003, 125: 979-985.
- [3]C Kong, S Kho, J Ki. Component map generation of a gas turbine using genetic algorithms. *Journal of Engineering for Gas Turbines and Power*, 2006, 128(5): 92-96.

-
- [4]C Kong, J Ki. Components map generation of gas turbine engine using genetic algorithms and engine performance deck data. *Journal of Engineering for Gas Turbines and Power*, 2007, 129(2): 312-317.
- [5]Y G Li, P Pilidis, M A Newby. An adaptation approach for gas turbine design- point performance simulation. ASME-2005-GT-68140, 2005.
- [6]Y G Li, P Pilidis, M A Newby. Improved multiple point nonlinear genetic algorithm based performance adaptation using least square method. *Journal of Engineering for Gas Turbines and Power*, 2011, 134(1): 202-207.
- [7]Dong Z, Zhou W X, Pan M X, Huang K M. Modification and updating method in component characteristics of turboshaft engine. *Aeroengine*, 2018, 44(06): 11-16.
- [8]Zhou W X, Huang J Q, Dou J P. Development of component-level startup model for a turbofan engine. *Journal of Aerospace Power*, 2006, 21(2): 248-253.
- [9]Jiang B, Wang J Q, He X L, Hu Z Z. Sub-idle Modeling of Aeroengine Based on Improved Newton's Method. *Aeroengine*, 2018, 44(01): 32-38.

Figures

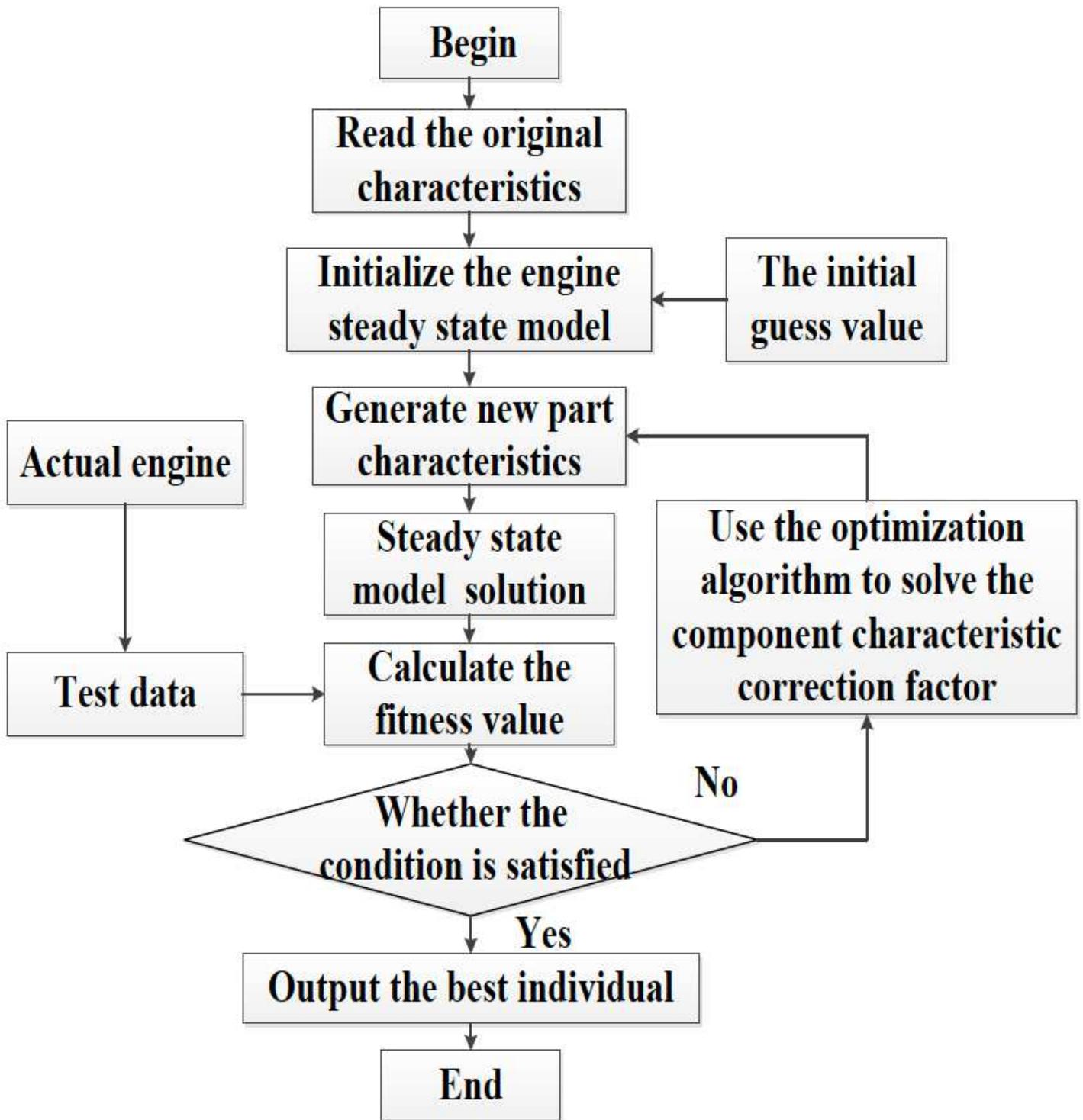


Figure 1

Component feature modification process

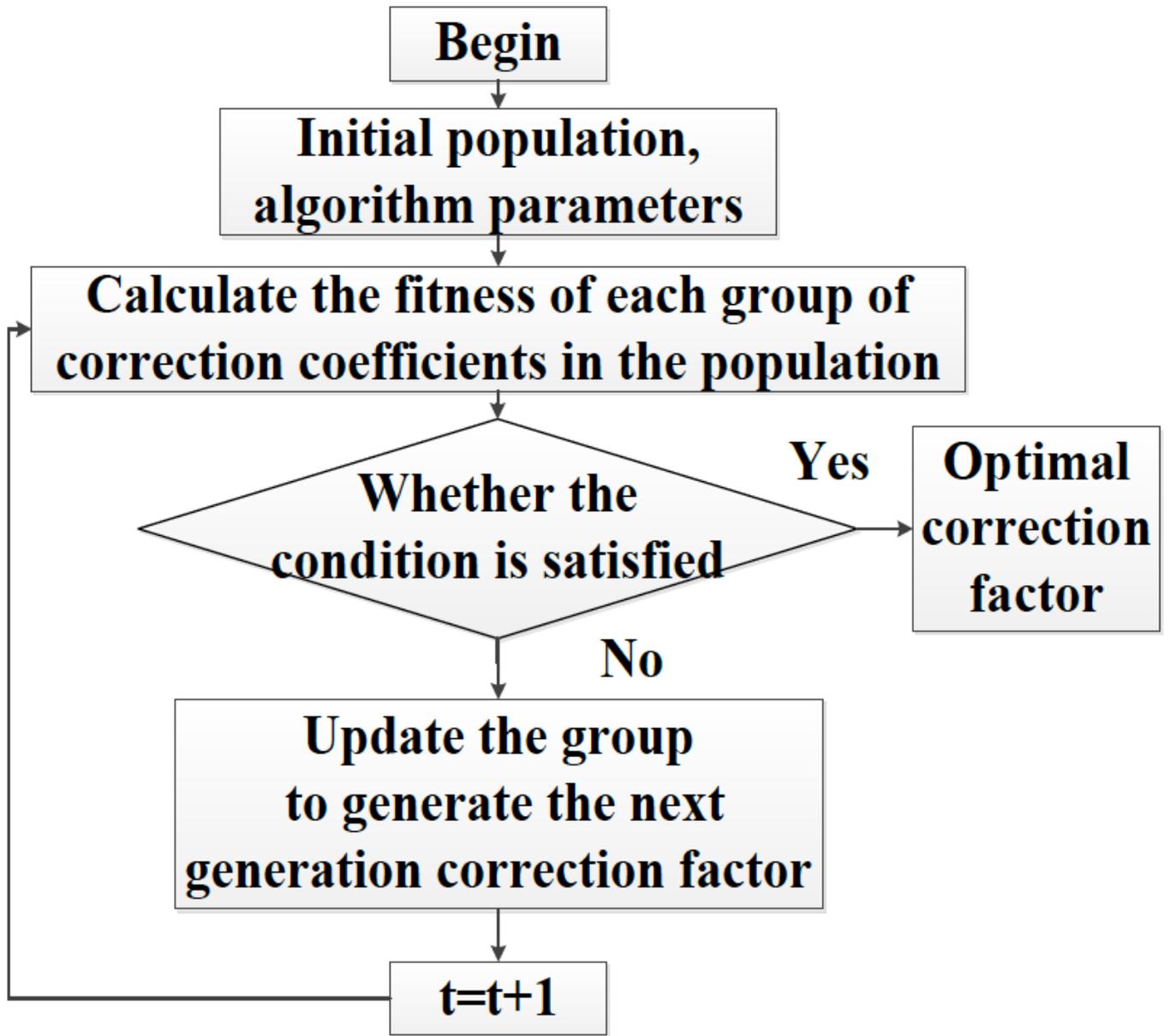


Figure 2

Optimization algorithm solution process

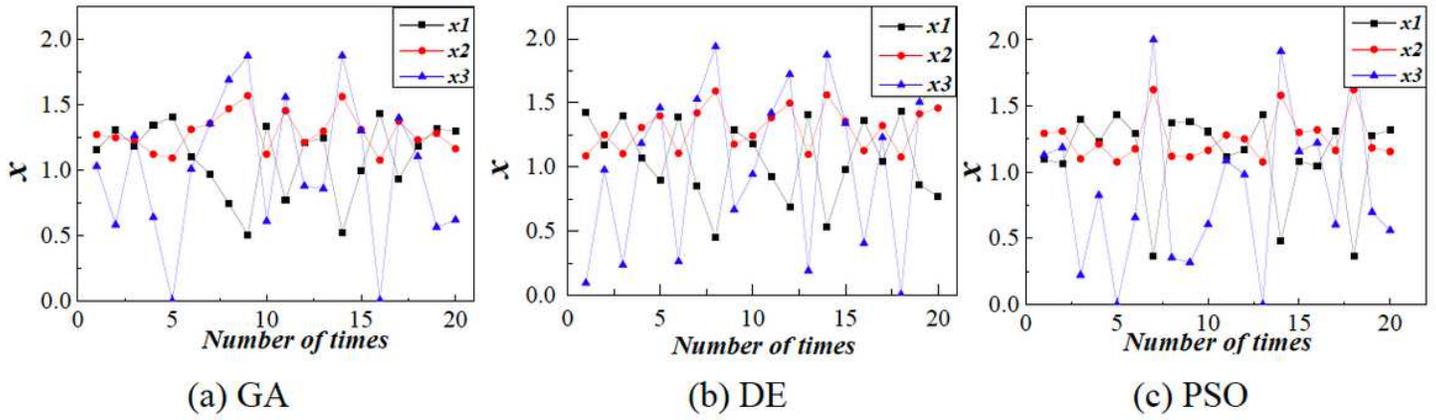


Figure 3

X value results of each algorithm running 20 times

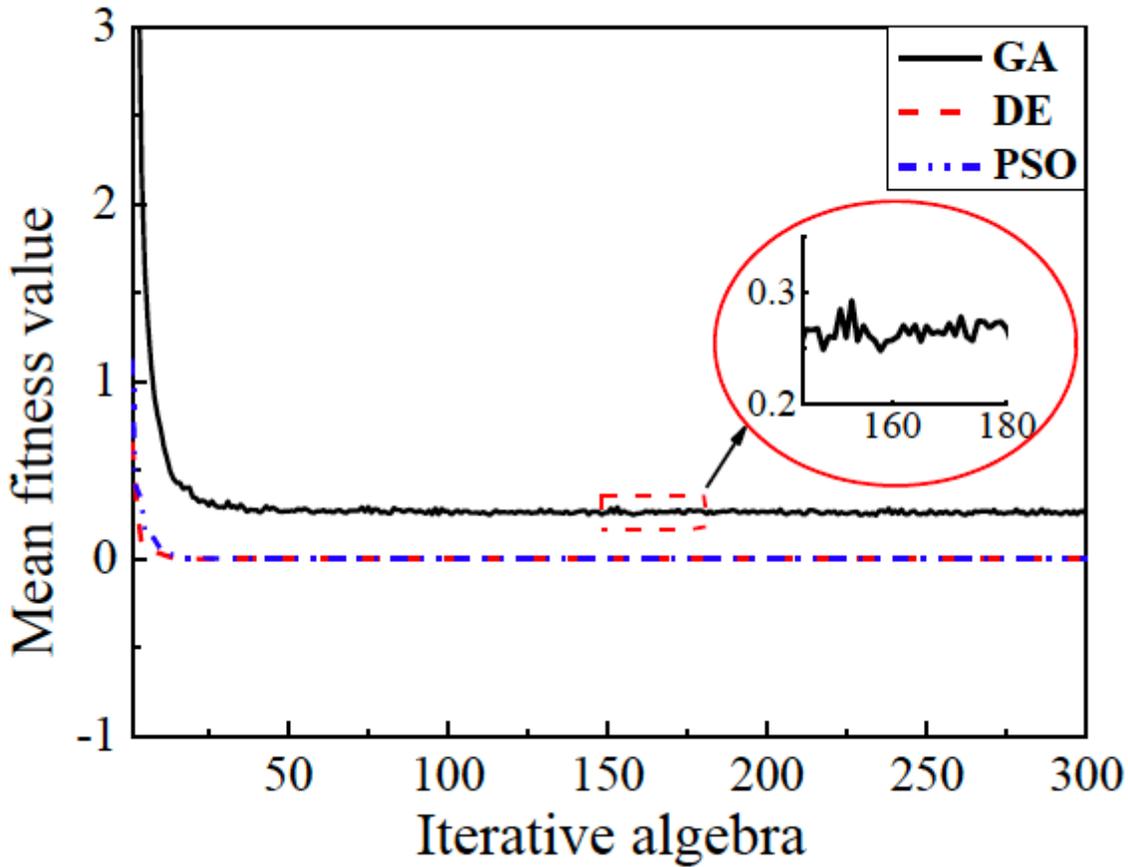


Figure 4

The mean fitness value of the system of equations 1 for each algorithm

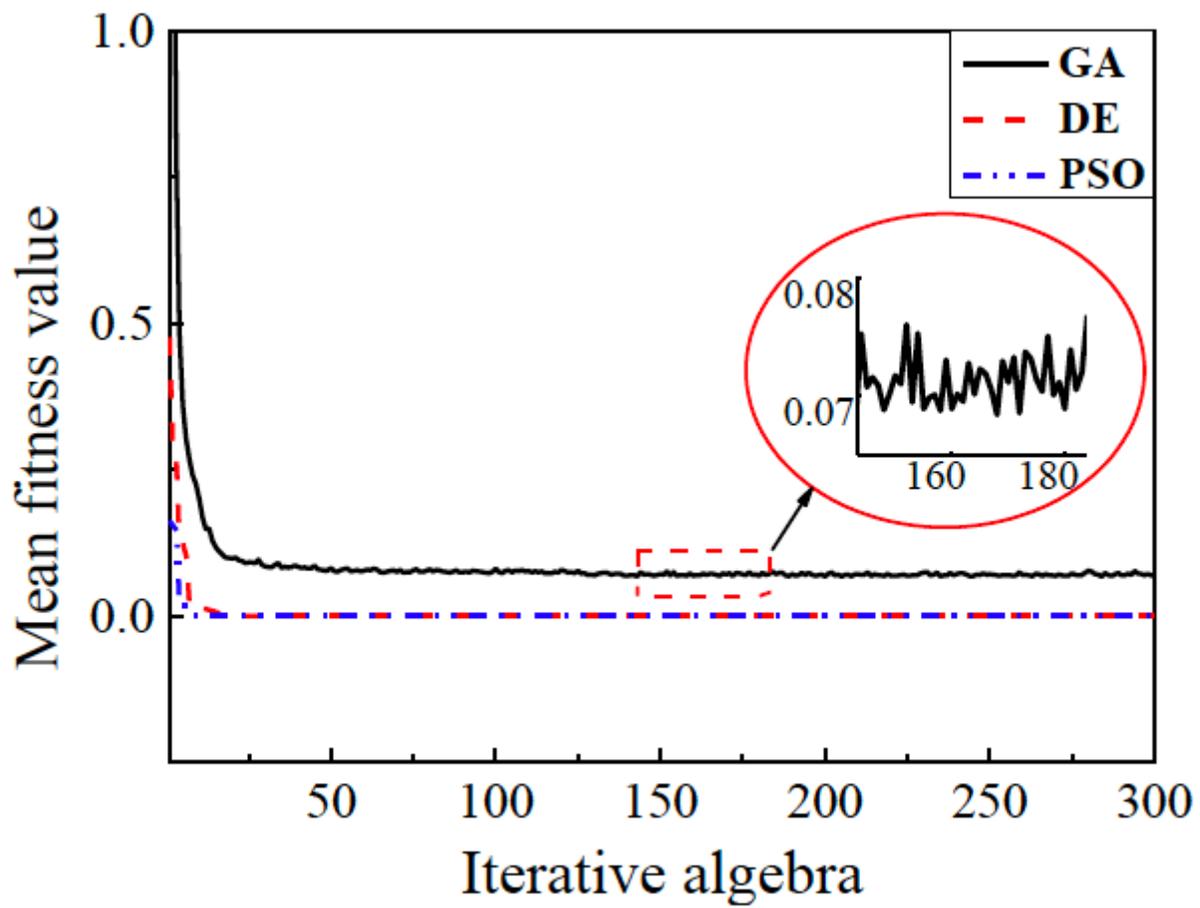
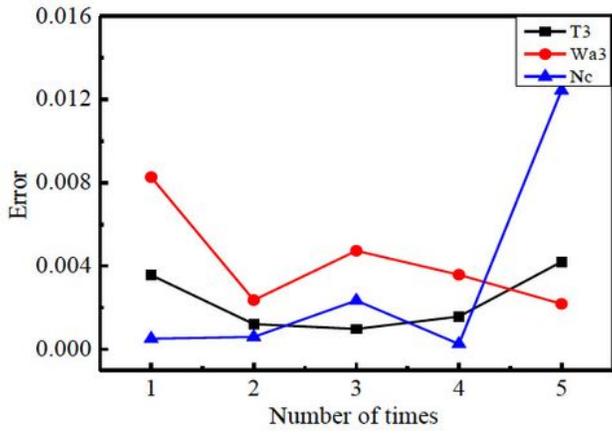
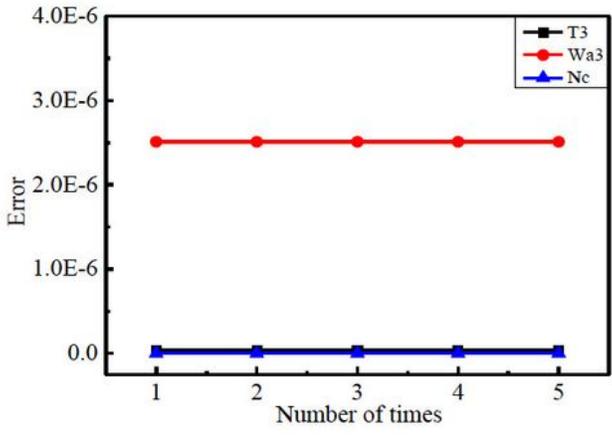


Figure 5

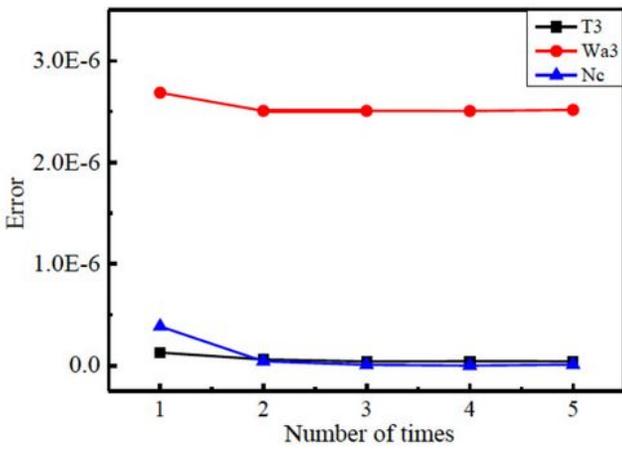
The mean fitness value of the system of equations 2 for each algorithm



(a) GA



(b) DE



(c) PSO

Figure 6

The correction error of each algorithm

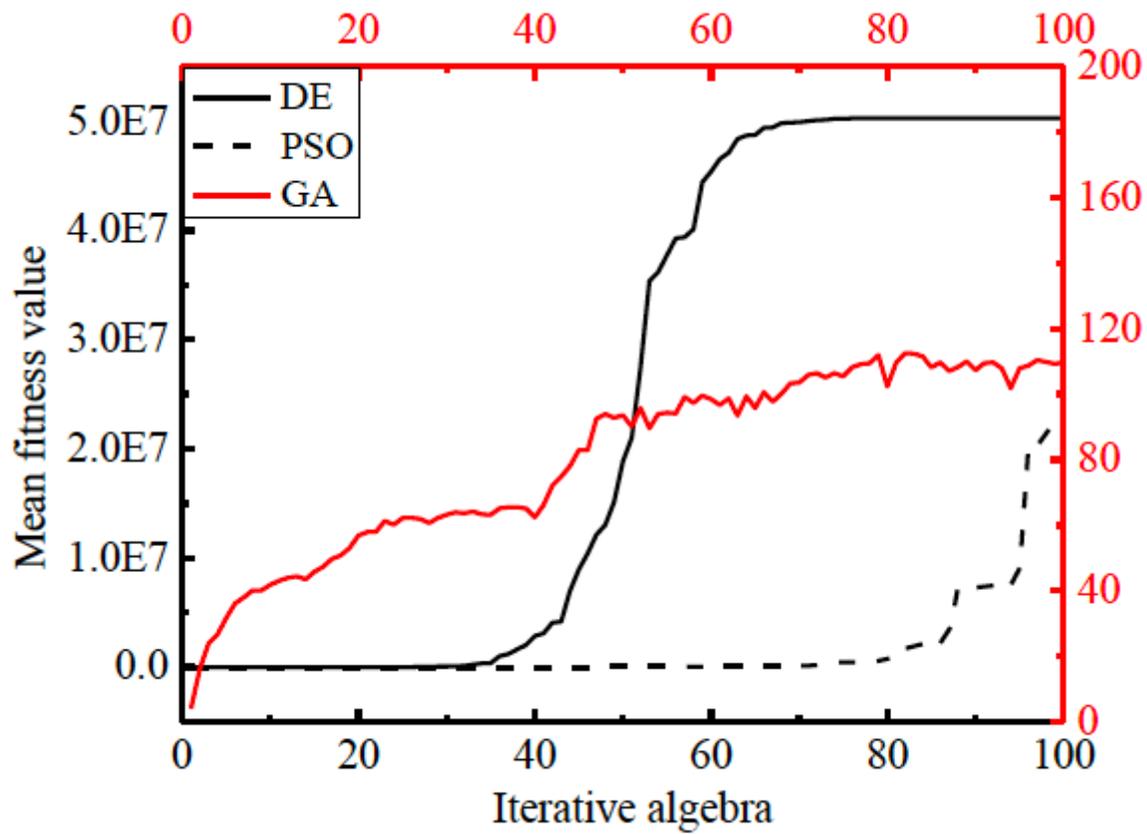


Figure 7

The average fitness of each algorithm

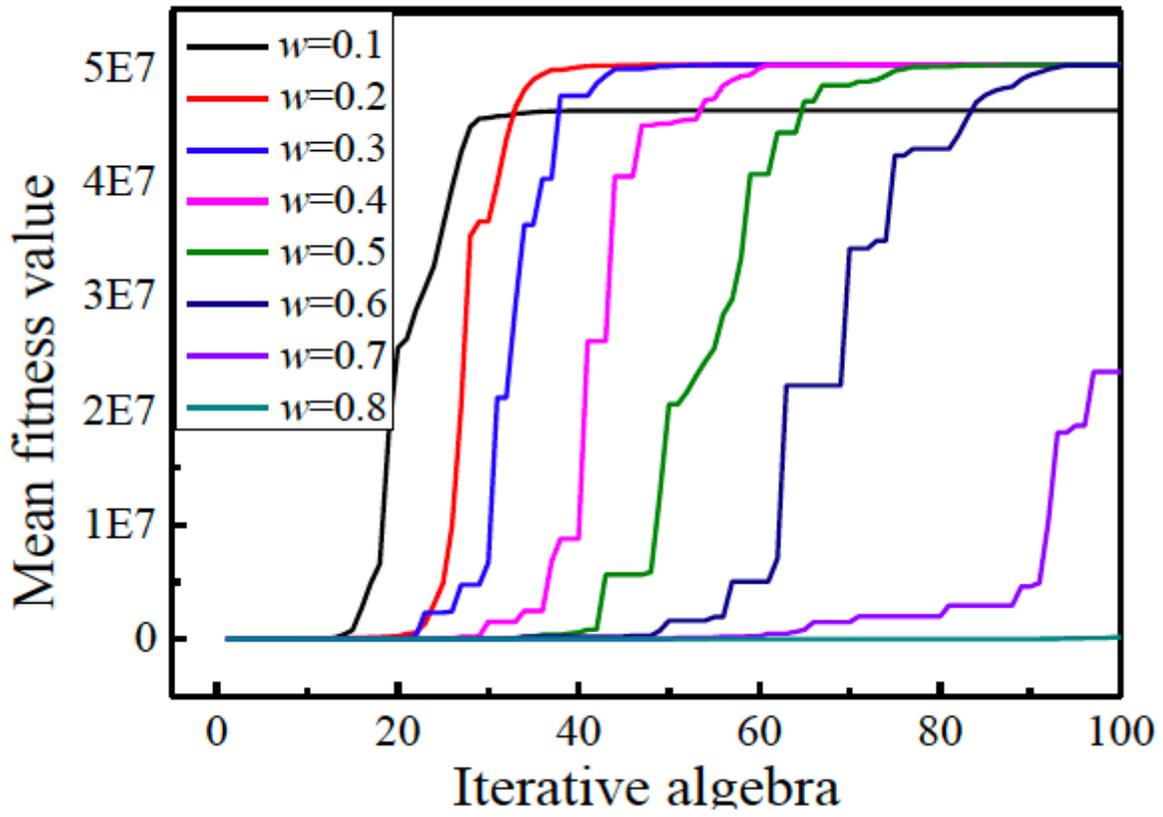


Figure 8

Mean fitness value of PSO algorithm under different w values

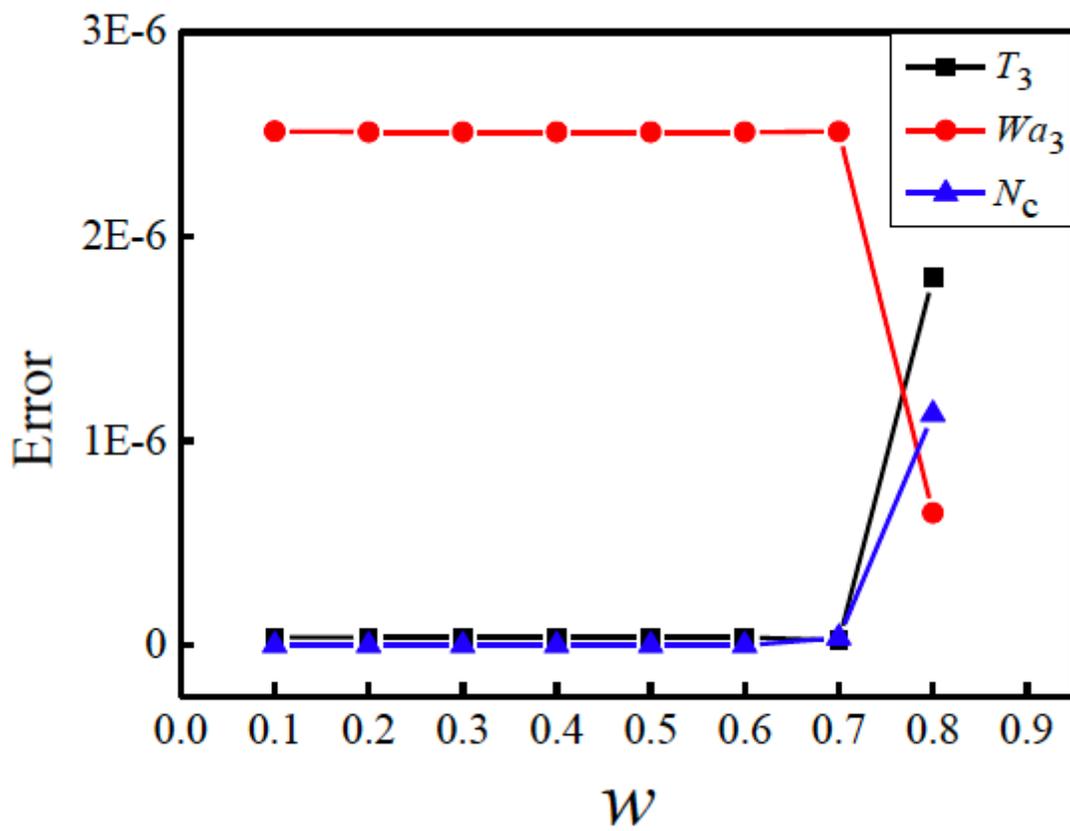


Figure 9

Relative error of performance parameters under different ω values