Original articles

A parallel Archimedes optimization algorithm based on Taguchi method for application in the control of variable pitch wind turbine

Shi-Jie Jiang[a], Shu-Chuan Chu[b,c], Fu-Min Zou[d], Jie Shan[d], Shi-Guang Zheng[d], Jeng-Shyang Pan[a,b,e],*

[a] *School of Computer Science and Mathematics, Fujian University of Technology, Fuzhou, 350000, China*
[b] *College of Computer Science and Engineering, Shandong University of Science and Technology, China*
[c] *College of Science and Engineering, Flinders University, Australia*
[d] *School of Electronic Engineering and Physics, Fujian University of Technology, Fuzhou, 350000, China*
[e] *Department of Information Management, Chaoyang University of Technology, 168, Jifeng E. Rd., Wufeng District, Taichung, 41349, Taiwan*

Abstract

Archimedes optimization algorithm (AOA) is a recent metaheuristic algorithm that offers several advantages, including a few parameters, an easy-to-understand interface, and easy implementation. Still, some drawbacks exist, e.g., lack of diversity for search-exploring capacity, drop-trap local optimum. This study suggests a new variant of AOA based on the parallel and Taguchi method (TPAOA) for the global optimization problems and the wind turbine parameter adjust-tuning variable pitch controller problem. The parallel mechanism with communication strategy and the Taguchi orthogonal combination deal with the AOA's drawbacks. The experimental results show that the proposed algorithm is more competitive than the other algorithms under the CEC2017 test suite. The wind turbine problem of parameter tuning difficulty of variable pitch controller is solved by applying the TPAOA. Compared solution results show that the TPAOA proves the feasibility smooth the output power of wind turbines and reducing the impact of wind speed fluctuations on the power grid, which has high feasibility.
© 2022 International Association for Mathematics and Computers in Simulation (IMACS). Published by Elsevier B.V. All rights reserved.

Keywords: AOA; Parallel mechanism; Taguchi method; Wind turbine

1. Introduction

With the progress of social science and technology, many optimization problems in life are becoming more and more complex [29]. It is more and more to solve these problems with traditional methods, so reliable optimization methods are needed to solve these problems virtually. The meta-heuristic algorithm has the advantages of flexibility, no gradient mechanism, and avoiding being trapped in the local optimum [39,41]. This kind of algorithm has high flexibility because only its input parameters and output data need to be considered when solving different problems. Because the algorithm is a stochastic optimization technique, it can avoid falling into the local optimum when solving problems with a large number of local optimum solutions. At present, the meta-heuristic algorithm has

---

* Corresponding author at: School of Computer Science and Mathematics, Fujian University of Technology, Fuzhou, 350000, China.
 E-mail address: jspan@cc.kuas.edu.tw (J.-S. Pan).

been used in a large number of practical application scenarios, such as path planning [24], sensor layout [23], location of logistics hub [31], digital watermarking techniques [26], and task scheduling [25].

There is no unified standard for the classification of meta-heuristics. Generally, according to the source of inspiration, it can be divided into four categories: Algorithms Based on natural evolution, algorithms based on swarm intelligence, algorithms based on physical theory, and algorithms based on human social activities [1]. The evolutionary algorithm is the simulation of the biological evolution process. The most popular of these algorithms is the Genetic algorithm (GA) [13], based on Darwinian evolution theory. Other similar algorithms include the Differential evolution algorithm (DE) [33], etc. Physics-based algorithms are inspired by physical laws, chemical phenomena, and mathematical principles. The common algorithms are the Simulated annealing algorithm (SA) [16], Sine cosine algorithm (SCA) [20], etc. Algorithms based on human social activities are proposed by simulating human habits and social activities. The main representative algorithms are the Tabu Search algorithm (TS) [10], the Teaching–Learning-Based Optimization Algorithm (TLBO) [30], etc. Swarm intelligence algorithm is to simulate the movement law and collective behavior of different biological groups in nature. For example, Particle swarm optimization (PSO) [8] is to simulate the migration of birds, and Whale optimization algorithm (WOA) [21] is to simulate the foraging process of whales. There is Cat swarm optimization (CSO) [5] and Fish migration optimization [27] and so on.

All kinds of meta-heuristic algorithms are inseparable from two critical search stages: exploration and exploitation [2]. The balanced use of the two stages can effectively enhance the ability of the algorithm to escape from the local optimal and find the optimal global solution. Exploration means that the algorithm in the global scope of space exploration can widely select the search field's global optimal solution. Exploitation means that the algorithm implements search near the local area, which is convenient for the search agent to mine the optimal global solution in a small area.

Archimedes optimization algorithm (AOA) is a meta-heuristic optimization algorithm based on physical laws proposed by scholar Hashim [12]. Based on Archimedes' buoyancy principle, the algorithm updates the position by simulating the collision between objects. The optimization is carried out by simulating the process that the object gradually presents neutral buoyancy after the collision. The AOA has the advantages of fewer parameters, simple interface, and easy implementation. But there are also some shortcomings, such as lack of diversity in search and exploration capabilities, falling into local optima, and so on. Inspired by Taguchi method and parallel mechanism, this paper proposes a parallel Archimedes optimization algorithm (TPAOA) as a communication strategy based on Taguchi method. Parallel mechanism [4] is an important optimization method, which can exchange information between groups. According to different algorithms and applications, different parallel communication strategies can be proposed to improve algorithm performance, so as to achieve faster convergence and better solutions. Taguchi's method [34] is mainly based on orthogonal tables. It conducts orthogonal experiments on individuals, strengthens the development performance of the algorithm, and can improve the accuracy of the algorithm.

Energy is an important driving force to promote human development and progress. As the main form of energy utilization, electric energy is transformed from primary energy and supplied to all walks of life. It is an important support for the development of the national economy. For a long time, fossil energy has been the main energy consumption, which leads to environmental pollution, climate change, resource shortage, and other threats to human survival. With the advancement of science and technology, the development and utilization of fossil energy such as coal, oil, and natural gas have been transformed into the development and utilization of clean energy such as water, wind, and solar energy [38]. Wind power generation occupies a large proportion in the field of new energy. The technology is relatively mature and safe. Wind power generation has become one of the most promising power generation methods. The development of wind power generation is significant for adjusting the energy structure, reducing environmental pollution, ensuring energy security, and achieving sustainable development [17].

A wind power generation system is a kind of nonlinear, large inertia, and multivariable complex system, which is usually difficult to establish an accurate mathematical model. Therefore, the control system is one of the most important wind power generation technologies. Wind power generation control technology is a comprehensive technology involving many subjects. For wind power generation control technology, variable pitch control is the key step, the main purpose of its control is to make the wind turbine achieve stable and reliable operation, capture the maximum wind energy, and high-quality power supply. PID control method is mainly used in variable pitch control. PID control has the characteristics of a simple algorithm, easy implementation, good robustness, and high stability. In the field of industrial production automatic control, PID control has been widely used and achieved great success.

PID is mainly composed of proportional, integral, and differential operations. Johnson and Smith proposed in 1976 that the fast control of pitch angle is beneficial to smooth the output of wind turbine, and PID controller was used to design the control strategy of pitch angle [14]. In literature [11,28], a complete PID controller model is applied to the pitch angle control of wind turbines in the whole wind speed range. In recent years, the rise of artificial intelligence has caused a research upsurge in the field of wind turbine control [22]. However, it is essentially a combination of artificial intelligence and PID control. So far, the PID controller is still the best choice for most wind turbines, but the parameter tuning of the PID controller brings difficulties to designers. The traditional PID parameter tuning method is Ziegler–Nichols (Z-N) tuning rule [37]. PID parameter tuning can be realized by manual parameter adjustment and experimental comparison. Therefore, parameter tuning is time-consuming and laborious. The improper parameter setting of the pitch angle controller will make the effect worse and result in unstable power output [43]. In response to these problems, TPAOA is used to optimize the PID controller parameters of the wind turbine to improve power stability. Compared with the Z-N manual PID parameter tuning method [15,19], TPAOA avoids tedious and time-consuming manual adjustment and improves power stability.

The main contributions of this paper are as follows:

1. An improved Archimedes optimization algorithm (TPAOA) is proposed, which uses the parallel mechanism and is based on the Taguchi method to achieve communication between sub-groups

2. The performance of TPAOA is verified by comparing it with other algorithms under the CEC2017 test suite

3. TPAOA is applied to PID control parameter tuning of the wind turbine to achieve the stable output of power. Compared with other algorithms, TPAOA has higher feasibility in solving this problem.

The rest of the paper is as follows. The second section reviews the original Archimedes optimization algorithm (AOA) and the working principle of the wind power generation system. The third section introduces the parallel Archimedes optimization algorithm based on the Taguchi method (TPAOA). In the fourth section, the performance of TPAOA is verified by the CEC2017 test suite, and compared with other algorithms. In the fifth section, TPAOA is applied to the variable pitch control of the wind power generation system and compared with different algorithms. The sixth section summarizes this paper and puts forward the possible development direction in the future.

2. Related work

This section introduces the mathematical model of the original AOA algorithm. The algorithm can be divided into two stages during operation: the exploration stage and the development stage. At the beginning of the algorithm, the main focus is on the exploration stage. The collision between objects makes the population as diverse as possible. As time goes by, the objects gradually reach equilibrium and enter the development stage. This stage can provide sufficient optimization depth for the algorithm. At the same time, this section also introduces the working principle of the wind power system and builds a physical model of the wind power system by studying the aerodynamic characteristics of the wind turbine.

2.1. Archimedes optimization algorithm

AOA is a metaheuristic optimization algorithm based on Archimedes' buoyancy principle. In this algorithm, the individual population is represented by submerged objects. Each individual in the population has its attributes, such as volume, density, and acceleration. Based on these properties, the object can determine its position in the fluid. At the beginning of the algorithm, the attribute and position of the object are initialized. AOA will update the object's volume, density, and acceleration in the iterative process. Finally, the object's position is updated by the attributes of the individual. The main steps of AOA are as follows [12].

Initialization

Initialize the position and attributes of the object.

$$x_i = lb_i + rand \times (ub_i - lb_i) \, ; \, i = 1, 2, \ldots, N \tag{1}$$

$$vol_i = rand \tag{2}$$

$$den_i = rand \tag{3}$$

$$acc_i = lb_i + rand \times (ub_i - lb_i) \tag{4}$$

where x_i is the ith object in the population with N objects. lb_i and ub_i are the upper and lower bounds of the search space, respectively. $rand$ is a d-dimensional vector generated randomly between $[0, 1]$. vol_i, den_i, and acc_i represent the volume, density, and acceleration of the ith object respectively.

After initialization, the objects with the best fitness value are selected by evaluating each object, and the position and attribute of the optimal object are respectively x_{best}, den_{best}, vol_{best} and acc_{best} indicate.

Update object properties:

During the iteration, the volume and density of the object are updated according to the following formula.

$$vol_i^{t+1} = vol_i^t + rand \times \left(vol_{best} - vol_i^t\right) \tag{5}$$

$$den_i^{t+1} = den_i^t + rand \times \left(den_{best} - den_i^t\right) \tag{6}$$

where vol_i^{t+1} and den_i^{t+1} denote the volume and density of the ith object in the $t + 1$ iteration.

At the beginning of AOA, there will be collisions between objects, and as time goes on, objects will gradually reach equilibrium. The simulation of the process can realize the transformation of the algorithm from exploration to exploitation. TF denotes the transition operator.

$$TF = \exp\left(\frac{t - t_{max}}{t_{max}}\right) \tag{7}$$

where t and t_{max} denote the current number of iterations and the maximum number of iterations, respectively. TF will gradually increase to 1 overtime. $TF \leq 0.5$ means that one-third of the iteration is in the exploration phase.

The update of acceleration in object attributes is related to the collision between objects. If $TF \leq 0.5$, there is a collision between objects, the acceleration update formula of object i in iteration $t + 1$ is as follows:

$$acc_i^{t+1} = \frac{den_{mr} + vol_{mr} \times acc_{mr}}{den_i^{t+1} \times vol_i^{t+1}} \tag{8}$$

where den_{mr}, vol_{mr}, and acc_{mr} is the density, volume, and acceleration of random material (mr).

If $TF > 0.5$, there is no collision between objects, the acceleration update formula of object i in the $t + 1$ iteration is as follows:

$$acc_i^{t+1} = \frac{den_{best} + vol_{best} \times acc_{best}}{den_i^{t+1} \times vol_i^{t+1}} \tag{9}$$

When the object is far away from the best position, the acceleration value is large and the object is in the exploration phase. The acceleration value is small, which means that the object is close to the optimal solution, and in the exploitation phase. Acceleration changes from big to small, indicating algorithm from the exploration phase to exploitation phase, which helps the object approach the global optimal solution and avoids falling into local optimal.

The normalization method after acceleration update is shown in Eq. (10).

$$acc_{i,norm}^{t+1} = u \times \frac{acc_i^{t+1} - \min(acc)}{\max(acc) - \min(acc)} + l \tag{10}$$

where $acc_{i,norm}^{t+1}$ represents the normalized acceleration of the ith object in the $t + 1$ iteration. u and l are the normalized ranges, which are set to 0.9 and 0.1 respectively.

Update position of the object:

If $TF \leq 0.5$ (exploration phase), the position update formula of object i at the $t + 1$ iteration is as follows:

$$x_i^{t+1} = x_i^t + C_1 \times rand \times acc_{i,norm}^{t+1} \times d \times \left(x_{rand} - x_i^t\right) \tag{11}$$

where, C_1 is a constant and its value is 2. d is the density factor. d will decrease over time. This is helpful to search from global to local and can converge in the region where the optimal solution exists.

$$d = \exp\left(\frac{t - t_{max}}{t_{max}}\right) - \left(\frac{t}{t_{max}}\right) \tag{12}$$

If $TF > 0.5$ (exploitation phase), the position update formula of object i in iteration $t + 1$ is as follows:

$$x_i^{t+1} = x_{best}^t + F \times C_2 \times rand \times acc_{i,norm}^{t+1} \times d \times \left(T \times x_{best} - x_i^t\right) \tag{13}$$

Among them, C_2 is a constant with a value of 6. F is the direction of motion and its expression is as follows. T is a variable proportional to the transfer operator, the percentage used to grab the best position, $T = C_3 \times TF$.

$$F = \begin{cases} +1 \, if \, P \leq 0.5 \\ -1 \, if \, P > 0.5 \end{cases} \tag{14}$$

where $P = 2 \times rand - C_4$.

Evaluation:

After updating the position of the object each time, the objective function is used for evaluation. By evaluating each object, record the object with the best fitness value found in the current position, to update x_{best}, den_{best}, vol_{best} and acc_{best}.

2.2. Model of the wind turbine system

In this section, the physical model of the wind power generation system is built by studying the aerodynamic characteristics of the wind turbine. The wind power generation system is mainly composed of the wind turbine, transmission mechanism, generator, variable pitch actuator, and control system.

The wind turbine is to convert the captured wind energy into mechanical energy. According to Bates theory, the mathematical model of the wind turbine can be obtained as follows:

$$\begin{cases} P_m = \frac{1}{2} \rho \pi R^2 V^3 C_p (\lambda, \beta) \\ T_m = \frac{P_m}{\omega_m} \end{cases} \tag{15}$$

where, P_m is the mechanical power of the wind turbine ; ρ is the air density; R is the blade length; V is the wind speed; ω_m is the speed of the wind turbine; T_m is the mechanical torque of the wind turbine. $C_p(\lambda, \beta)$ is the wind energy utilization coefficient, which represents the ability of wind turbines to capture power from wind energy. In engineering, the fitting curve of the wind energy utilization coefficient is often obtained according to experience:

$$C_p (\lambda, \beta) = (0.44 - 0.0167\beta) \sin \left[\frac{\pi (\lambda - 3)}{15 - 0.3\beta} \right] - 0.00184 (\lambda - 3) \beta \tag{16}$$

where β is the pitch angle, which refers to the angle between the blade and the plane of the wind turbine .λ is the ratio of blade tip speed to wind speed, that is, the ratio of the blade tip linear speed to the wind speed:

$$\lambda = \frac{R\omega_m}{V} \tag{17}$$

The transmission system is a bridge connecting wind turbine and generator, which plays the role of energy transfer. The transmission system is mainly composed of the low-speed shaft on the wind turbine side, the gearbox, and the high-speed shaft on the generator side, as shown in the figure. According to the dynamic theory, the mathematical model of the transmission system can be obtained as follows:

$$\left(J_m + k^2 J_g \right) \frac{d\omega_m}{dt} = T_m - k T_g \tag{18}$$

where J_m is the moment of inertia of the wind turbine, J_g is the moment of inertia of the generator, k is the gear ratio, which can be expressed as $k = \frac{\omega_g}{\omega_m}$, ω_m is the speed of the wind turbine , ω_g is the speed of the generator, T_m is the mechanical torque of the wind turbine, T_g is the electromagnetic torque of the generator.

The generator is a machine that converts mechanical energy into electrical energy. In this paper, the three-phase induction generator is selected as the research object, to realize variable speed regulation by changing its electromagnetic torque. The mathematical model of the ideal generator can be expressed as follows:

$$\begin{cases} T_g = \frac{pmu^2 r_2}{(\omega_g - \omega_0) \left[\left(r_1 - \frac{c_1 r_2 \omega_0}{\omega_g - \omega_0} \right)^2 + (x_1 + c_1 x_2)^2 \right]} \\ P_g = \omega_g T_g \end{cases} \tag{19}$$

where T_g is the electromagnetic torque of the generator; p is the number of poles of the generator; m is the number of stator phases of the generator; u is the rated voltage of the generator. ω_g is the speed of the generator ; ω_0 is the synchronous speed of the generator; c_1 is the correction factor; r_2 is the resistance of rotor winding reduced to

stator side; r_1 is the resistance of stator winding; x_2 is the reactance of rotor winding reduced to stator side; x_1 is the reactance of stator winding; P_g is the output power of the generator.

The variable pitch system is that controls the rotation angle of the wind turbine blade. The actual implementation process will be limited by the unit's structure, such as the change angle range limit and the change angle rate limit. For the variable pitch wind turbine, the blade can only rotate within a certain physical limit, and the allowable range of variation angle is $-2°\sim90°$. At the same time, due to the large size of the wind turbine blade, the change rate of pitch angle is slow, and the maximum change rate of general angle is only within $10°$/s. At present, the main variable pitch modes of the megawatt wind turbine include electric pitch and hydraulic pitch. Its mathematical model can be expressed as follows:

$$\frac{\beta}{\beta_r} = \frac{1}{\tau s + 1} \tag{20}$$

where τ is the time constant, β_r is the given pitch angle, and β is the current actual pitch angle.

The control system is the core part of the whole system. Due to the fluctuation of wind speed, the wind turbine's power output is not stable, so the controller is needed to control the output of pitch angle. At present, the PID controller is commonly used in wind turbines. The PID controller is composed of proportion, integral, and differential. Where, the differential part has the function of predicting the error trend, and has the effect of leading adjustment for the object with large lag. The PID controller combines the deviation between the rated power and the actual output power, and the integral and differential of the deviation to get the given pitch angle. The actual pitch angle is obtained after the value passes through the variable pitch actuator to change the capture amount of wind energy and achieve the purpose of stable output power. The continuous PID control part can be expressed as:

$$\beta_r = K_p \left(P_{ref} - P_g\right) + K_i \int_0^t \left(P_{ref} - P_g\right) dt + K_d \frac{d}{dt} \left(P_{ref} - P_g\right) \tag{21}$$

where P_{ref} is the rated power of the wind turbine, K_p, K_i, K_d represent the proportional coefficient, integral coefficient, and differential coefficient of PID controller respectively. Since the computer system is a sampling control mechanism, Eq. (21) cannot be directly adopted. It is necessary to change into discrete PID control, and then calculate the control quantity according to the deviation value of sampling time. The specific transformation of the expression is as follows:

$$\begin{cases} t = kT, (k = 0, 1, 2, \ldots) \\ \int_0^t \left(P_{ref} - P_g\right) dt = T \sum_{j=0}^{k} \left(P_{ref} - P_g^{jT}\right) \\ \frac{d}{dt} \left(P_{ref} - P_g\right) = \frac{\left(P_{ref} - P_g^{kT}\right) - \left(P_{ref} - P_g^{(k-1)T}\right)}{T} \end{cases} \tag{22}$$

where, T is the collection period, k is the sampling sequence number, and P_g^{jT} is the value of P_g at time jT. The continuous time t is represented by a series of sampling time points kT. The integral calculation is replaced by the rectangular method numerical integration, and the differential calculation is replaced by the first-order backward difference. The discrete PID control part can be expressed as:

$$\beta_r^{kT} = K_p \left(P_{ref} - P_g^{kT}\right) + K_i T \sum_{j=0}^{k} (P_{ref} - P_g^{jT}) + K_d \frac{\left(P_{ref} - P_g^{kT}\right) - \left(P_{ref} - P_g^{(k-1)T}\right)}{T} \tag{23}$$

where, β_r^{kT} is the given pitch angle at time kT. The overall structure of the wind turbine is shown in Fig. 1.

3. Parallel Archimedes optimization algorithm with Taguchi (TPAOA)

In order to improve the search performance of AOA, this section describes how to improve AOA. The exploration phase and development phase of the original AOA algorithm is not very effective. The algorithm proposed in this paper uses a parallel mechanism to improve the original algorithm, and at the same time uses Taguchi's method as a communication strategy. The combination of the two enhances the original algorithm.
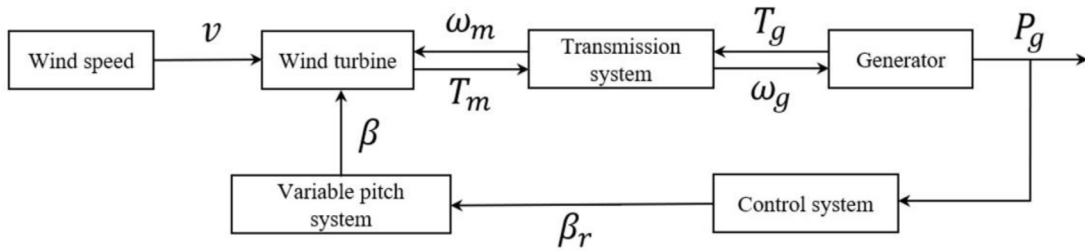
Fig. 1. The diagram of the wind turbine structure.

3.1. Taguchi method

The Taguchi method was proposed by Dr. Genichi Taguchi [34]. Taguchi method has the advantages of reducing the number of experiments, reducing the cost of experiments, and improving the efficiency of experiments [35]. Therefore, the Taguchi method is widely used in the manufacturing industry. The orthogonal matrix is one of the main tools in the Taguchi method. Suppose that the structure of an experiment is determined by K factors and each factor has Q levels. If we need to find the best level of each factor, we need to carry out K^Q times of combination experiments. The experimental efficiency will be greatly reduced [36].

Through the orthogonal matrix design combination experiment, each factor is analyzed with a small number of experiments. It can effectively reduce the number of experiments. Each row represents a combined experiment in the orthogonal matrix, and each column represents different level values of each factor. The orthogonal matrix is expressed as $L_M \left(Q^K \right)$, where K is the factor, Q is the level, and M is the number of combined experiments. Orthogonal array $L_9 \left(3^4 \right)$ is as follows:

$$L_9 \left(3^4 \right) = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 2 & 2 \\ 1 & 3 & 3 & 3 \\ 2 & 1 & 2 & 3 \\ 2 & 2 & 3 & 1 \\ 2 & 3 & 1 & 2 \\ 3 & 1 & 3 & 2 \\ 3 & 2 & 1 & 3 \\ 3 & 3 & 2 & 1 \end{bmatrix} \tag{24}$$

In a four-factor and three-level experiment, if the complete combination experiment is carried out, $3^4 = 81$ combined experiments are needed. If the experiment is arranged according to the $L_9 \left(3^4 \right)$ orthogonal table, only 9 combined experiments are needed, which greatly reduces the workload. Therefore, orthogonal experimental design has been widely used in many fields.

3.2. Parallel mechanism

In the parallel mechanism, the whole population is divided into groups at the beginning stage, forming several subgroups of equal size. Each subgroup is searched independently according to the original algorithm. The formula for updating the position of objects after grouping is as follows:

$$\begin{cases} x_{i,g}^{t+1} = x_{i,g}^t + C_1 \times rand \times acc_{i,norm}^{t+1} \times d \times \left(x_{rand,g} - x_{i,g}^t \right), TF \leq 0.5 \\ x_{i,g}^{t+1} = x_{best,g}^t + F \times C_2 \times rand \times acc_{i,norm}^{t+1} \times d \times \left(T \times x_{best,g} - x_{i,g}^t \right), TF > 0.5 \end{cases} \tag{25}$$

Each subgroup will search for the best in the search range in the iterative process. After a certain number of iterations, the subgroup and the sub-group will exchange information through different communication strategies. This kind of operation can increase the diversity of populations and realize the cooperation between groups. The parallel mechanism is shown in Fig. 2.
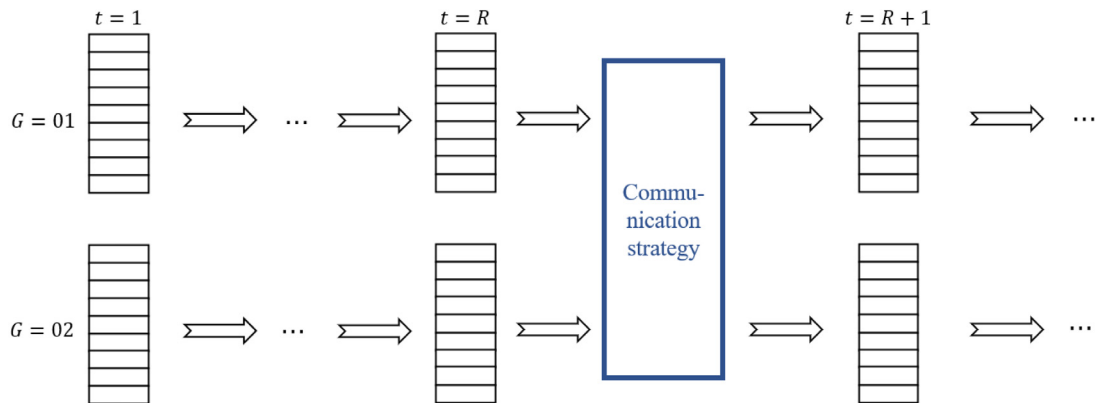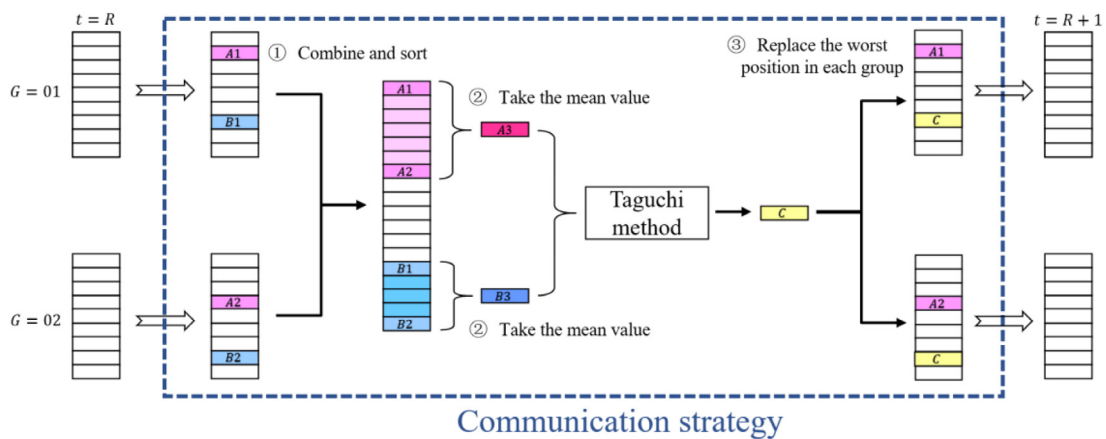
Fig. 2. The diagram of the parallel mechanism.

Fig. 3. The diagram of a communication strategy based on the Taguchi method.

3.3. Communication strategy

Different communication strategies have a great influence on the algorithm. This paper adopts the communication strategy based on the Taguchi method. The population is divided into two subgroups through the parallel mechanism to search independently and communicate after a certain number of iterations. There are the best and worst objects in each subgroup, and the positions of the two objects are marked respectively. After the combination of the two subgroups, they are arranged into a new population according to the fitness value, so that the position needed can be extracted. By averaging all positions between $A1$ and $A2$ of the best objects in the two subgroups, an average superior position $A3$ is obtained. Similarly, all positions between $B1$ and $B2$ of the worst objects in the two subgroups are averaged, and an average inferior position $B3$ is obtained. According to this operation, we can know that the number of all positions between the best or worst positions of the two subgroups is not fixed each time the communication strategy is run. If there is a big difference between the two subgroups, we can get the location information of the better and the worse objects. Taking the average of them can generate the position which can represent the better and worse population ($A3$, $B3$). The schematic diagram of a communication strategy based on the Taguchi method is shown in Fig. 3.

Each position of an object has some better dimensions. If we can mine the dimensions, we can combine them into a better position. For example, a given three-dimensional spherical function $f(x) = x_1^2 + x_2^2 + x_3^2$. It is obvious that the minimum point of this function is $[0, 0, 0]$, and the corresponding function value is 0. Suppose there are two points, $[4, 2, 0]$ and $[0, 7, 1]$, whose function values are 20 and 50 respectively, which are far from 0. However, the two points do have better information. If we can find a better dimension, we can combine it into a new point

Table 1
The position of the candidate object.

Position	Dimension							
	1	2	3	4	5	6	7	Fitness value
A3	1	0	3	6	2	0	1	51
B3	4	3	0	4	6	3	4	204

Table 2
The accumulated fitness value and the position of the new object.

Times	Dimension							
	1	2	3	4	5	6	7	Fitness value
1	1	0	3	6	2	0	1	51
2	1	0	3	4	6	3	4	87
3	1	3	0	6	2	3	4	75
4	1	3	0	4	6	0	1	63
5	4	0	0	6	6	0	4	104
6	4	0	0	4	2	3	1	46
7	4	3	3	6	6	3	1	116
8	4	3	3	4	2	0	4	70
A3 cumulative fitness value	276	288	324	346	242	288	276	–
B3 cumulative fitness value	336	324	288	266	370	324	336	–
Selected dimension source	A3	A3	B3	B3	A3	A3	A3	–
Position of the new object C	1	0	0	4	2	0	1	22

$[0, 2, 0]$. Among them, the first dimension 0 comes from the dimension corresponding to the second point, and the second dimension 2 and the third dimension 0 come from the dimension corresponding to the first point. The function value of the new point $[0, 2, 0]$ is 4, which is closer to the minimum point $[0, 0, 0]$. Therefore, a new location can be obtained by designing the experiment according to the orthogonal matrix of the Taguchi method. The specific operation is as follows. The fitness function is assumed to be:

$$f(x) = sum(x^2) \tag{26}$$

The search target is the minimum fitness value. Through the above operation, the positions $A3$ and $B3$ of candidate objects are obtained, and their dimensions are 7, as shown in Table 1.

According to the orthogonal matrix design experiment, if the value of the orthogonal matrix is 1, then select the value of $A3$ corresponding dimension. Similarly, if the value in the orthogonal matrix is 2, the value of dimension $B3$ is selected. Orthogonal matrix $L_8(2^7)$ is as follows.

$$L_8(2^7) = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 2 & 2 & 2 & 2 \\ 1 & 2 & 2 & 1 & 1 & 2 & 2 \\ 1 & 2 & 2 & 2 & 2 & 1 & 1 \\ 2 & 1 & 2 & 1 & 2 & 1 & 2 \\ 2 & 1 & 2 & 2 & 1 & 2 & 1 \\ 2 & 2 & 1 & 1 & 2 & 2 & 1 \\ 2 & 2 & 1 & 2 & 1 & 1 & 2 \end{bmatrix} \tag{27}$$

Table 2 shows the accumulated fitness value and the position of the new object of the obtained experimental results. According to the cumulative fitness value in the table, the better dimension of the two positions ($A3$, $B3$) can be mined to generate position C of the new object. It can be seen that position C is better than the original

position. Replacing some poor places in each subgroup with these positions can eliminate some poor positions and increase the diversity in the subgroups.

After the above introduction, this paper puts forward TPAOA. TPAOA adopts the parallel mechanism and the Taguchi method as a communication strategy in the parallel mechanism. At the beginning of TPAOA, the population is divided into average groups. In this paper, the number of groups $G = 2$, and the two subgroups independently run AOA. At $t = nR$, $(n = 1, 2, 3 \ldots)$, that is, the algorithm will enter the communication phase after each R iterations. The subgroups communicate with each other to improve the utilization rate of the algorithm. The communication strategy based on the Taguchi method can obtain a guide individual and replace the poor individual in each subgroup with this individual, which can eliminate some poor individuals and increase the diversity in the subgroup. Finally, the whole population is evaluated, the optimal individuals are selected and recorded.

The pseudocode for TPAOA is as follows.

Algorithm 1 TPAOA pseudo code

Input: the Population size N, dimension D, the number of iterations T, the number of groups G, the number of iterations that trigger the communication strategy R, C_1, C_2, C_3, C_4.

Initialization: initialize the position, volume, density, and acceleration of each object in the population by Eqs. (1) - (4). The population was divided into G groups. Evaluate the position of each object and select the best object in each group. The iteration counter t = 1. The group counter g.

While t < T **do**

    For $g = 1:G$ **do**

        For $i = 1:N/G$ **do**

            Update the volume and density of the object by Eqs. (5) - (6).

            The transfer operator TF and density factor d are updated by Eqs. (7) and (12)

            If $TF \leq 0.5$ **then**

                Update the acceleration of the object by Eqs. (8) and (10)

                Update the position of the object by Eq. (11)

            Else $TF > 0.5$

                Update the acceleration of the object by Eqs. (9) and (10)

                Update the position of the object by Eq. (25)

            End if

        End for

    End for

    If t = $nR, (n = 1,2,3 \ldots)$ **then**

        Use the Taguchi method to communicate

    End if

    Evaluate the position of each object and select the optimal object of the whole population

    Record the optimal object of the whole population

    Set t = t + 1

End while

Output: the optimal object of the whole population

---

4. Discussion of experimental results

This section verifies the performance of the proposed algorithms for the test suit. The obtained results are compared with other algorithms and analyzed from the optimization results and the time spent.

Table 3
Parameters settings.

Algorithm	Parameters settings
TPAOA	$G = 2, R = 20, C_3 = 2, C_4 = 0.5$
AOA	$C_3 = 2, C_4 = 0.5$
WOA	$a = 2$ to $0, b = 1, l = [-1, 1]$
PSO	$V_{max} = 10, V_{min} = -10, \omega = 0.9$ to $0.4, c_1 = c_2 = 1.49455$
PPSO	$G = 4, R = 20, V_{max} = 10, V_{min} = -10, \omega = 0.9$ to $0.4,$ $c_1 = c_2 = 1.49455$
CS	$Pa = 0.25$
cCS	$Pa = 0.25$

4.1. Experimental results for global optimization

For the performance of TPAOA, this paper will use the CEC2017 test suite [40] to evaluate and verify it. The CEC2017 test suite contains 30 benchmark functions, among which the F2 function is not used because of the dimension setting problem. Therefore, a total of 29 different test functions will be used for TPAOA evaluation. Among them, F1 and F3 are unimodal functions, F4 ∼F10 are multimodal functions, F11 ∼F20 are hybrid functions, and F21 ∼F30 are compound functions. The objective function obtained in this paper is the error function corresponding to each benchmark function Δf $\left(\Delta f = f_i - f_i^*\right)$. Where, f_i is the actual value of the ith benchmark test function, f_i^* is the minimum value of the ith benchmark test function. The smaller the error function, the better the optimization effect.

To further verify the performance of TPAOA, this paper not only compares it with AOA, but also compares it with PSO, PPSO [4], WOA [21], CS [42], and cCS [32]. To ensure the fairness of the experiment, this paper sets the basic conditions for all the algorithms. The maximum number of evaluations is set to 25 000, the size of the population is set to 200, the dimension is set to 30, and the solution range of all test functions is between [−100, 100]. At the same time, all the algorithms are tested 31 times, which makes the experimental results more objective. The basic parameter settings of each algorithm are shown in Table 3.

In this paper, the mean value, optimal value and standard deviation are used to measure the performance of the algorithm. The mean value is used to evaluate the searching ability and quality of the algorithm, and the standard deviation is used to evaluate the robustness of the algorithm. The data analysis in Table 4 shows that compared with AOA, TPAOA has better mean value in 82.8% function and better standard deviation in 65.5% function. Compared to WOA, TPAOA gets better values on most functions. The data analysis in Tables 5 and 6 shows that PPSO and cCS are improved algorithms of PSO and CS respectively. Compared with PPSO, TPAOA obtained a better mean at 70.0% of the function. It can be seen that the algorithm using the parallel mechanism strategy has a better improvement than the original algorithm. Compared with PSO, CS and cCS, TPAOA also shows strong competitiveness. TPAOA obtains the best values and average values on most benchmark functions better than other algorithms.

Meanwhile, Wilcoxon signed-rank [9] and Friedman tests [7] are performed on TPAOA and other algorithms. Wilcoxon signed-rank test pairs the results of the pairwise algorithm. When p value is greater than 0.05, the difference is not statistically significant, indicating that the difference is not significant. The specific results are shown in Table 7, where p value greater than 0.05 is indicated in bold. It can be seen that most p values are below 0.05, indicating that the optimization results of TPAOA are significantly different from other algorithms. It can be seen from Table 7 of the Friedman test results that the TPAOA ranks one and average ranking values (ARV) is 2.04, which is lower than that of other algorithms, indicating that TPAOA's optimization effect is superior to the different algorithms.

In order to more intuitively reflect the performance of several algorithms, Fig. 4 shows the convergence curves of 12 benchmark functions. In Fig. 4, TPAOA has the fastest convergence speed on the benchmark functions F6, F8, F9, and F28, and it also obtains better results than other algorithms. For most other functions, the convergence speed of TPAOA is similar to other algorithms in the early stage of algorithm execution, but in the middle and late stages of algorithm execution, TPAOA begins to show its development capabilities, and TPAOA has achieved better results than other algorithms. This is due to the Taguchi method used in TPAOA. The communication strategy of

Table 4

Performance for AOA, WOA and TPAOA under CEC2017.

ID	AOA			WOA			TPAOA		
	Mean	Best	Std	Mean	Best	Std	Mean	Best	Std
F1	3.629E+08	9.348E+07	3.326E+08	4.588E+08	2.085E+08	1.525E+08	**8.238E+07**	**1.735E+07**	**7.093E+07**
F3	2.383E+04	**9.882E+03**	**6.495E+03**	2.421E+05	1.419E+05	6.321E+04	**2.288E+04**	1.184E+04	7.137E+03
F4	1.648E+02	1.012E+02	**3.275E+01**	3.152E+02	1.705E+02	8.216E+01	**1.493E+02**	**9.226E+01**	3.541E+01
F5	1.480E+02	9.263E+01	3.104E+01	2.934E+02	1.937E+02	5.988E+01	**1.225E+02**	**7.117E+01**	**2.943E+01**
F6	2.501E+01	5.039E+00	1.078E+01	7.205E+01	5.123E+01	1.044E+01	**1.647E+01**	**3.451E+00**	**8.815E+00**
F7	2.100E+02	1.352E+02	5.005E+01	5.323E+02	3.580E+02	8.086E+01	**1.721E+02**	**1.231E+02**	**3.982E+01**
F8	1.249E+02	**6.862E+01**	2.363E+01	2.299E+02	1.417E+02	5.340E+01	**1.064E+02**	8.084E+01	**1.561E+01**
F9	2.196E+03	**4.235E+02**	**1.053E+03**	8.459E+03	4.050E+03	2.752E+03	**2.069E+03**	4.594E+02	1.265E+03
F10	**4.123E+03**	3.084E+03	7.610E+02	5.636E+03	4.613E+03	**6.996E+02**	4.186E+03	**2.798E+03**	7.986E+02
F11	2.560E+02	**1.304E+02**	6.588E+01	3.415E+03	7.670E+02	2.718E+03	**2.440E+02**	1.345E+02	**6.482E+01**
F12	6.519E+06	1.135E+06	6.003E+06	1.804E+08	2.850E+07	1.221E+08	**4.856E+06**	**6.777E+05**	**5.184E+06**
F13	9.776E+04	2.338E+04	6.866E+04	6.588E+05	1.061E+05	7.079E+05	**6.072E+04**	**1.468E+04**	**5.189E+04**
F14	2.321E+04	1.841E+03	2.252E+04	1.734E+06	2.295E+04	1.850E+06	**1.880E+04**	**9.781E+02**	**2.113E+04**
F15	7.754E+03	1.038E+03	**4.922E+03**	1.061E+05	1.900E+04	7.397E+04	**6.759E+03**	**8.099E+02**	7.019E+03
F16	**1.004E+03**	5.049E+02	**2.711E+02**	2.367E+03	1.315E+03	5.723E+02	1.061E+03	**4.659E+02**	3.036E+02
F17	3.825E+02	1.187E+02	1.733E+02	8.306E+02	3.031E+02	2.482E+02	**3.700E+02**	**8.343E+01**	**1.659E+02**
F18	3.879E+05	**3.611E+04**	4.582E+05	6.490E+06	2.592E+05	1.023E+07	**3.085E+05**	5.061E+04	**3.545E+05**
F19	**6.240E+03**	**2.369E+02**	1.113E+04	1.350E+07	1.596E+05	1.201E+07	7.039E+03	7.016E+02	**9.373E+03**
F20	**5.097E+02**	2.079E+02	**2.009E+02**	8.247E+02	**2.602E+02**	2.372E+02	5.109E+02	2.475E+02	**1.787E+02**
F21	3.124E+02	2.634E+02	**2.228E+01**	4.737E+02	3.766E+02	5.623E+01	**3.078E+02**	**2.551E+02**	2.779E+01
F22	2.215E+02	1.441E+02	**4.453E+01**	5.374E+03	4.170E+02	1.485E+03	**1.889E+02**	**1.349E+02**	5.313E+01
F23	4.929E+02	**3.986E+02**	3.657E+01	7.731E+02	6.370E+02	6.659E+01	**4.784E+02**	4.354E+02	**2.965E+01**
F24	5.536E+02	4.889E+02	4.131E+01	8.097E+02	6.516E+02	8.569E+01	**5.442E+02**	**4.865E+02**	3.669E+01
F25	4.708E+02	4.253E+02	**2.471E+01**	5.624E+02	4.729E+02	4.795E+01	**4.489E+02**	**3.920E+02**	3.042E+01
F26	**1.791E+03**	**4.651E+02**	1.039E+03	5.155E+03	2.105E+03	1.043E+03	2.211E+03	4.959E+02	**9.662E+02**
F27	5.411E+02	5.146E+02	**1.483E+01**	7.212E+02	5.466E+02	1.166E+02	**5.312E+02**	**4.984E+02**	1.782E+01
F28	5.479E+02	4.963E+02	3.695E+01	6.561E+02	5.306E+02	8.876E+01	**5.237E+02**	**4.749E+02**	**3.329E+01**
F29	1.116E+03	6.987E+02	2.411E+02	2.347E+03	1.522E+03	4.046E+02	**9.582E+02**	**6.019E+02**	**2.140E+02**
F30	7.161E+05	5.364E+04	7.439E+05	2.773E+07	2.661E+06	2.372E+07	**2.276E+05**	**3.135E+04**	**2.015E+05**
Win	24	20	19	29	29	28	–	–	–
Lose	3	9	10	0	0	1	–	–	–
Draw	0	0	0	0	0	0	–	–	–

Taguchi method can make TPAOA less likely to fall into the local optimum during the algorithm execution, and improve the accuracy of optimizing TPAOA.

4.2. Experiments to analyze the time spent

The time it takes to run the algorithm is also a performance consideration. A good algorithm should find the best solution quickly with less time. In order to represent the time consumption more intuitively, the calculation time of the comparison algorithm is given in Table 8. Fig. 5 shows the proportion of calculation time of each algorithm. In order to unify analysis and comparison, all experiments were conducted under the same conditions.

It can be seen from Table 8 and Fig. 5 that the time spent by each algorithm on the test function increases with the increase of the ID of the test function. This is because the test function comprises the single-peak, mixed-function, and combination functions. As the complexity increases step by step, more time is needed. At the same time, the improved algorithm will take more time than the original algorithm because the new mechanism is added to the enhanced algorithm.

In addition, we can also see that TPAOA takes less time than other improved algorithms in the same function. The TPAOA takes more time than AOA, but not by much. This is because TPAOA adds two new mechanics to the AOA. The Taguchi method will consume a lot of time for the algorithm. Therefore, to reduce unnecessary time expenditure, a parallelism mechanism is adopted. Parallelism by itself does not increase the time of the algorithm. Communication is needed at regular intervals for the parallel mechanism, and communication strategy is critical.
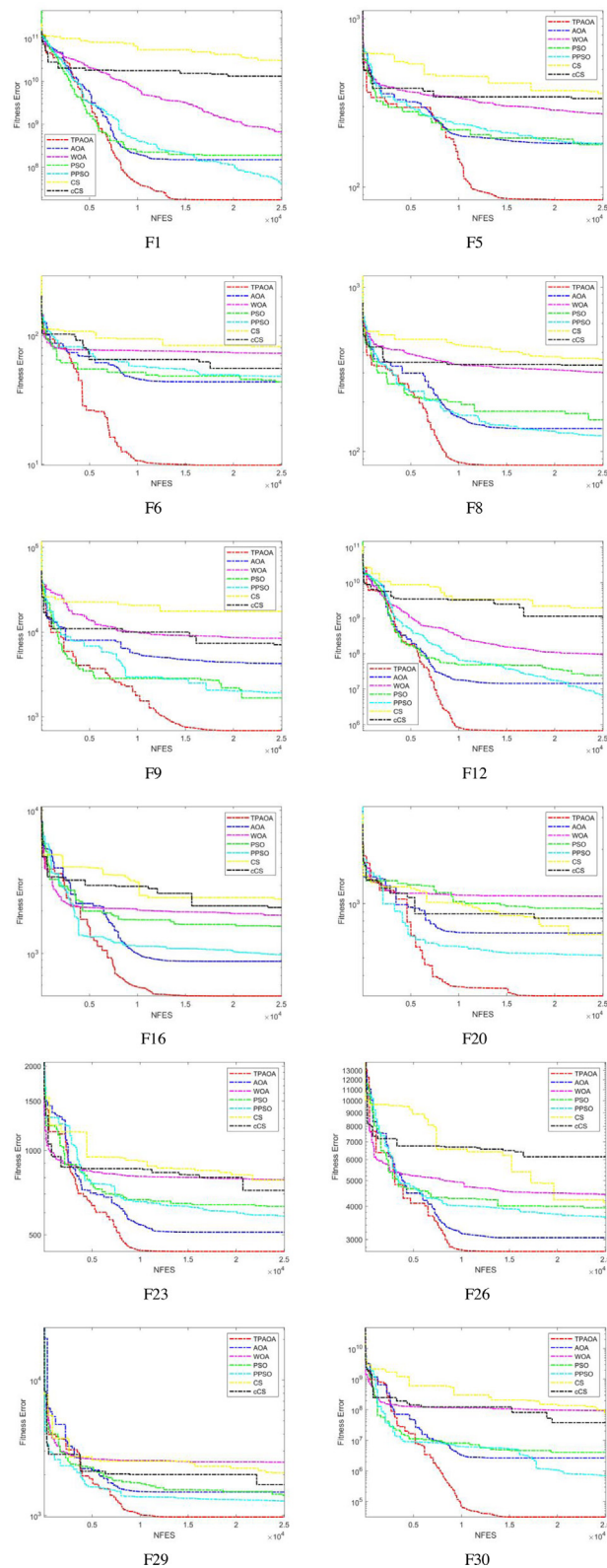
Fig. 4. Comparison of the best fitness error of 12 functions with 30D optimization.

Table 5
Performance for PSO, PPSO and TPAOA under CEC2017.

ID	PSO			PPSO			TPAOA		
	Mean	Best	Std	Mean	Best	Std	Mean	Best	Std
F1	1.374E+08	7.327E+07	3.553E+07	**2.684E+07**	**1.383E+07**	**8.325E+06**	8.238E+07	1.735E+07	7.093E+07
F3	3.485E+04	2.184E+04	7.848E+03	2.640E+04	1.562E+04	**6.068E+03**	**2.288E+04**	**1.184E+04**	7.137E+03
F4	1.294E+02	8.761E+01	2.127E+01	**1.253E+02**	**7.450E+01**	**2.085E+01**	1.493E+02	9.226E+01	3.541E+01
F5	2.115E+02	1.471E+02	3.349E+01	1.562E+02	1.002E+02	**2.393E+01**	**1.225E+02**	**7.117E+01**	2.943E+01
F6	4.101E+01	1.557E+01	1.214E+01	4.677E+01	2.734E+01	7.348E+00	**1.647E+01**	**3.451E+00**	8.815E+00
F7	2.639E+02	2.334E+02	**1.656E+01**	1.997E+02	1.531E+02	2.830E+01	**1.721E+02**	**1.231E+02**	3.982E+01
F8	1.589E+02	1.202E+02	2.269E+01	1.212E+02	8.340E+01	1.729E+01	**1.064E+02**	**8.084E+01**	**1.561E+01**
F9	2.875E+03	**6.805E+01**	1.809E+03	2.710E+03	1.121E+03	**9.753E+02**	**2.069E+03**	4.594E+02	1.265E+03
F10	5.938E+03	4.557E+03	7.205E+02	4.396E+03	2.956E+03	**6.696E+02**	**4.186E+03**	**2.798E+03**	7.986E+02
F11	2.499E+02	1.697E+02	3.732E+01	**1.928E+02**	1.379E+02	**2.828E+01**	2.440E+02	**1.345E+02**	6.482E+01
F12	1.931E+07	4.711E+06	1.020E+07	1.182E+07	2.286E+06	8.006E+06	**4.856E+06**	**6.777E+05**	**5.184E+06**
F13	1.913E+06	2.283E+05	1.244E+06	6.818E+04	2.771E+04	**2.833E+04**	**6.072E+04**	**1.468E+04**	5.189E+04
F14	4.500E+04	5.456E+03	3.854E+04	**1.590E+04**	**3.342E+02**	1.630E+04	1.880E+04	9.781E+02	2.113E+04
F15	1.861E+05	5.107E+04	1.330E+05	2.804E+04	1.176E+04	1.665E+04	**6.759E+03**	**8.099E+02**	**7.019E+03**
F16	1.232E+03	6.672E+02	**2.267E+02**	1.231E+03	7.705E+02	2.769E+02	**1.061E+03**	**4.659E+02**	3.036E+02
F17	4.601E+02	1.480E+02	**1.458E+02**	3.929E+02	1.185E+02	1.715E+02	**3.700E+02**	**8.343E+01**	1.659E+02
F18	1.241E+06	1.649E+05	1.198E+06	**1.833E+05**	**3.299E+04**	**1.043E+05**	3.085E+05	5.061E+04	3.545E+05
F19	7.695E+05	2.189E+05	6.731E+05	1.232E+05	8.374E+03	1.198E+05	**7.039E+03**	**7.016E+02**	**9.373E+03**
F20	4.819E+02	2.676E+02	1.464E+02	**4.720E+02**	3.071E+02	**1.090E+02**	5.109E+02	**2.475E+02**	1.787E+02
F21	3.867E+02	3.307E+02	2.814E+01	3.506E+02	3.040E+02	**2.541E+01**	**3.078E+02**	**2.551E+02**	2.779E+01
F22	8.918E+02	1.427E+02	1.951E+03	4.555E+02	**1.254E+02**	1.242E+03	**1.889E+02**	1.349E+02	**5.313E+01**
F23	6.291E+02	5.207E+02	7.164E+01	6.116E+02	5.353E+02	5.990E+01	**4.784E+02**	**4.354E+02**	**2.965E+01**
F24	6.700E+02	5.688E+02	4.584E+01	6.457E+02	5.329E+02	5.289E+01	**5.442E+02**	**4.865E+02**	**3.669E+01**
F25	**4.397E+02**	4.003E+02	**2.113E+01**	4.398E+02	4.058E+02	2.263E+01	4.489E+02	**3.920E+02**	3.042E+01
F26	2.399E+03	3.972E+02	1.727E+03	**1.997E+03**	**3.289E+02**	1.718E+03	2.211E+03	4.959E+02	**9.662E+02**
F27	5.865E+02	5.262E+02	5.364E+01	6.092E+02	5.177E+02	4.339E+01	**5.312E+02**	**4.984E+02**	**1.782E+01**
F28	4.894E+02	4.441E+02	**2.269E+01**	**4.822E+02**	**4.272E+02**	2.578E+01	5.237E+02	4.749E+02	3.329E+01
F29	1.244E+03	8.039E+02	2.621E+02	1.463E+03	9.018E+02	2.650E+02	**9.582E+02**	**6.019E+02**	**2.140E+02**
F30	3.356E+06	1.162E+06	1.557E+06	1.774E+06	3.025E+05	1.158E+06	**2.276E+05**	**3.135E+04**	**2.015E+05**
Win	25	25	19	20	22	12	–	–	–
Lose	4	4	10	9	7	17	–	–	–
Draw	0	0	0	0	0	0	–	–	–

This paper applies the Taguchi method to the ac strategy in the parallel mechanism. This operation can significantly reduce the time consumption caused by the Taguchi method to the algorithm, but the final solution will be better.

In general, the performance of the proposed TPAOA is better than that of the AOA, WOA, PSO, PPSO, CS, and cCS under the CEC2017 test suite.

5. Application of variable pitch control in wind turbine

In this section, in order to verify the effectiveness of the proposed algorithm, first of all, through Matlab software to mathematically model the pitch angle control system of the 3MW wind turbine with PID as the controller, and at the same time set the parameters to construct the objective function. Finally, the TPAOA algorithm is used to improve the stability of the output power in the wind power generation system. Test the proposed TPAOA with AOA, PSO, PPSO, WOA in this question. In order to ensure fairness, the population size of all algorithms is 100, and the maximum number of iterations is 2000. Each algorithm runs independently 20 times. Other parameters are consistent with Table 2.

5.1. Experimental parameter setting

In the process of wind power generation, wind turbines capture instantaneous wind energy and then convert the energy to output electrical energy. Due to the random fluctuation of wind speed, the output power cannot be

Table 6
Performance for CS, cCS and TPAOA under CEC2017.

ID	CS			cCS			TPAOA		
	Mean	Best	Std	Mean	Best	Std	Mean	Best	Std
F1	2.825E+10	1.837E+10	4.320E+09	1.476E+10	9.976E+09	2.944E+09	**8.238E+07**	**1.735E+07**	**7.093E+07**
F3	1.427E+05	9.669E+04	1.825E+04	8.676E+04	5.539E+04	1.627E+04	**2.288E+04**	**1.184E+04**	**7.137E+03**
F4	4.029E+03	2.961E+03	7.200E+02	1.821E+03	8.350E+02	5.489E+02	**1.493E+02**	**9.226E+01**	**3.541E+01**
F5	3.825E+02	3.351E+02	2.773E+01	3.217E+02	2.784E+02	**2.266E+01**	1.225E+02	**7.117E+01**	2.943E+01
F6	8.442E+01	7.414E+01	**4.583E+00**	5.392E+01	3.990E+01	7.015E+00	**1.647E+01**	**3.451E+00**	8.815E+00
F7	1.100E+03	8.248E+02	1.019E+02	5.817E+02	4.843E+02	5.838E+01	**1.721E+02**	**1.231E+02**	**3.982E+01**
F8	3.577E+02	3.164E+02	1.734E+01	2.979E+02	2.340E+02	2.531E+01	**1.064E+02**	**8.084E+01**	**1.561E+01**
F9	1.462E+04	1.094E+04	2.012E+03	7.196E+03	3.584E+03	1.869E+03	**2.069E+03**	**4.594E+02**	**1.265E+03**
F10	6.130E+03	5.710E+03	**2.062E+02**	7.549E+03	6.444E+03	3.716E+02	**4.186E+03**	**2.798E+03**	7.986E+02
F11	4.620E+03	2.331E+03	1.062E+03	1.841E+03	7.012E+02	5.908E+02	**2.440E+02**	**1.345E+02**	**6.482E+01**
F12	1.781E+09	9.909E+08	3.551E+08	1.093E+09	5.925E+08	3.171E+08	**4.856E+06**	**6.777E+05**	**5.184E+06**
F13	4.371E+08	1.949E+08	1.536E+08	1.917E+08	5.018E+07	1.330E+08	**6.072E+04**	**1.468E+04**	**5.189E+04**
F14	1.314E+05	4.317E+04	5.747E+04	4.681E+05	1.800E+04	3.673E+05	**1.880E+04**	**9.781E+02**	**2.113E+04**
F15	1.183E+07	9.458E+05	6.068E+06	4.542E+06	5.887E+05	2.815E+06	**6.759E+03**	**8.099E+02**	**7.019E+03**
F16	2.195E+03	1.928E+03	**1.400E+02**	2.307E+03	1.588E+03	2.596E+02	**1.061E+03**	**4.659E+02**	3.036E+02
F17	8.712E+02	6.713E+02	**8.532E+01**	8.756E+02	4.071E+02	2.101E+02	**3.700E+02**	**8.343E+01**	1.659E+02
F18	2.087E+06	5.478E+05	8.035E+05	5.906E+06	1.156E+06	4.594E+06	**3.085E+05**	**5.061E+04**	**3.545E+05**
F19	3.471E+07	1.423E+07	1.490E+07	8.121E+06	1.974E+06	4.660E+06	**7.039E+03**	**7.016E+02**	**9.373E+03**
F20	7.167E+02	5.612E+02	**7.040E+01**	7.154E+02	3.812E+02	1.629E+02	**5.109E+02**	**2.475E+02**	1.787E+02
F21	5.472E+02	5.080E+02	**1.828E+01**	4.944E+02	4.358E+02	2.897E+01	**3.078E+02**	**2.551E+02**	2.779E+01
F22	5.733E+03	4.411E+03	6.361E+02	1.959E+03	1.198E+03	3.669E+02	**1.889E+02**	**1.349E+02**	**5.313E+01**
F23	7.875E+02	7.513E+02	**2.463E+01**	7.012E+02	6.416E+02	2.968E+01	**4.784E+02**	**4.354E+02**	2.965E+01
F24	8.495E+02	7.774E+02	3.715E+01	7.730E+02	6.941E+02	**3.188E+01**	**5.442E+02**	**4.865E+02**	3.669E+01
F25	2.322E+03	1.391E+03	3.316E+02	1.054E+03	7.826E+02	1.582E+02	**4.489E+02**	**3.920E+02**	**3.042E+01**
F26	5.050E+03	3.646E+03	**5.671E+02**	4.031E+03	1.974E+03	1.129E+03	**2.211E+03**	**4.959E+02**	9.662E+02
F27	7.357E+02	6.946E+02	2.188E+01	7.364E+02	5.003E+02	1.065E+02	**5.312E+02**	**4.984E+02**	**1.782E+01**
F28	2.081E+03	1.321E+03	3.748E+02	1.264E+03	5.097E+02	2.552E+02	**5.237E+02**	**4.749E+02**	**3.329E+01**
F29	1.964E+03	1.693E+03	**1.324E+02**	1.945E+03	1.177E+03	3.037E+02	**9.582E+02**	**6.019E+02**	2.140E+02
F30	5.014E+07	3.164E+07	1.302E+07	3.004E+07	1.042E+07	1.259E+07	**2.276E+05**	**3.135E+04**	**2.015E+05**
Win	29	29	19	29	29	23	–	–	–
Lose	0	0	10	0	0	6	–	–	–
Draw	0	0	0	0	0	0	–	–	–

output smoothly. Therefore, it is very important to maintain stable output power by pitch control. At present, the PID controller is the most widely used controller in variable pitch mechanism. PID controller is mainly K_p, K_i, K_d three control parameters. The traditional PID parameter tuning method cannot meet the control requirements well, so it needs to rely on the designer's rich experience for secondary adjustment, which makes the design of the variable pitch mechanism more difficult. Therefore, this paper uses TPAOA to adjust PID parameters, which can not only save the time of debugging these parameters manually, but also improve the control accuracy and speed of the system. This method fills the defects of traditional PID tuning methods.

To further verify the performance of TPAOA, this paper compares it with AOA, PSO, PPSO, WOA. To ensure that TPAOA does not increase the number of function calls, the search agent of TPAOA is set to 98, and the search agent of other algorithms is 100. The number of iterations is set to 200. The range of optimization parameters is preset according to design experience, and the value range of optimization parameters is shown in Table 9.

In order to more intuitively observe whether the output power of the system is stable or not, this paper takes the index of time multiplied by absolute error integral (ITAE) as the fitness function. ITAE index is a kind of control system performance evaluation index with good engineering practicability and selectivity. Therefore, the objective function F can be expressed as:

$$F = \int t \, |e(t)| \, dt \tag{28}$$

where $e(t)$ is the error between the actual power and the rated power of the system. t is the running time of the system. The smaller the objective function is, the more stable the output power is.

Table 7
Comparison results of each algorithm with Wilcoxon sign rank test and Friedman test.

ID	TPAOA	AOA	WOA	PSO	PPSO	CS	cCS
F1	N/A	6.9641E−08	1.7018E−11	1.1205E−05	2.5668E−07	1.4018E−11	1.4018E−11
F3	N/A	7.1665E−03	1.4018E−11	2.2080E−07	1.3749E−02	1.4018E−11	1.4018E−11
F4	N/A	1.8717E−02	8.5710E−11	1.8717E−02	4.6578E−03	1.4018E−11	1.4018E−11
F5	N/A	1.5456E−02	1.4018E−11	4.8753E−11	2.7237E−05	1.4018E−11	1.4018E−11
F6	N/A	5.3326E−05	1.4018E−11	1.8376E−09	4.0332E−11	1.4018E−11	1.5447E−11
F7	N/A	6.5678E−04	1.4018E−11	4.4659E−10	9.8637E−04	1.4018E−11	1.4018E−11
F8	N/A	1.6922E−03	1.7018E−11	9.4096E−11	6.2370E−04	1.4018E−11	1.4018E−11
F9	N/A	**1.0244E−01**	1.8745E−11	**8.0856E−02**	1.2706E−02	1.4018E−11	3.6674E−11
F10	N/A	**3.9045E−01**	1.2873E−08	2.0041E−09	**2.6004E−01**	4.8753E−11	1.4018E−11
F11	N/A	**9.7754E−01**	1.4018E−11	**5.8296E−01**	1.5366E−03	1.4018E−11	1.4018E−11
F12	N/A	**1.4703E−01**	1.4018E−11	1.5439E−09	3.1620E−06	1.4018E−11	1.4018E−11
F13	N/A	1.3749E−02	6.4699E−11	1.5447E−11	4.1212E−02	1.4018E−11	1.4018E−11
F14	N/A	**8.7693E−01**	7.1071E−11	1.2847E−04	**8.2178E−01**	7.1071E−11	7.8055E−11
F15	N/A	3.7194E−02	2.5021E−11	1.4018E−11	3.6588E−09	1.4018E−11	1.4018E−11
F16	N/A	**7.2487E−01**	4.0332E−11	2.9096E−02	**7.3779E−02**	1.4018E−11	1.4018E−11
F17	N/A	**7.3779E−02**	2.1854E−09	2.7082E−02	**6.3217E−01**	1.4018E−11	3.7291E−10
F18	N/A	**9.8877E−01**	1.6408E−10	1.4729E−06	**7.2487E−01**	7.1071E−11	1.8745E−11
F19	N/A	**6.1228E−01**	1.4018E−11	1.4018E−11	6.4699E−11	1.4018E−11	1.4018E−11
F20	N/A	**8.9917E−01**	1.8183E−06	**4.8148E−01**	**4.6412E−01**	2.7567E−06	5.3326E−05
F21	N/A	2.5189E−02	1.8745E−11	1.0328E−10	4.3218E−07	1.4018E−11	1.4018E−11
F22	N/A	7.4745E−03	1.4018E−11	**3.3134E−01**	6.9641E−08	1.4018E−11	1.4018E−11
F23	N/A	3.7194E−02	1.4018E−11	2.7539E−11	2.5021E−11	1.4018E−11	1.4018E−11
F24	N/A	**2.2599E−01**	1.4018E−11	9.4096E−11	2.5970E−09	1.4018E−11	1.4018E−11
F25	N/A	1.8717E−02	7.8055E−11	**2.0014E−01**	**1.7206E−01**	1.4018E−11	1.4018E−11
F26	N/A	**8.2178E−01**	1.3643E−10	**4.4711E−01**	**3.1751E−01**	2.2729E−11	1.3989E−07
F27	N/A	2.8074E−02	3.0304E−11	1.1106E−07	2.3679E−10	1.4018E−11	3.9843E−09
F28	N/A	5.9218E−04	5.8443E−10	2.1213E−05	2.2408E−06	1.4018E−11	9.4096E−11
F29	N/A	2.0181E−02	1.4018E−11	7.1825E−05	4.3379E−09	1.4018E−11	2.2729E−11
F30	N/A	6.2370E−04	1.4018E−11	1.4018E−11	7.8055E−11	1.4018E−11	1.4018E−11
ARV	**1.6207**	2.4138	5.6207	3.6207	2.4483	6.5517	5.7241
Rank	**1**	2	6	4	3	8	7

To verify the effectiveness of the model and algorithm, this paper uses Matlab/Simulink simulation software to verify. The parameters of the wind turbine are shown in Table 10, and the wind speed model is shown in Fig. 6.

5.2. Experimental results and analysis

The experiment was carried out according to the above parameters, the convergence curve of each algorithm, the power curve of the wind turbine, the parameter setting value and the objective function value can be obtained.

From the convergence curve Fig. 7, it can be seen that compared with other algorithms, TPAOA proposed in this paper can achieve better accuracy and faster convergence in the optimization process.

In order to evaluate the performance of the power curve more objectively, this paper evaluates the power curve from the five performance indicators of mean value, maximum value, minimum value, maximum fluctuation range, and standard deviation, as shown in Table 10.

Combined with Fig. 8, Tables 11, and 12, it is not difficult to find that the output power of the controller tuned by the meta-heuristic algorithm has a good performance in all performance indicators compared with the PID controller tuned by conventional methods. This is because the meta-heuristic algorithm has the advantages of higher convergence accuracy and faster convergence speed, it is easier to find the appropriate PID control parameters, and it has better performance than the conventional PID control. Not only that but also the PID control tuned by meta-heuristic algorithm also avoids the shortcomings of the conventional PID, such as time-consuming and laborious.

Table 8
The time consumption of the compared algorithms.

ID	TPAOA	AOA	WOA	PSO	PPSO	CS	cCS
F1	2.085E+01	2.007E+01	1.896E+01	1.699E+01	6.368E+01	1.722E+01	4.046E+01
F3	2.470E+01	2.360E+01	2.260E+01	2.145E+01	6.749E+01	2.240E+01	4.531E+01
F4	2.594E+01	2.501E+01	2.400E+01	2.206E+01	6.910E+01	2.228E+01	4.583E+01
F5	2.712E+01	2.614E+01	2.519E+01	2.321E+01	6.983E+01	2.410E+01	4.751E+01
F6	3.084E+01	2.988E+01	2.844E+01	2.692E+01	7.340E+01	2.772E+01	5.123E+01
F7	2.857E+01	2.782E+01	2.662E+01	2.505E+01	7.168E+01	2.545E+01	4.901E+01
F8	3.034E+01	2.931E+01	2.851E+01	2.640E+01	7.344E+01	2.717E+01	5.132E+01
F9	3.074E+01	2.967E+01	2.858E+01	2.716E+01	7.385E+01	2.781E+01	5.190E+01
F10	3.142E+01	3.046E+01	2.960E+01	2.763E+01	7.470E+01	2.872E+01	5.260E+01
F11	3.231E+01	3.134E+01	3.063E+01	2.880E+01	7.651E+01	2.929E+01	5.398E+01
F12	3.489E+01	3.396E+01	3.266E+01	3.128E+01	7.944E+01	3.172E+01	5.775E+01
F13	3.675E+01	3.590E+01	3.423E+01	3.313E+01	8.121E+01	3.364E+01	6.009E+01
F14	3.858E+01	3.761E+01	3.659E+01	3.539E+01	8.314E+01	3.582E+01	6.174E+01
F15	3.836E+01	3.744E+01	3.639E+01	3.466E+01	8.267E+01	3.516E+01	6.085E+01
F16	4.054E+01	3.944E+01	3.926E+01	3.713E+01	8.571E+01	3.777E+01	6.396E+01
F17	4.256E+01	4.148E+01	4.107E+01	3.923E+01	8.752E+01	3.964E+01	6.578E+01
F18	4.176E+01	4.064E+01	4.068E+01	3.805E+01	8.677E+01	3.858E+01	6.439E+01
F19	5.591E+01	5.471E+01	5.455E+01	5.264E+01	1.009E+02	5.284E+01	7.856E+01
F20	4.646E+01	4.531E+01	4.512E+01	4.327E+01	9.129E+01	4.364E+01	6.910E+01
F21	5.004E+01	4.873E+01	4.921E+01	4.702E+01	9.549E+01	4.719E+01	7.282E+01
F22	5.336E+01	5.190E+01	5.178E+01	5.001E+01	9.898E+01	5.002E+01	7.573E+01
F23	5.734E+01	5.611E+01	5.582E+01	5.369E+01	1.020E+02	5.364E+01	8.091E+01
F24	6.035E+01	5.956E+01	5.840E+01	5.696E+01	1.053E+02	5.554E+01	8.362E+01
F25	6.178E+01	6.097E+01	5.950E+01	5.794E+01	1.069E+02	5.614E+01	8.409E+01
F26	6.392E+01	6.295E+01	6.263E+01	5.984E+01	1.097E+02	5.998E+01	8.626E+01
F27	6.890E+01	6.767E+01	6.651E+01	6.413E+01	1.136E+02	6.442E+01	9.146E+01
F28	7.160E+01	7.037E+01	6.809E+01	6.604E+01	1.163E+02	6.637E+01	9.343E+01
F29	6.872E+01	6.778E+01	6.775E+01	6.456E+01	1.144E+02	6.485E+01	9.153E+01
F30	8.339E+01	8.088E+01	8.392E+01	7.978E+01	1.298E+02	8.082E+01	1.080E+02

Table 9
The value range of parameters to be optimized.

Parameters	Search range
K_p	0∼8000
K_i	0∼5000
K_d	0∼5000

Table 10
The parameters of wind turbine.

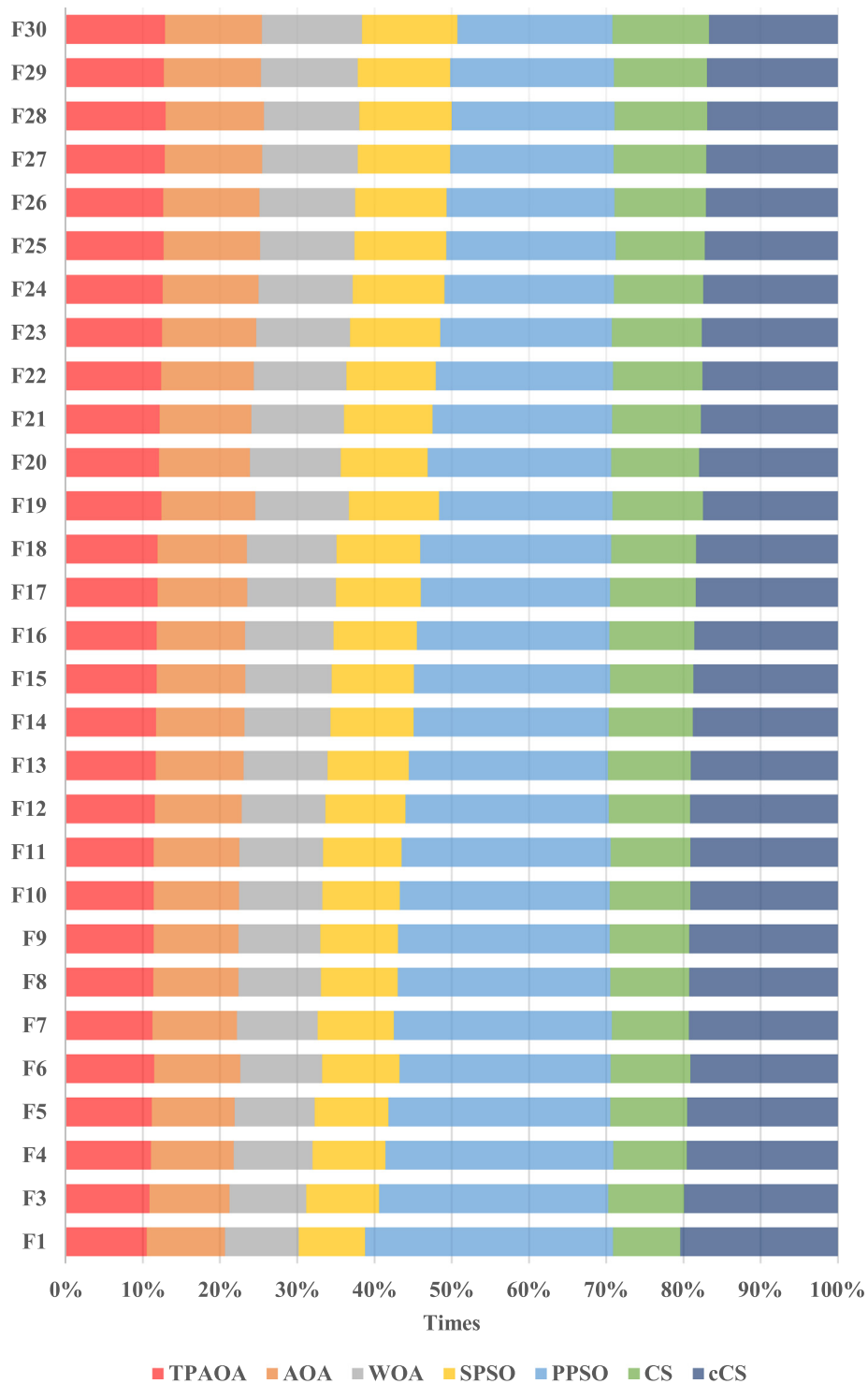| Parameters	Values
Rated power of the generator	3 MW
Rated speed of the wind turbine	3 m/s
Rated wind speed	12 m/s
Cut-in wind speed	3 m/s
Cut-out wind speed	25 m/s
Moment of inertia of the wind turbine	6250000 kg m^2
Moment of inertia of the generator	15 kg m^2
Gear ratio	80
Range of pitch angle adjustment	0∼30°
The maximum rate of pitch angle change	10°/s

Fig. 5. Comparison of the time consumption by each algorithm on different functions.
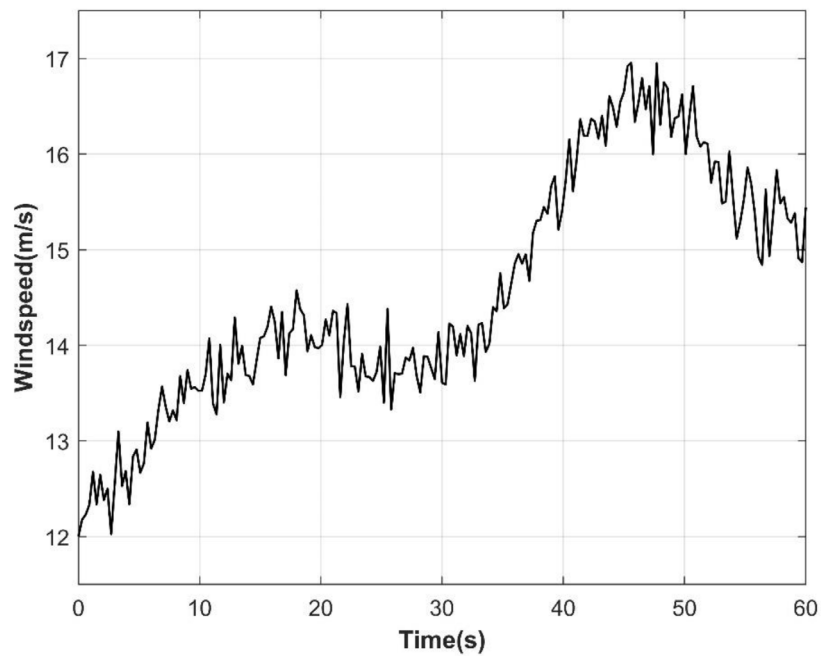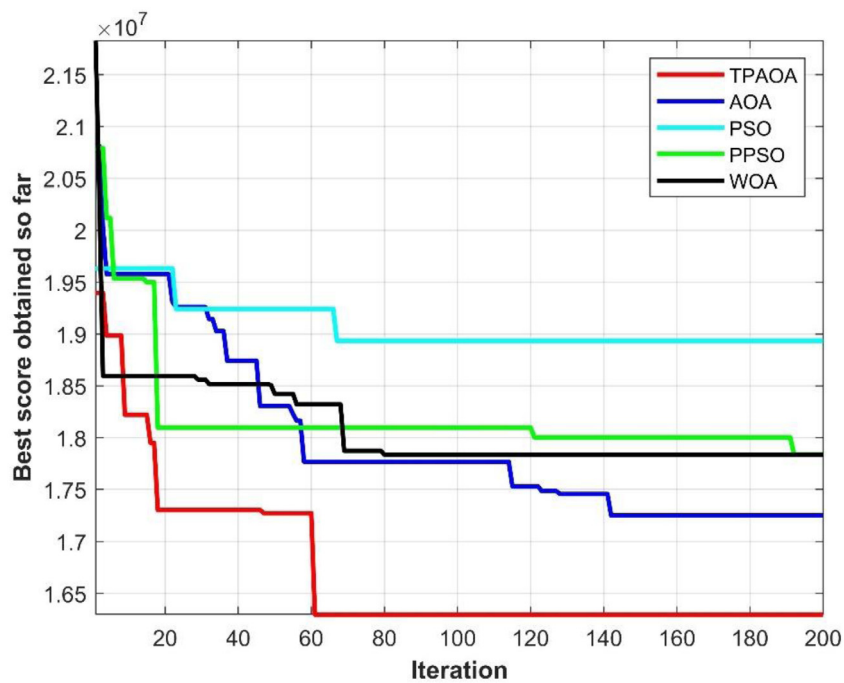
Fig. 6. The change curve of wind speed.

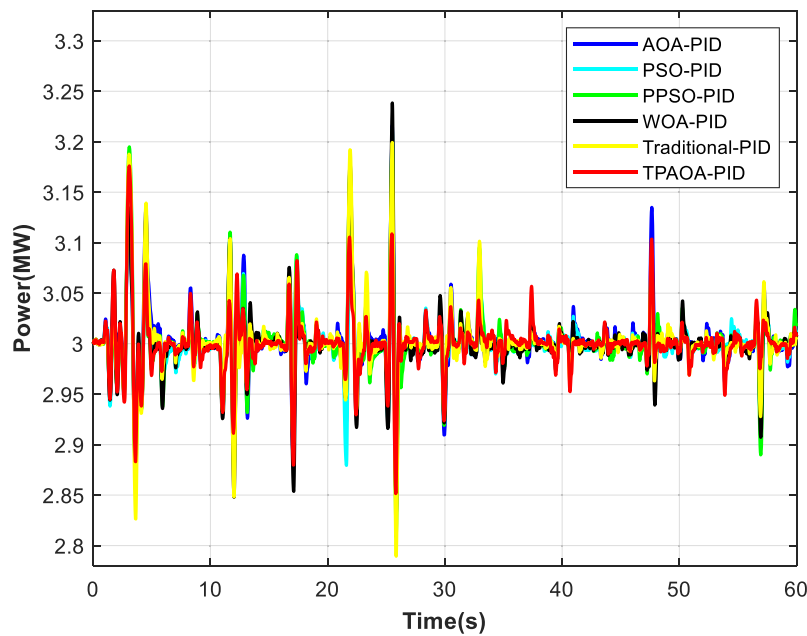**Fig. 7.** The convergence curve of various algorithms.

Fig. 8. The power curve of wind turbine.

Table 11

Optimization results of PID parameters.

	K_p	K_i	K_d	F
AOA-PID	7668.1389	37.5974	2126.3114	1.7249e+07
PSO-PID	7454.3350	2720.6619	2353.3254	1.8934e+07
PPSO-PID	4665.6989	1069.8844	1118.4964	1.7838e+07
WOA-PID	1151.7932	231.3585	227.6240	1.7835e+07
Traditional-PID	5000.0000	60.0000	900.0000	2.0907e+07
TPAOA-PID	5241.4571	608.4090	1051.7498	1.6294e+07

Table 12

The dynamic performance indicators of power curves.

Algorithm	Performance index (MW)				
	Mean value	Maximum value	Minimum value	Maximum fluctuation	Standard deviation
AOA-PID	3.001746	3.216296	2.827306	0.388990	0.028590
PSO-PID	3.000269	3.200704	2.831383	0.369321	0.026415
PPSO-PID	2.999576	3.212669	2.808700	0.403970	0.029408
WOA-PID	2.999764	3.238366	2.829633	0.408732	0.029941
Traditional-PID	3.000388	3.199228	2.789461	0.409767	0.029744
TPAOA-PID	2.999762	3.175978	2.851515	0.324463	0.022891

Comparing TPAOA with other algorithms, it is obvious that TPAOA achieves the best performance in PID parameter tuning. It can be seen that TPAOA can smooth the output power of wind turbines and reduce the impact of wind speed fluctuation on the power grid when solving the problem of variable pitch control in the wind power generation system. This method has high feasibility.

6. Conclusion

The parallel Archimedes optimization algorithm (TPAOA) based on the Taguchi method with communication strategy was implemented in this paper to parallelize the population and avoid the disadvantages of the original AOA,

such as slow convergence and low optimization accuracy. The obtained results of the TPAOA are compared with the AOA, WOA, PSO, and PPSO algorithms for the CEC2017 test suite show that the TPAOA has achieved good results in the comparison of various indicators. In solving the variable pitch control problem in the wind power generation system, the TPAOA is compared with other algorithms and traditional methods. The comparison results show that using the TPAOA algorithm in the control process can output power more stable than other methods. Therefore, the proposed TPAOA delivers excellent performance for the test suite and offers superior adaptability on the PID parameter setting of the pitch controller of the wind power unit, which improves the stability of the wind power unit. In the future work, we will further apply the proposed algorithm to maximum power point tracking of photovoltaic systems [18], photovoltaic based power prediction and short-term scheduling of hydropower plants [3,6].

References

[1] L. Abualigah, A. Diabat, S. Mirjalili, et al., The arithmetic optimization algorithm, Comput. Methods Appl. Mech. Engrg. 376 (2021) 113609.

[2] E. Alba, B. Dorronsoro, The exploration/exploitation tradeoff in dynamic cellular genetic algorithms, IEEE Trans. Evol. Comput. 9 (2) (2005) 126–142.

[3] S. Behera, S. Sahoo, B.-B. Pati, A review on optimization algorithms and application to wind energy integration to grid, Renew. Sustain. Energy Rev. 48 (2015) 214–227.

[4] J.-F. Chang, J.F. Roddick, J.-S. Pan, et al., A parallel particle swarm optimization algorithm with communication strategies, 2005.

[5] S.-C. Chu, P.-W. Tsai, J.-S. Pan. Cat swarm optimization, Pacific Rim International Conference on Artificial Intelligence, Springer, Berlin, Heidelberg, 2006, pp. 854–858.

[6] U.K. Das, K.S. Tey, M. Seyedmahmoudian, et al., Forecasting of photovoltaic power generation and model optimization: A review, Renew. Sustain. Energy Rev. 81 (2018) 912–928.

[7] J. Derrac, S. García, D. Molina, et al., A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms, Swarm Evol. Comput. 1 (1) (2011) 3–18.

[8] R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in: MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science, IEEE, 1995, pp. 39–43.

[9] S. García, D. Molina, M. Lozano, F. Herrera, A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: A case study on the CEC'2005 special session on real parameter optimization, J. Heuristics 15 (6) (2009) 617–644.

[10] F. Glover, Tabu search—Part I. ORSA, J. Comput. 1 (3) (1989) 190–206.

[11] A.D. Hansen, P. Sørensen, F. Iov, et al., Control of variable speed wind turbines with doubly-fed induction generators, Wind Eng. 28 (4) (2004) 411–432.

[12] F.A. Hashim, K. Hussain, E.H. Houssein, et al., Archimedes optimization algorithm: A new metaheuristic algorithm for solving optimization problems, Appl. Intell. 51 (3) (2021) 1531–1551.

[13] J.H. Holland, Genetic algorithms, Sci. Am. 267 (1) (1992) 66–73.

[14] C.C. Johnson, R.T. Smith, Dynamics of wind generators on electric utility networks, IEEE Trans. Aerosp. Electron. Syst. (4) (1976) 483–493.

[15] R. Karthik, A.S. Hari, Y.V.P. Kumar, et al., Modelling and control design for variable speed wind turbine energy system, in: 2020 International Conference on Artificial Intelligence and Signal Processing, AISP, IEEE, 2020, pp. 1–6.

[16] S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi, Optimization by simulated annealing, Science 220 (4598) (1983) 671–680.

[17] B. Liu, Z.-J. He, H. Jin, Wind power status and development trends, J. Northeast Dianli Univ. 36 (2) (2016) 7–13.

[18] V. Machairas, A. Tsangrassoulis, K. Axarli, Algorithms for optimization of building design: A review, Renew. Sustain. Energy Rev. 31 (2014) 101–112.

[19] V.T. Makvana, R.K. Ahir, D.K. Patel, et al., Study of PID controller based pitch actuator system for variable speed HAWT using MATLAB, Int. J. Innov. Res. Sci. Eng. Technol. 2 (5) (2013) 1496–1504.

[20] S. Mirjalili, SCA: A sine cosine algorithm for solving optimization problems, Knowl.-Based Syst. 96 (2016) 120–133.

[21] S. Mirjalili, A. Lewis, The whale optimization algorithm, Adv. Eng. Softw. 95 (2016) 51–67.

[22] E.C. Navarrete, M.T. Perea, J.C.J. Correa, et al., Expert control systems implemented in a pitch control of wind turbine: A review, IEEE Access 7 (2019) 13241–13259.

[23] T.T. Nguyen, J.-S. Pan, T.K. Dao, An improved flower pollination algorithm for optimizing layouts of nodes in wireless sensor network, IEEE Access 7 (2019) 75985–75998.

[24] J.-S. Pan, N. Liu, S.-C. Chu, A hybrid differential evolution algorithm and its application in unmanned combat aerial vehicle path planning, IEEE Access 8 (2020) 17691–17712.

[25] J.-S. Pan, Z. Meng, S. Ni, et al., A novel differential evolution approach to scheduling the freight trains in intervals of passenger trains, in: 2019 IEEE International Conference on Systems, Man and Cybernetics, SMC, IEEE, 2019, pp. 3777–3783.

[26] J.-S. Pan, X.-X. Sun, S.-C. Chu, et al., Digital watermarking with improved SMS applied for QR code, Eng. Appl. Artif. Intell. 97 (2021) 104049.

[27] J.-S. Pan, P.-W. Tsai, Y.-B. Liao, Fish migration optimization based on the fishy biology, in: 2010 Fourth International Conference on Genetic and Evolutionary Computing, IEEE, 2010, pp. 783–786.

[28] K. Qi, Research on variable pitch wind turbine control system, in: 2014 IEEE PES Asia-Pacific Power and Energy Engineering Conference, APPEEC, IEEE, 2014, pp. 1–4.

[29] S. Qin, C. Sun, G. Zhang, et al., A modified particle swarm optimization based on decomposition with different ideal points for many-objective optimization problems, Complex Intell. Syst. (2020) 1–12.

[30] R.V. Rao, V.J. Savsani, D.P. Vakharia, Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems, Comput. Aided Des. 43 (3) (2011) 303–315.

[31] P.-C. Song, S.-C. Chu, J.-S. Pan, et al., Simplified Phasmatodea population evolution algorithm for optimization, Complex Intell. Syst. (2021) 1–19.

[32] P.-C. Song, J.-S. Pan, S.-C. Chu, A parallel compact cuckoo search algorithm for three-dimensional path planning, Appl. Soft Comput. 94 (2020) 106443.

[33] R. Storn, K. Price, Differential evolution–A simple and efficient heuristic for global optimization over continuous spaces, J. Global Optim. 11 (4) (1997) 341–359.

[34] G. Taguchi, G. Chowdhury, G. Taguchi, Robust Engineering, McGraw-Hill, New York, 2000.

[35] J.-T. Tsai, T.-K. Liu, J.-H. Chou, Hybrid taguchi-genetic algorithm for global numerical optimization, IEEE Trans. Evol. Comput. 8 (4) (2004) 365–377.

[36] P.-W. Tsai, J.-S. Pan, S.-M. Chen, et al., Enhanced parallel cat swarm optimization based on the Taguchi method, Expert Syst. Appl. 39 (7) (2012) 6309–6319.

[37] D. Valerio, J.S. Da Costa, Tuning of fractional PID controllers with Ziegler–Nichols-type rules, Signal Process. 86 (10) (2006) 2771–2784.

[38] D.G. Vyas, N. Trivedi, V. Pandya, et al., Future challenges and issues in evolution of the smart grid and recommended possible solutions, in: 2019 IEEE 16th India Council International Conference, INDICON, IEEE, 2019, pp. 1–4.

[39] H. Wang, W. Wang, Z. Cui, et al., A new dynamic firefly algorithm for demand estimation of water resources, Inform. Sci. 438 (2018) 95–106.

[40] G. Wu, R. Mallipeddi, P.N. Suganthan, Problem Definitions and Evaluation Criteria for the CEC 2017 Competition on Constrained Real-Parameter Optimization, Technical Report, National University of Defense Technology, Changsha, Hunan, PR China and Kyungpook National University, Daegu, South Korea and Nanyang Technological University, Singapore, 2017.

[41] X. Xue, J. Chen, J.-S. Pan, Evolutionary Algorithm Based Ontology Matching Technique, Science Press, Beijing, 2018.

[42] S.-X. Yang, S. Deb, Cuckoo search via Lévy flights, in: 2009 World Congress on Nature & Biologically Inspired Computing, NaBIC, IEEE, 2009, pp. 210–214.

[43] B. Zahra, H. Salhi, A. Mellit, Wind turbine performance enhancement by control of pitch angle using PID controller and particle swarm optimization, in: 2017 5th International Conference on Electrical Engineering-Boumerdes, ICEE-B, IEEE, 2017, pp. 1–5.

Further reading

[1] Z.M. Ali, I.M. Diaaeldin, A. El-Rafei, et al., A novel distributed generation planning algorithm via graphically-based network reconfiguration and soft open points placement using archimedes optimization algorithm, Ain Shams Eng. J. 12 (2) (2021) 1923–1941.

[2] Y. Li, H. Zhu, D. Wang, et al., Comprehensive optimization of distributed generation considering network reconstruction based on archimedes optimization algorithm, in: IOP Conference Series: Earth and Environmental Science, vol. 647, (1) IOP Publishing, 2021, 012031.