IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 5 August 2023, accepted 28 August 2023, date of publication 1 September 2023, date of current version 7 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3311271

RESEARCH ARTICLE

T Cell Immune Algorithm: A Novel Nature-Inspired Algorithm for Engineering Applications

HONGZHI ZHANG, YONG ZHANG, YIXING NIU, KAI HE, AND YUKUN WANG
School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan 114051, China

Corresponding author: Yong Zhang (zy9091@163.com)

This work was supported by the Scientific Research Project of Liaoning Provincial Department of Education under Grant LJKZ0320.

ABSTRACT In this paper, inspired by the T cell immune process, a new meta-heuristic algorithm named T cell immune algorithm (TCIA) is proposed to better solve various complex practical problems. The core difficulty of meta-heuristic algorithms is balancing exploration and exploitation. TCIA mimics the behavior of T-cell immune process in recognizing antigens, activating cells and attacking pathogens, thus to better balance exploration and exploitation. Recognition is the process of judging the cell concentration at the current location, which is used to evaluate the fitness value of the current cell. Activation is the process by which T cells are activated and thus differentiate into multiple cells, helping the population to explore on a larger scope. Attacks are divided into random attack and directed attack. Random attack helps the population escape from local optima by randomly selecting a target cell to attack. Targeted attacks push the overall approach to the location with the highest concentration of target cells, which makes better use of the global optimal solution. The TCIA is tested on CEC2022 and compared with other 10 algorithms. Experimental results and statistical analysis show that TCIA is more effective than other 10 classical and new meta-heuristic algorithms in solving constraint function problems. TCIA is tested on three multi-objective functions, which verifies its good ability in solving multi-objective functions. Two classical engineering design problems are applied to verify the ability of TCIA to solve practical problems. The parameters of Back propagation neural network are optimized by TCIA to predict the compressive strength of concrete. The results show that the ability of TCIA to solve a variety of problems is balanced and excellent.

INDEX TERMS T-cell immune algorithm, meta-heuristic algorithm, engineering design problem, neural network, prediction of concrete compressive strength.

I. INTRODUCTION

Meta-heuristic algorithm is an effective solution method to solve optimization problems by simulating natural phenomena or human behavior through computer and mathematical methods [1]. Most real-world problems are nonlinear. Although accurate algorithms can find the optimal solution, the difficulty of solving the problem increases exponentially as the complexity of the problem increases. The solution directions of practical problems are often undiscovered and very complex, and there are a large number of local optimums [2], so meta-heuristic algorithm is a good choice for optimizing these challenging practical problems. Its advantages are as follows: (1) The meta-heuristic algorithm can combine the advantages of many different heuristic algorithms to improve the optimization effect. (2) The meta-heuristic algorithm is less affected by data changes, noise and other factors, so the stability of the algorithm is higher [3]. (3) Meta-heuristic algorithm has a stronger ability to avoid local optimization than traditional optimization methods [4]. This is because of its stronger randomness [5]. (4) Multiple solutions in meta-heuristic algorithm can cooperate with each other to push the overall approach to the optimal value. Its optimization results can be more easily interpreted

The associate editor coordinating the review of this manuscript and approving it for publication was Sun-Yuan Hsieh.

and understood, which contributes to a better understanding of the problem itself [6]. In conclusion, meta-heuristic algorithm has a great significance in improving optimization effect, stability and applicability. It has become an important method to solve complex optimization problems.

Meta-heuristic algorithm is developing rapidly. Since the introduction of the Simulated annealing algorithm in 1953, various meta-heuristic algorithms and their variants have been studied. The inspiration of Simulated annealing algorithm comes from the change of lattice structure during metal cooling. The Simulated annealing algorithm finds a solution through random search and is applied to solve combinatorial optimization problems, such as knapsack problems [7]. In 1975, people began to study Genetic algorithms. Darwin's theory of evolution is the inspiration for the algorithm. Genetic algorithm finds the best solution by simulating natural genetic phenomena and is generally used to solve combinatorial optimization problems [8]. In 1992, people started to study the Ant colony optimization(ACO). ACO finds the optimal solution by mimicking the behavior of ants that leave pheromones when searching for food. ACO can be applied to path planning [9], data clustering [10] and other problems. Since 1995, scholars have begun to study Particle swarm optimization algorithm(PSO) [11]. The PSO is related to the predation of birds. Flocks of birds gather near optimal locations by sharing food sources. The PSO algorithm is simple and easy to implement, and has high computational efficiency. However, it is sensitive to the choice of parameters and it is not effective in solving problems with strict constraints. The Artificial fish swarm algorithm (AFSA) was proposed in 2003 to find the optimal solution by the characteristics of fish gathering around food [12]. AFSA has fast convergence speed and high efficiency, but it also has the problems of high time complexity and imbalance between global search and local search. In 2006, people began to study the Bee colony optimization algorithm [13]. This algorithm is inspired by the behavior of bees as they search for flowers. Bee colony optimization algorithm can be applied to combinatorial optimization and data clustering. The Fruit fly optimization algorithm (FOA) was proposed in 2012 [14]. FOA is a simple and feasible optimization algorithm, but it is easy to fall into local optimum when solving high-dimensional nonlinear functions. In 2020, the Sparrow search algorithm (SSA) was proposed based on the behavior of sparrows looking for food and avoiding predators [15]. The SSA has strong local search ability, but its global search ability is weak and easy to fall into local optimum. Meta-heuristic algorithm has high efficiency and stability in solving optimization problems. Therefore, it has been widely used in practical production and life.

As the research progresses, meta-heuristic algorithm has aroused the interest of more and more scholars, and has been applied to optimize a large number of real-world problems [16]. Meta-heuristic algorithm is well suited to solve the optimization problems of complex systems, such as network design [17], smart grid scheduling [18], supply chain optimization [19] and so on. In the field of machine learning, meta-heuristic algorithm can optimize the hyperparameters of models such as deep neural networks [20] and support vector machines [21]. Scheduling problem [22], bin packing problem [23] and traveling salesman problem [24] are all classical combinatorial optimization problems, which can be well solved by meta-heuristic algorithm. Meta-heuristic algorithm can also be applied to optimize practical problems in other fields, such as energy management [25], intelligent manufacturing [26], financial investment [27], truss design problems [28], multi-objective structural optimization problem [29] and so on. In summary, the meta-heuristic algorithm has a wide range of applications and is capable of solving a variety of complex optimization problems. Due to their recognized advantages, these algorithms have gained widespread adoption. In the future, with the continuous advancement of artificial intelligence and big data technology, the meta-heuristic algorithm is promising to be more widely used and explored.

In practical applications, there are many factors to be considered in the optimization problems. They include but are not limited to nonlinear constraints, non-convex search intervals, high computational cost, large solution spaces and dynamic objective functions. As the difficulty of solving practical problems increases, some classical algorithms are difficult to apply, so more and more improved algorithms and new algorithms are proposed. According to the "No free lunch theorems for optimization" [30], when the ability of any algorithm to solve one class of problems increases, its ability to solve another class of problems will decrease. The purpose of this paper is to find a new algorithm to solve a variety of problems ability balance, so as to better solve a variety of practical problems.

Inspired by the mechanism of T cell immunity, a new meta-heuristic algorithm called T cell immune algorithm (TCIA) is proposed in this paper. The main contents and structure of this paper are plotted as a graphical abstract shown in FIGURE 1. TCIA simulates the behaviors of recognizing antigens, activating cells and attacking pathogens in the process of T cell immunity, thus realizing three main steps of recognition, activation and attack [31]. The first section introduces some basic information and application scope of intelligent optimization algorithms. The second section introduces the proposed TCIA algorithm, and explains its inspiration source and mathematical model. The third section verifies the ability of TCIA to solve constrained problems through single and multi-objective optimization functions. At first, TCIA is tested on 12 test functions of CEC2022 and compared with Atom search optimization (ASO), Grey wolf optimizer (GWO), Sine cosine algorithm (SCA), Whale optimization algorithm (WOA), Butterfly optimization algorithm (BOA), Harris hawks optimization (HHO), Beluga whale optimization (BWO), Aquila optimizer (AO), Pelican optimization algorithm (POA), Osprey optimization

algorithm (OOA). Second, ASO is the best performing algorithm in the CEC2022 test set except TCIA, so it is compared with TCIA on three multi-objective functions. Through the test of single-objective functions and multiple-objective functions, it is verified that TCIA has high efficiency and universal applicability in solving constrained problems. Third, the results of all 11 algorithms tested on CEC2022 are analyzed by five statistical analysis methods. They are box plot, status tracking of four random cells Wilcoxon rank sum test, Friedman rank test and radar chart. The results of five statistical analyses show that TCIA is more stable and superior in solving constrained function problems compared with the other ten comparison algorithms. To verify the ability of TCIA to solve practical problems, it is tested on pressure vessel design and welded beam design in the fourth section. The parameters of BP neural network are optimized by TCIA to predict the compressive strength of concrete. The fifth section summarizes the performance of all aspects of TCIA, and looks forward to the future work direction.

II. T CELL IMMUNE ALGORITHM (TCIA)

T cell immune algorithm is a new optimization algorithm by simulating T cell immune process. To better understand this new algorithm, some facts about cellular immunobiology and the algorithm construction process are discussed in the following sections.

A. INSPIRATION

This section mainly introduces the composition of the immune system and the role and workflow of T cells in it. The immune system relies heavily on T cells. Because T cells have the ability to recognize and attack pathogens, cancer cells and other abnormal cells [32]. T cells are generated in the thymus and then migrate to lymphoid tissue. When a pathogen enters the body, some specific proteins and other substances on its surface are able to bind to receptors on the surface of immune cells, thus triggering an immune response. These substances that can trigger an immune response are called antigens [33]. B cells, dendritic cells and macrophages are able to absorb and process antigens and can expose antigenic information to the cell surface for presentation to other immune cells, so these cells are collectively referred to as antigen-presenting cells (APC) [34]. When foreign antigens are detected by the immune system, antigen-presenting cells ingest and decompose antigens and present them to helper T cells. T cell receptors (Tcr) on the surface of helper T cells (Th) recognize and bind specific antigenic peptide (MHC complexes) [35]. When a Th recognizes a specific antigen, it releases immune cytokines to activate the naive T cell (Tn). Naive T cells become active T cells (active Tn) when activated, thus differentiate into cytotoxic T cells (Tc), memory T cells (Tm), regulatory T cells (Treg) and Th. The differentiation process is shown in FIGURE 2 Th produces cytokines that can boost immune responses and attract other immune cells to participate. Tc kill infected target cells by recognizing MHC molecules on antigen-presenting cells and releasing a variety of cytotoxic molecules. Tm remember the antigen that triggered the immune response and can quickly re-identify and attack the same antigen in the future. Treg also known as suppressor T cells, can inhibit immune response, regulate the immune response process, and prevent autoimmune diseases and excessive inflammatory responses. At the end of cellular immunity, T cells involved in the immune process will gradually apoptosis, except Tm [36]. The diagram of T cell immune process is shown in FIGURE 3. Inspired by the above T cell immune process, a T cell immune algorithm (TCIA) is proposed.

B. MATHEMATICAL MODEL AND ALGORITHM

T cells can be classified into four common types based on their function and characteristics: Tc, Tm, Treg and Th. These four types of T cells undertake the main task of T cell immunity. The proposed TCIA is mainly composed of three parts: recognizing antigens, activating cells and attacking pathogens. The mathematical modeling process is reflected in the following three sections.

1) IDENTIFY

The first step in T cell immunity is the recognition of antigens carried by antigen-presenting cells through helper T cells. When normal cells are infected, antigen-presenting cells ingest the infected cells and produce antigens on the surface. The T cell receptors of helper T cells bind to newly generated antigens to complete the antigen recognition process. The recognition step of TCIA is the process by which helper T cells recognize antigen concentration. Helper T cells recognize the concentration of antigen at the current location. When the concentration is higher than the preset threshold, then the pairing is successful and the following activation step is started. The threshold value in TCIA is set to the median fitness value of all cells, and cells are activated if the fitness value at the current location is higher than the median value. For TCIA, this step is to determine the level of the current position fitness value.

2) ACTIVATION

After successful recognition in the previous step, the naive T cells are activated to become active T cells. Active T cells will differentiate into Tc, Tm, Treg and Th. The differentiation process is shown in FIGURE 2. For TCIA, the current T cell generates four new cells at four random locations around it to assist in completing the immune response. The new cell formula is randomly generated as shown in (1). $\vec{X}_k$ is a vector that represents the location of newly generated cell, and the length is the dimension of the question (all the vectors mentioned below are the dimension of the questions). The $n = 1,2,3,4$, which represents the new four cells. $\vec{X}$ is a vector that represents the location of initial T cell. The a in (2) is the step size of generating a new individual, which decreases linearly from 2.1 to 0.1 with the iteration of the algorithm. The $\vec{r}$ is a random vector, and all its elements are

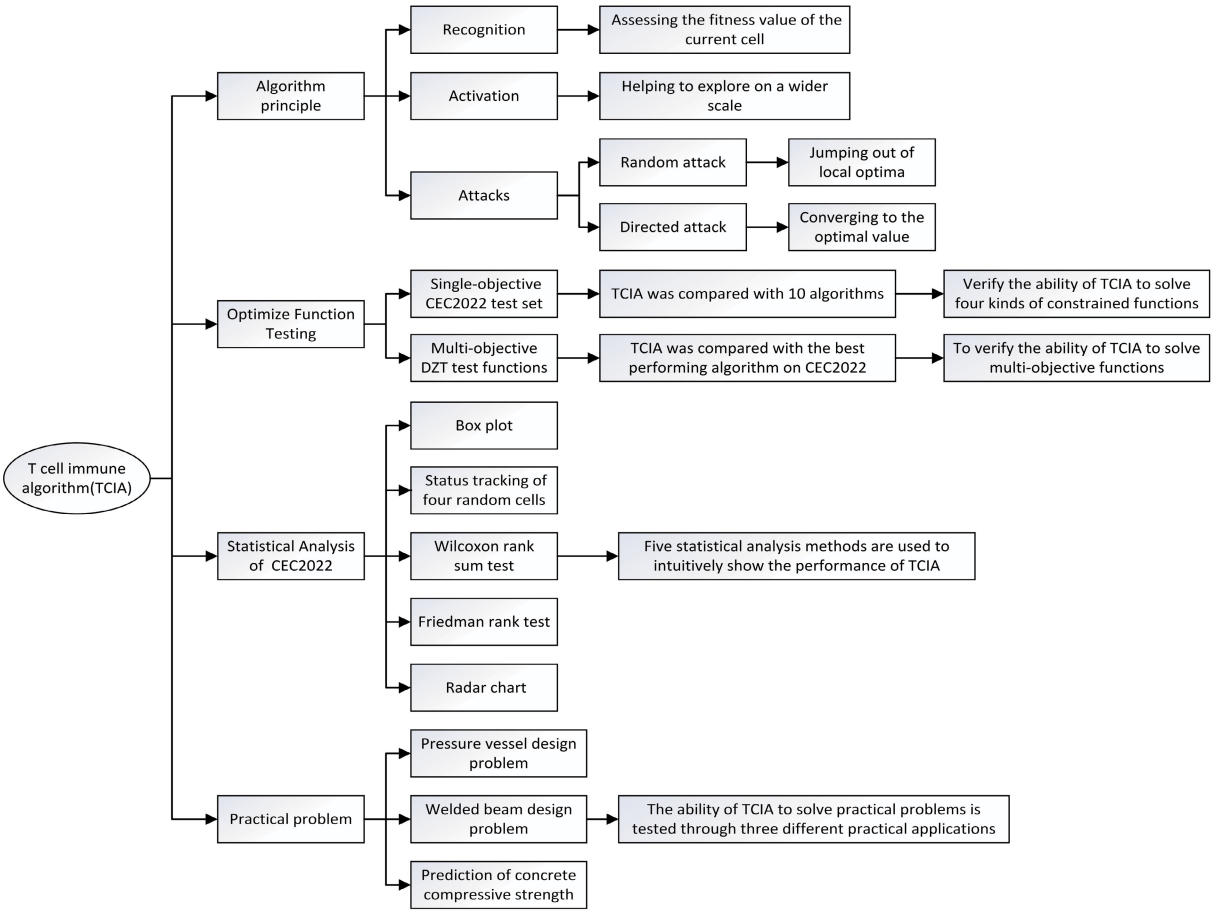


FIGURE 1. Graphical abstract.

random numbers between [0-1]. The t shows the number of algorithm iterations, and M is the upper limit of the number of algorithm iterations. After comparing the initial T cells with the four newly generated T cells, the one closest to the global optimal value is selected from all five cells to replace the current one as a new individual. The cells are activated to differentiate four new cells that can effectively enrich population diversity.

$$\vec{X_{k_n}} = \vec{X} + (a * \vec{r}). * \vec{X} \tag{1}$$

$$a = 2.1 - t * (\frac{2}{M}) \tag{2}$$

3) ATTACK

When the antigen concentration at the current position does not reach the threshold, the initial T cells will not be activated to differentiate, skipping the activation process. Although the concentration of antigen is low, there is still a risk of infection. Therefore, the cytotoxic T cells actively play the role of killing target cells (infected cells or cancer cells). Cytotoxic T cells initiate killing of target cells by releasing cytotoxic particles to induce apoptotic molecules to destroy the cell membrane or nucleus of target cells. The cytotoxic T cells randomly select an initial T cell to attack the target cell at its location. If the antigen concentration of the current position is greater than that of the randomly selected location, (3) is used to attack, otherwis (4) is used.

$$\vec{X_m} = \vec{X} + \vec{r}. * \left(\vec{X_q} - I * \vec{X}\right) \tag{3}$$

$$\vec{X_m} = \vec{X} + \vec{r}. * (I * \vec{X} - \vec{X_q}) \tag{4}$$

$\vec{X_m}$ is a new location after toxic T cells randomly attack a target cell.

$\vec{X}$ is a vector that represents the current location of cytotoxic T cells. The $\vec{r}$ is a vector whose elements are random numbers between [0-1]. $\vec{X_q}$ is the position vector of randomly selected target cells. I is a random number between [1-2]. After comparing the current location of cytotoxic T cells with the new location after random attack, the current location is updated with a better fitness value. Memory T cells preserve the location of the highest antigen concentration. After completion of activation or random attack, memory T cells guide the current T cells closer to them and update the initial T cell location. Update the location formula as shown in (5).

$$X_{t+1} = X_t + w * X_t + 2 * \vec{r} * (X_p - X_t) + 2 * \vec{r} * (X_g - X_t) \tag{5}$$

$$w = 0.9 - t * (0.7/M) \tag{6}$$

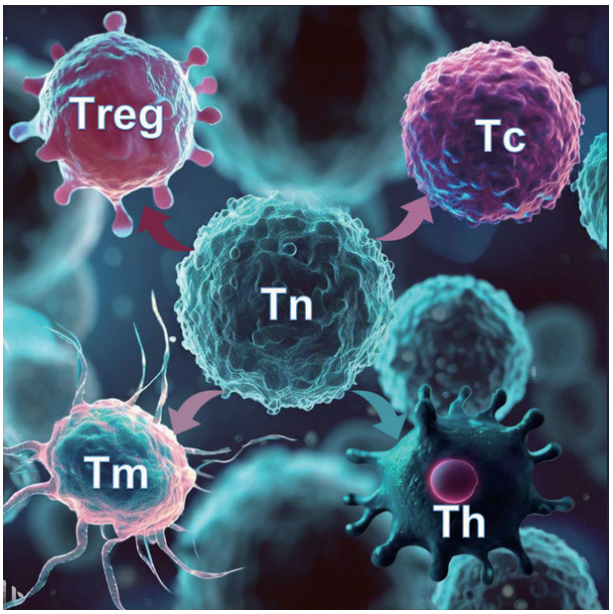


FIGURE 2. T cell differentiation.

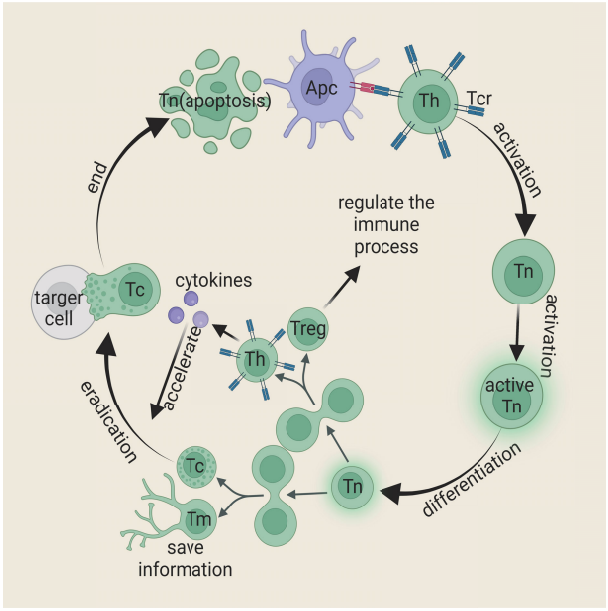


FIGURE 3. T cell immunologic process.

The w in (6) is the front proximity coefficient when updating the location, which decreases linearly from 0.9 to 0.2 with the calculation of the algorithm. The t shows the number of algorithm iterations. And the X_{t+1} in (5) is the new location updated by the current cell location X_t. X_p is the best location in the history of the current cell, X_g is the best location in the whole population.

4) ALGORITHM

TCIA includes three stages: identification, activation and attack. The recognition process is to evaluate the fitness value of the current cell and judge its value attribute. The activation process is explored in a wider area. Attacks are divided into random attacks and directed attacks, random attacks can help the population to jump out of the local optimal, directed attacks can better develop and make use of the current optimal solution. In order to more intuitively understand how TCIA works, its pseudo code is shown as Algorithm 1. The implementation steps of TCIA are shown as the flow chart FIGURE 4.

Algorithm 1 TCIA Algorithm

Input: Max number of iterations (m), number of search cells (n), dimensionality (d) and allowable position boundary
Output: Best score

```
Initialize the TCIA population X_i^j (i = 1, 2, 3 ··· , n) , (j = 1, 2, 3 ··· , d)
while t≤m do
Initialize a and w
Calculate the fitness of search cells
Find the median (med)
    for i≤n do
        if fitness(i)≤med then
 Use equation (1) to activate current cell
        else if fitness(i)>med then
 Use equation (3) or equation (4) to random attack target
        end if
Use equation (5) to guide the current cell towards the global optimal solution
i=i+1
    end for
t=t+1
end while
return Best score
```

III. FUNCTION TESTING AND STATISTICAL ANALYSIS

The TCIA, which simulates the process of T cell immunity, has three behavior patterns, but its effect remains to be tested. To verify the performance of TCIA, all the 11 algorithms in TABLE 1 are tested on the CEC2022 test set. Section III-A analyzes the solution quality and convergence ability of all 11 algorithms on the CEC2022 test set. In Section III-B, in order to further verify the ability of TCIA to solve constrained function problems, TCIA is tested against ASO on three multi-objective test functions. In section III-C, the performance of TCIA is compared and analyzed by using box plot, status tracking of four random cells Wilcoxon rank sum test, Friedman rank test and radar chart.

The TABLE 1 of parameter settings lists the information and independent parameter settings for TCIA and other comparison algorithms. In order to ensure the fairness and authenticity of the experiment, on the premise of ensuring the same computational complexity, the common parameters of each algorithm have been established as follows: the number of individuals is 50, the maximum number of iterations of the algorithm is limited to 1000, and the dimensions

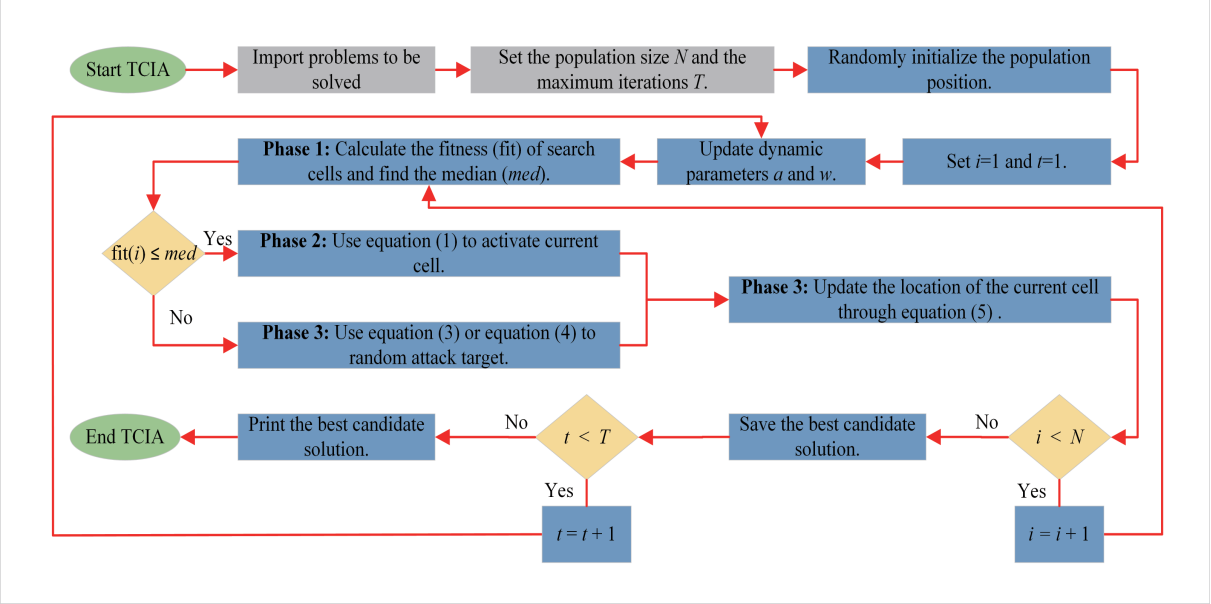


FIGURE 4. Flowchart of the TCIA.

TABLE 1. Compare algorithm information and parameter settings.

Algorithms	Reference	Year	Parameters settings
GWO	[37]	2014	$a = 2 - T * ((2)/Maxiter); A = 2 * a * r - a; C = 2 * r;$
SCA	[38]	2016	$a = 2; r1 = a - t * ((a)/Maxiter);$
WOA	[39]	2016	$a = 0.1 - 0.05 * (T/Maxiter); k = (1 - 0.5 * T/Maxiter);$
BOA	[40]	2019	$a = 2 - T * ((2)/Maxiter); A = 2 * a * r - a; C = 2 * r;$
HHO	[41]	2019	$\alpha = 0.1; \delta = 0.1;$
ASO	[42]	2019	$Depthweight = 50; Multiplierweight = 0.2;$
AO	[43]	2021	$a = 2 - T * ((2)/Maxiter); A = 2 * a * r - a; C = 2 * r;$
BWO	[44]	2022	$p = 0.8; powerexponent = 0.1; sensorymodality = 0.01;$
POA	[45]	2022	$a = 2 - T * ((2)/Maxiter); a2 = -1 + T * ((-1)/Maxiter);$
OOA	[46]	2023	$I = round(1 + rand);$
TCIA	present	present	$a = 2.1 - T * (\frac{2}{Maxiter}); w = 0.9 - T * (0.7/Maxiter).$

are 10 and 20 respectively. The difficulty and time of solving the problem increase with the dimension rise. To avoid the occurrence of special cases, each function is run 30 times independently to calculate its average value. All algorithms are written in MatLab 2021b. The model of the test equipment is a 3.2 GHz PC with 16 GB memory, which runs Microsoft Windows 11 operating system.

A. SINGLE-OBJECTIVE FUNCTION TESTING

The CEC2022 test set is a new single-objective test set, which can effectively evaluate the performance of the algorithm. The CEC2022 test set include 12 single-objective functions of four types. The CEC2022 test set contains a large number of local optimal solutions, which is easy to mislead the algorithm into local optimum. This test set requires high performance of the algorithms. The functions are complex and extremely challenging. Therefore, this paper uses the CEC2022 test set to verify the ability of TCIA to solve constrained functions. The CEC2022 test set includes four types of functions: Unimodal Function, Basic Functions, Hybrid Functions, and Composition Functions. The detailed information of 12 functions are enumerated in TABLE 2.

1) TEST RESULT

To verify the performance of TCIA, all the 11 algorithms are comparatively tested by CEC2022 test set, and test functions dimensions select 10 and 20 [47]. The results of the comparison are shown in TABLE 3 and TABLE 4. The average value of the winning algorithm in each function is shown in bold. The experimental results show that compared with the comparison algorithm, the TCIA can better solve unimodal functions, basic functions, mixed functions and combinatorial functions. To sum up, TCIA performs better than other contrast algorithms in terms of solving constraint functions. Therefore, the performance of TCIA is satisfactory and has strong competitiveness.

2) COMPARISON OF CONVERGENCE ABILITY

The FIGURE 5 and FIGURE 6 draw the convergence curves of all the 11 algorithms in TABLE 1 on 10-dimensional

TABLE 2. The basic information of CEC2022 benchmark functions.

Type	No.	Function	f_{opt}
Unimodal function	f_1	Shifted and full Rotated Zakharov Function	300
Basic functions	f_2	Shifted and full Rotated Rosenbrock¡¯s Function	400
	f_3	Shifted and full Rotated Expanded Schaffer's f_6 Function	600
	f_4	Shifted and full Rotated Non-Continuous Rastrigin¡¯s Function	800
	f_5	Shifted and full Rotated Lévy Function	900
Hybrid functions	f_6	Hybrid Function 1 (N = 3)	1800
	f_7	Hybrid Function 2 (N = 6)	2000
	f_8	Hybrid Function 3 (N = 5)	2200
Composition Functions	f_9	Composition Function 1 (N = 5)	2300
	f_{10}	Composition Function 2 (N = 4)	2400
	f_{11}	Composition Function 3 (N = 5)	2600
	f_{12}	Composition Function 4 (N = 6)	2700
*Search range: [-100,100], Dimensions: D = 10 and 20			

and 20-dimensional problems, respectively. In general, compared with other algorithms, the TCIA proposed in this paper shows obvious advantages in convergence speed and accuracy. Notably, the TCIA achieves superior performance at the early stages of iteration and outperforms the other contrast algorithms. Furthermore, its search ability can be maintained for a long time.

B. MULTI-OBJECTIVE FUNCTION TESTING

Compared with single-objective functions, multiple factors should be considered when solving multi-objective functions [48]. Moreover, the sub-objectives of the multi-objective optimization function are contradictory with each other, so it is necessary to coordinate all the sub-objectives and choose a reasonable set of solutions in a compromise [49]. In order to verify the ability of TCIA to solve multi-objective functions, MOASO, the multi-objective version of the ASO algorithm that performs best except TCIA in the CEC2022 test, is compared with TCIA on three multi-objective functions ZDT1, ZDT2 and ZDT6. The comparison plots of the test results are shown in FIGURE 7-9. TPF in the picture is the Pareto-optimal front. The higher the coincidence degree between the optimization result of the algorithm and TPF, the more effective the solution given by the algorithm. From the results, it can be clearly found that TCIA also has high competitiveness in solving multi-objective function problems.

C. STATISTICAL ANALYSIS

To verify the stability of TCIA, this section selects five methods to analyze it. They are box plot, status tracking of four random cells Wilcoxon rank sum test, Friedman rank test and radar chart.

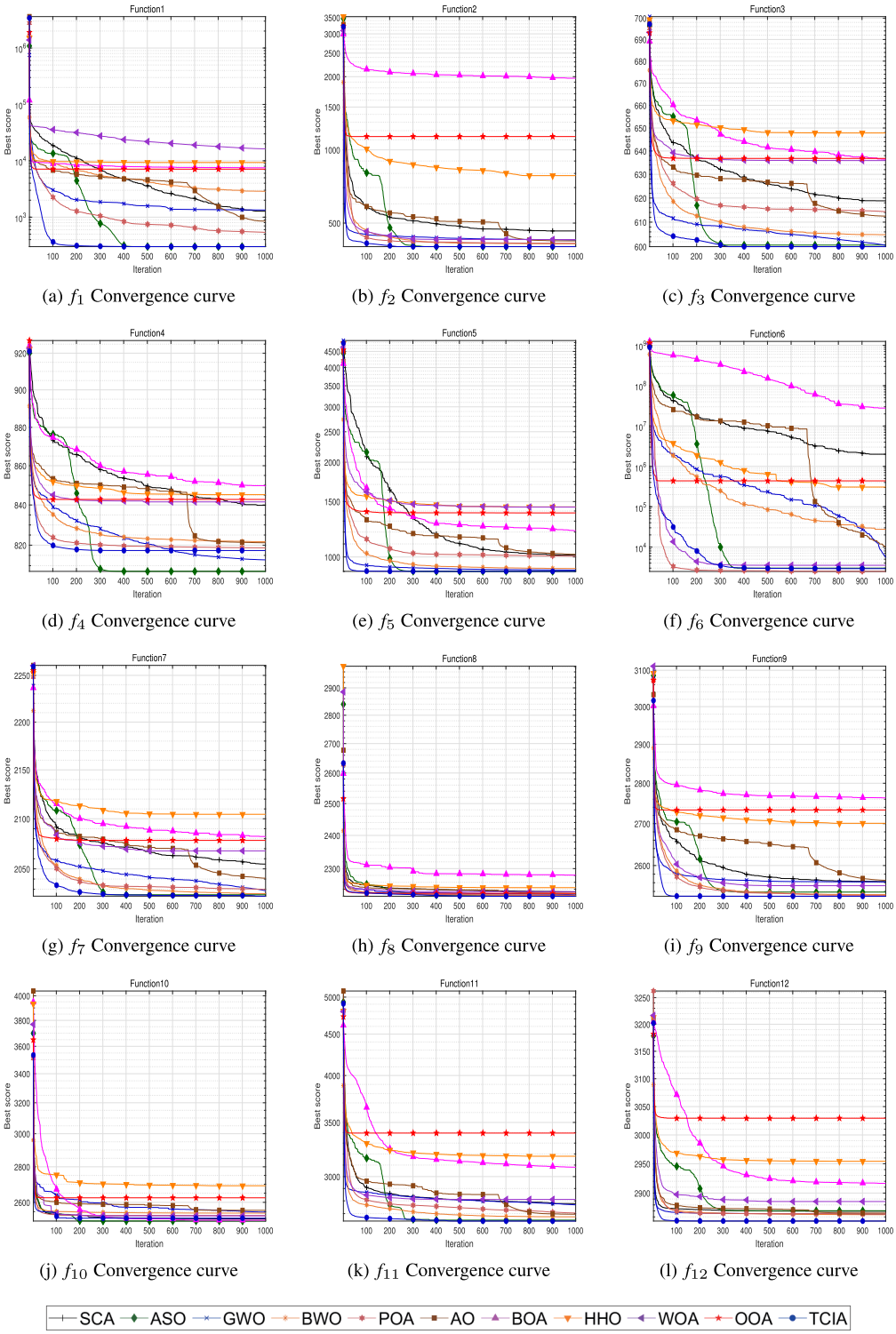
1) BOX PLOT

The box plot is one of the important tools for data analysis which is used to show the dispersion of a set of data. It is often used in outlier detection [50]. FIGURE 10 and FIGURE 11 show the stability comparison of TCIA and the other 10 comparison algorithms on CEC2022 test set in two kinds of dimensions. The long red horizontal line is the midline, which represents the median of the 30 best fitness values for 30 cycles. The top black line represents the upper limit of the fitness range, and the lowest black line represents the lower limit of the fitness range. More than these two black lines indicate that there is an outlier, marked with a red vertical bar. The upper and lower blue lines represent 75% and 25% positions in the fitness matrix, respectively. To put it simply, the closer all the lines are, the smaller the difference between each loop is, the more stable it is. When the red horizontal line is closer to the horizontal coordinate axis, the quality of the solution found by the algorithm in the current function is higher.

Parameter setting: Population size 50; Dimension 10 and 30; Maximum iterations: 1000; Number of cycles: 30.

2) STATUS TRACKING OF FOUR RANDOM CELLS

In this section, four cells were randomly selected to track their location and fitness. The qualitative analysis results of TCIA are displayed as FIGURE 12 and FIGURE 13. The first picture draws a 3-dimensional image of the objective function. The historical location of four cells were shown in the second picture. Obviously, the random four cells can gather near the global optimal solution. From the third picture, we can see that the average adaptation of the four cells tends to be the best with the fluctuation. This shows that they are constantly jumping out of the local optimal solution and the global optimal solution. The fourth and fifth pictures show the trajectories of four cells in the objective function one-dimensional and two-dimensional space. The trajectory curve of the cell exhibits frequent, vigorous, and broad fluctuations at the start of the algorithm?which indicates that the cells are constantly exploring the optimal solution at this time. With the progress of iteration, the frequency and amplitude of the trajectory curve gradually decrease and finally become stable. This shows that the cells gradually find the best position and start to aggregate. The last picture is the



(a) f_1 Convergence curve (b) f_2 Convergence curve (c) f_3 Convergence curve

(d) f_4 Convergence curve (e) f_5 Convergence curve (f) f_6 Convergence curve

(g) f_7 Convergence curve (h) f_8 Convergence curve (i) f_9 Convergence curve

(j) f_{10} Convergence curve (k) f_{11} Convergence curve (l) f_{12} Convergence curve

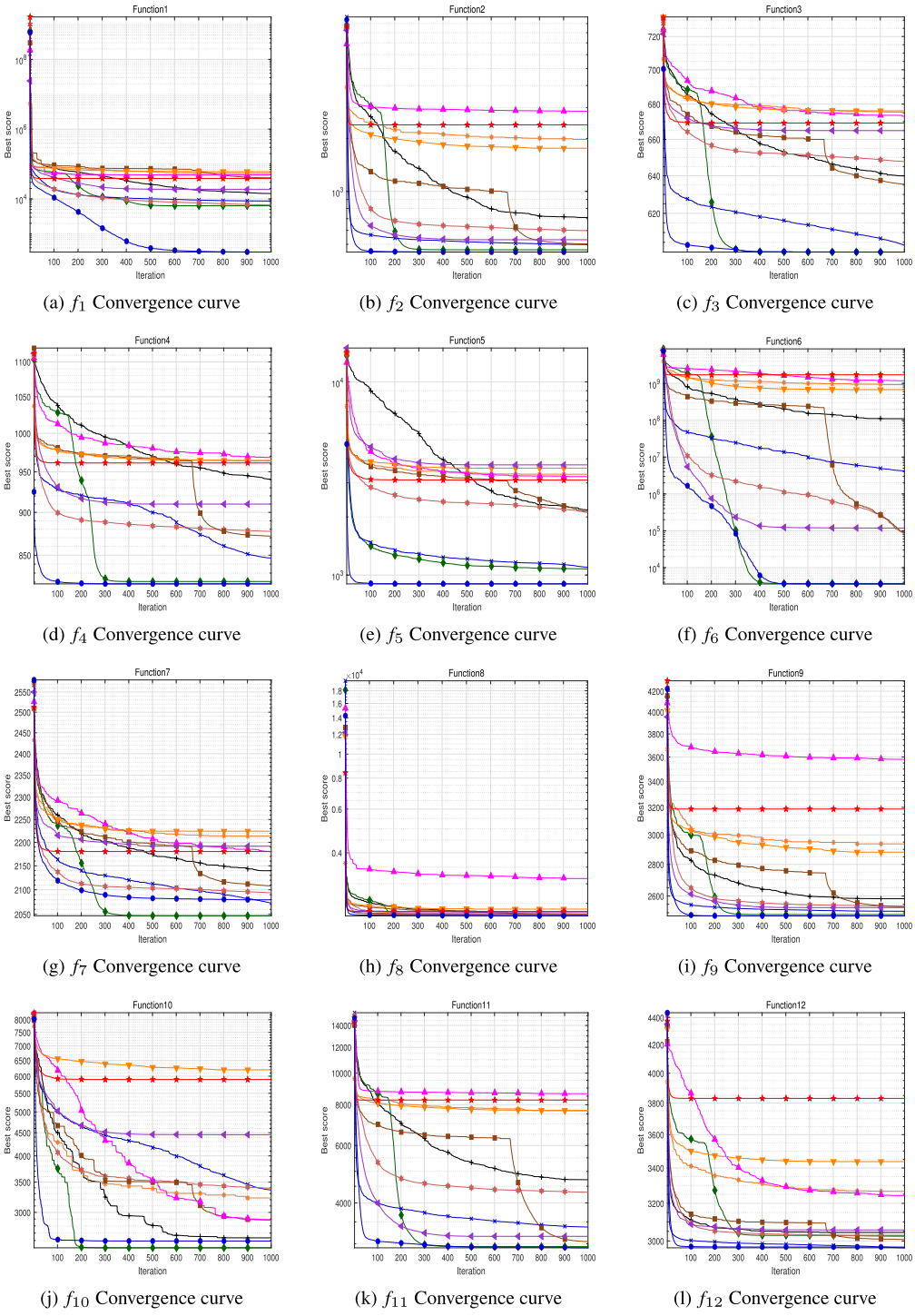
(m) Curve legend

FIGURE 5. Convergence curve on 10 dimensions.

convergence curve of the best individual of the four cells. It is obvious from the FIGURE [12-13] that the proposed TCIA has strong search ability. Parameter setting: Population size 50; Dimension 2; Maximum iterations 100.

3) WILCOXON RANK SUM TEST

In order to show the difference between TCIA and other comparison algorithms, Wilcoxon rank sum test is used to make a pairwise comparison [51].



(a) f_1 Convergence curve (b) f_2 Convergence curve (c) f_3 Convergence curve

(d) f_4 Convergence curve (e) f_5 Convergence curve (f) f_6 Convergence curve

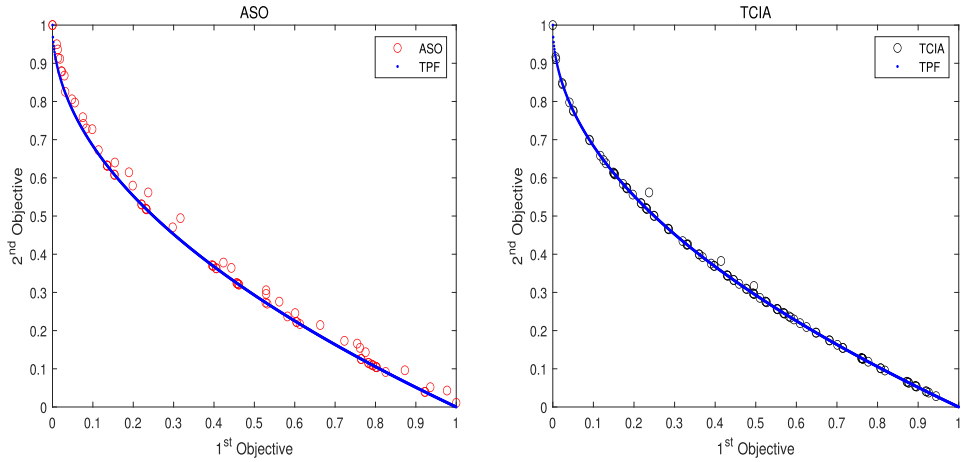
(g) f_7 Convergence curve (h) f_8 Convergence curve (i) f_9 Convergence curve

(j) f_{10} Convergence curve (k) f_{11} Convergence curve (l) f_{12} Convergence curve

(m) Curve legend

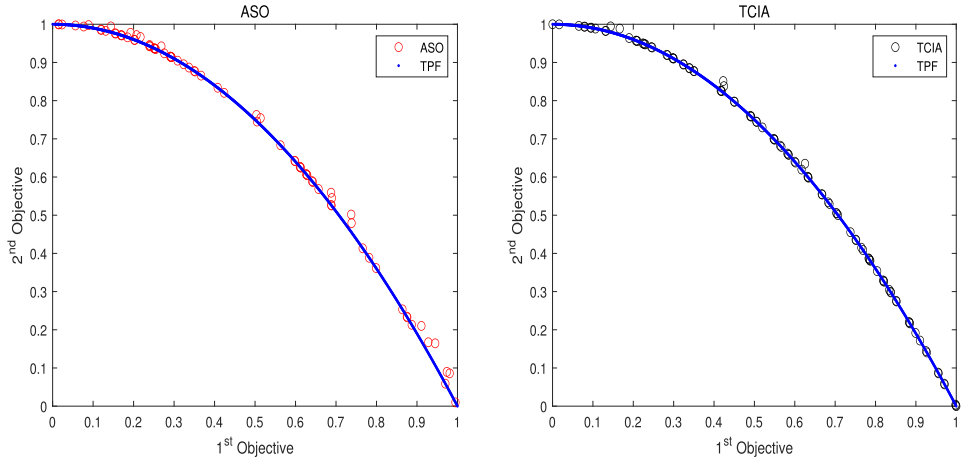
FIGURE 6. Convergence curve on 20 dimensions.

TABLE 5 and TABLE 6 show the test results under CEC2022 benchmark functions on dimension 10 and 20. Wilcoxon rank sum test takes the rank sum of the difference between the data to be tested and the hypothetical center as a statistic according to the symbol. The significance level is set to 0.05. The more p-value is less than 0.05, the greater the difference between the two algorithms. If the p-value is greater than 0.05, it ends in a draw. '$+/=/-$' represents the



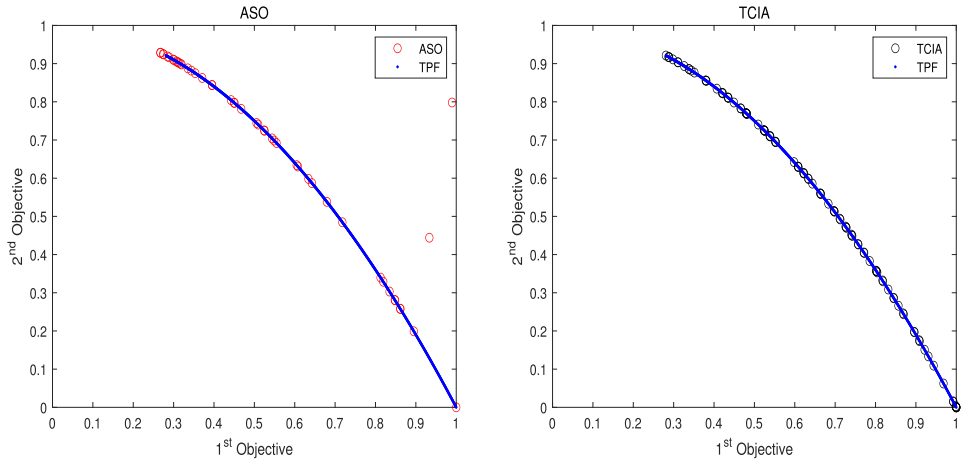
(a) ASO (b) TCIA

FIGURE 7. Results on ZDT1.



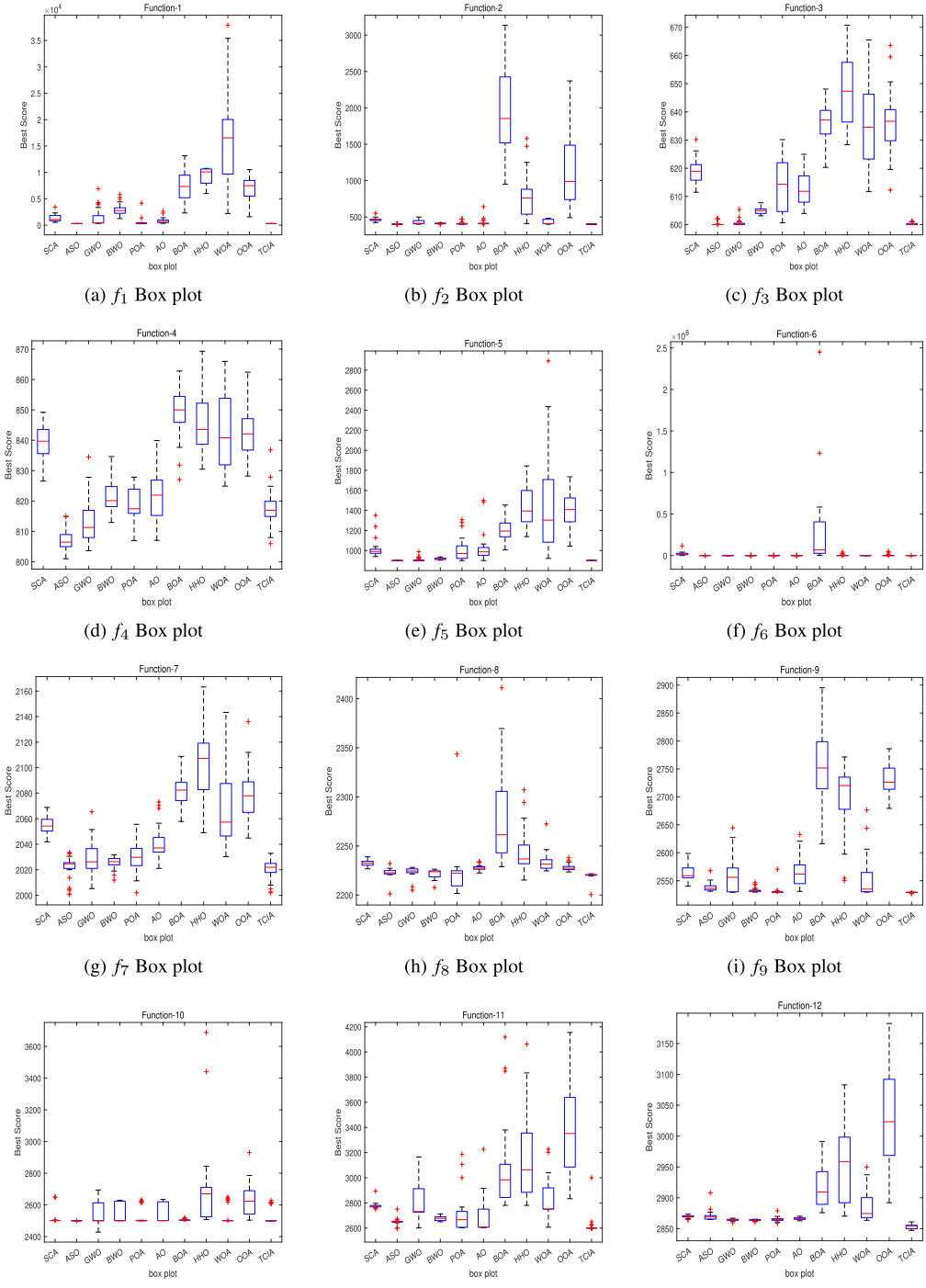
(a) ASO (b) TCIA

FIGURE 8. Results on ZDT2.



(a) ASO (b) TCIA

FIGURE 9. Results on ZDT6.



(a) f_1 Box plot (b) f_2 Box plot (c) f_3 Box plot

(d) f_4 Box plot (e) f_5 Box plot (f) f_6 Box plot

(g) f_7 Box plot (h) f_8 Box plot (i) f_9 Box plot

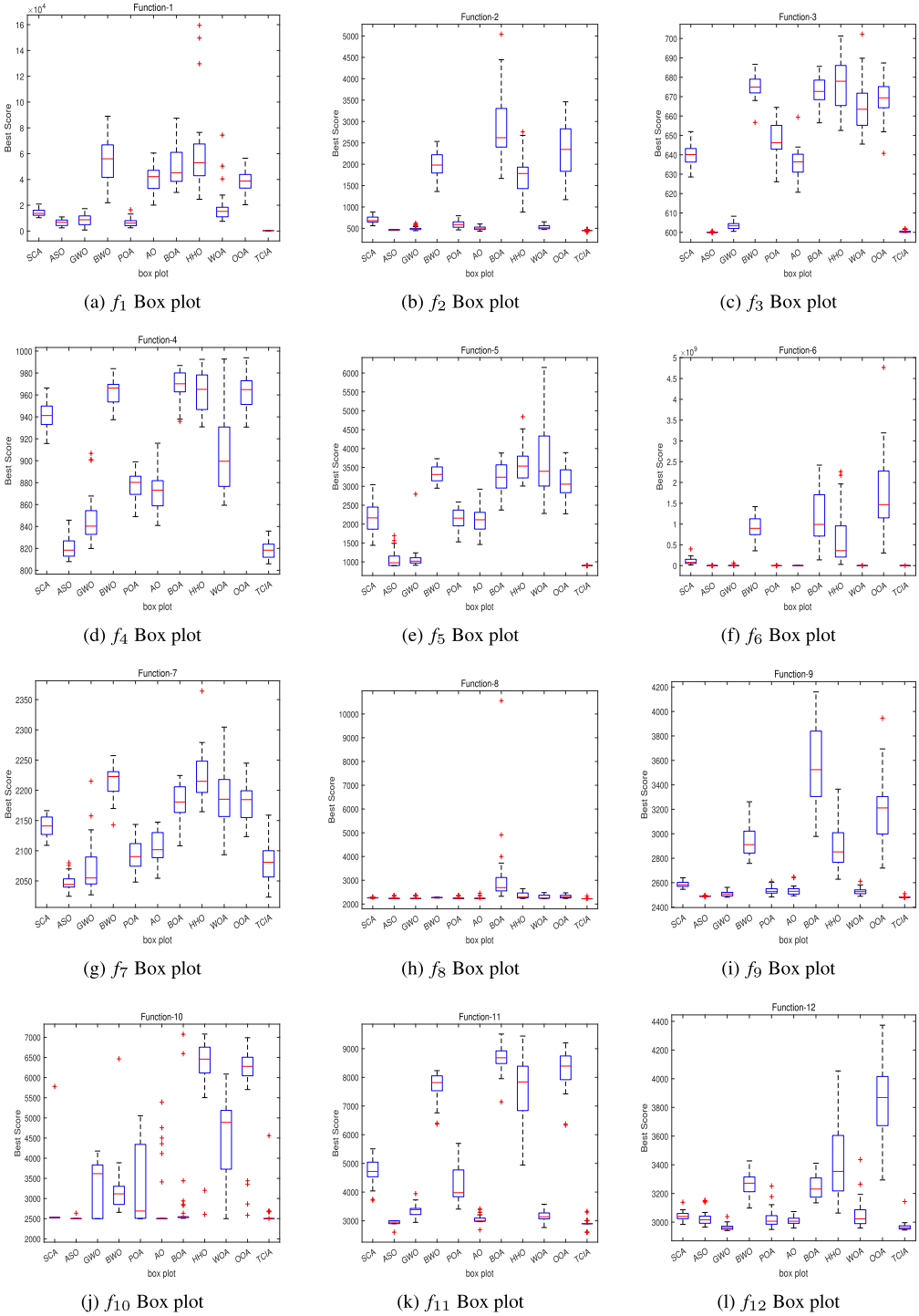
(j) f_{10} Box plot (k) f_{11} Box plot (l) f_{12} Box plot

FIGURE 10. Box plot on 10 dimensions.

number of victories, draws and defeats, respectively. Taking the comparison between TCIA and SCA in 10 dimensions as an example, '12/0/0' shows that TCIA is better than SCA in solving all 12 functions, and there is no draw or failure. It can be seen from the TABLE 5 and TABLE 6 that the difference between TCIA and other comparative algorithms is more prominent.

4) FRIEDMAN RANK TEST

The CEC2022 test set is used to test 11 algorithms in TABLE 1. In 10 dimensions and 20 dimensions, all the 11 algorithms were run 30 times to get the average fitness separately. Friedman rank test [52] is used to compare and analyze all 11 algorithms, the result is shown as TABLE 7 and FIGURE 14. The p-values in dimensions 10 and 20 are



(a) f_1 Box plot (b) f_2 Box plot (c) f_3 Box plot

(d) f_4 Box plot (e) f_5 Box plot (f) f_6 Box plot

(g) f_7 Box plot (h) f_8 Box plot (i) f_9 Box plot

(j) f_{10} Box plot (k) f_{11} Box plot (l) f_{12} Box plot

FIGURE 11. Box plot on 20 dimensions.

1.05E-14 and 4.20E-16, respectively. They are much less than 0.05, so there is a obvious difference between TCIA and other 10 comparison algorithms. The overall ranking of TCIA in all 12 functions in 10 and 20 dimensions is 1.54 and 1.33, respectively. It can be seen intuitively that among all 11 algorithms, TCIA has obvious advantages in solving the problems of 10 dimensions and 20 dimensions.

5) RADAR CHART

The radar chart is a visual tool that allows for the analysis and comparison of multi-index within the same coordinate system. Radar chart is a commonly used approach for conducting comprehensive evaluations, particularly when evaluating objects that can be described using multiple attributes. This method is well-suited for providing an overall and holistic

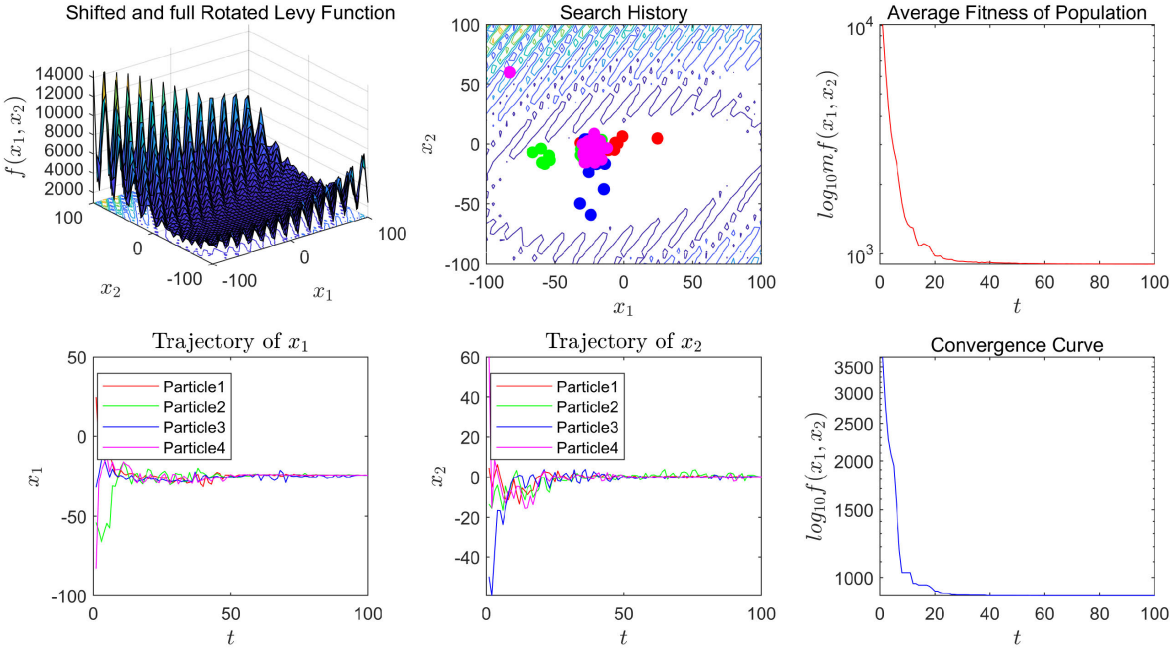


FIGURE 12. Qualitative results of TCIA on f_5.

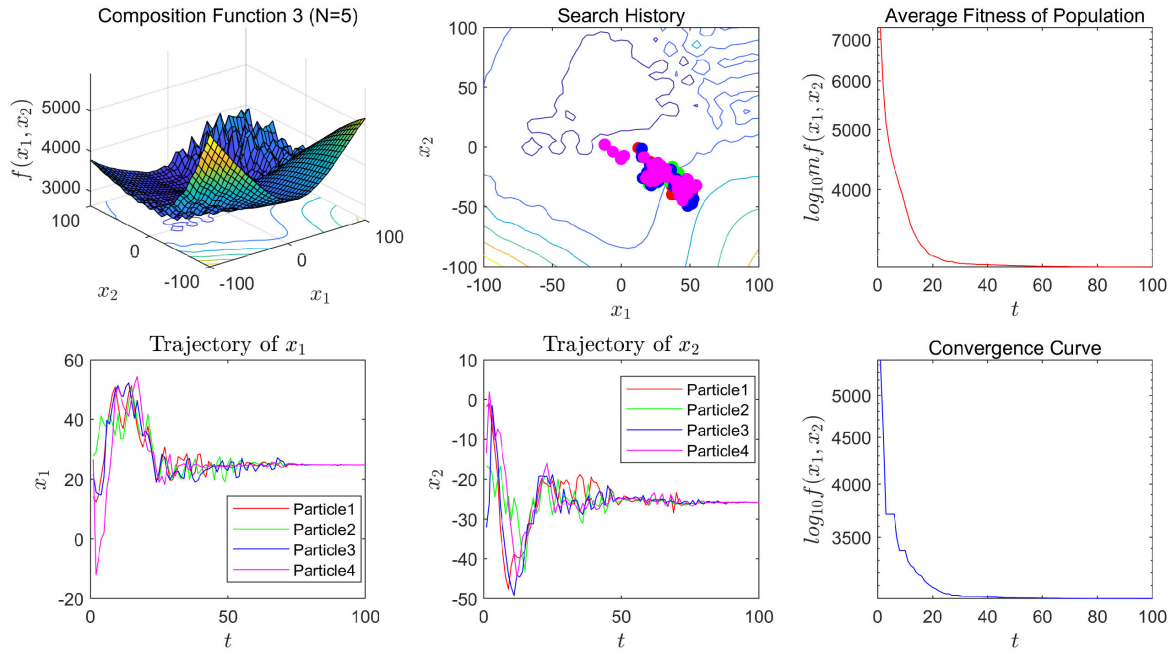


FIGURE 13. Qualitative results of TCIA on f_{11}.

assessment of such objects. It consists of multiple concentric circles and multiple coordinates. The closer the graph of the target is to the center circle, the higher the overall ranking will be. The number of coordinates is the number of attributes used to describe the target. In order to more intuitively show the ranking of TCIA in all 11 algorithms, the radar charts of algorithm rankings are drawn on the 10 and 20 dimensions of CEC2022 respectively. The result is shown in FIGURE 15. As the radar range spreads out from the center, the lower the ranking of the algorithm, the outermost circle is ranked 11th. It can be seen that in all 12 functions, TCIA is at the center of the radar chart. The results show that TCIA is better than other algorithms in solving 12 functions.

TABLE 3. CEC2022 test results on 10-dimension.

Fun	Value	SCA	ASO	GWO	BWO	POA	AO	BOA	HHO	WOA	OOA	TCIA
F1	Mean	1.2745E+03	3.0000E+02	1.3404E+03	2.8756E+03	5.3880E+02	8.5111E+02	7.5116E+03	9.3293E+03	1.6326E+04	7.1117E+03	**3.0000E+02**
	Std	6.4150E+02	4.2626E-14	1.6184E+03	1.1051E+03	7.0944E+02	5.2453E+02	3.0550E+03	1.5264E+03	8.6258E+03	2.1855E+03	2.9856E-14
	Best	5.4255E+02	3.0000E+02	3.5397E+02	1.2648E+03	3.0702E+02	3.4812E+02	2.3211E+03	6.0019E+03	2.2202E+03	1.6066E+03	3.0000E+02
	Rank	5	2	6	7	3	4	9	10	11	8	1
F2	Mean	4.6409E+02	4.0218E+02	4.2594E+02	4.1118E+02	4.1275E+02	4.2229E+02	1.9702E+03	7.8349E+02	4.2784E+02	1.1344E+03	**4.0132E+02**
	Std	2.4199E+01	1.7214E+00	2.5142E+01	3.5849E+00	2.0817E+01	4.4929E+01	6.4931E+02	2.8603E+02	3.0809E+01	4.9606E+02	1.4455E+00
	Best	4.2256E+02	4.0000E+02	4.0039E+02	4.0162E+02	4.0003E+02	4.0006E+02	9.5128E+02	4.0923E+02	4.0013E+02	4.9157E+02	4.0000E+02
	Rank	8	2	6	3	4	5	11	9	7	10	1
F3	Mean	6.1885E+02	6.0067E+02	6.0052E+02	6.0486E+02	6.1432E+02	6.1249E+02	6.3646E+02	6.4782E+02	6.3588E+02	6.3680E+02	**6.0003E+02**
	Std	4.0305E+00	1.0775E+00	1.0685E+00	1.0698E+00	9.6701E+00	5.4882E+00	6.4094E+00	1.1435E+01	1.3774E+01	1.1025E+01	2.0045E-03
	Best	6.1149E+02	6.0000E+02	6.0005E+02	6.0315E+02	6.0068E+02	6.0396E+02	6.2032E+02	6.2835E+02	6.1175E+02	6.1226E+02	6.0000E+02
	Rank	7	3	2	4	6	5	9	11	8	10	1
F4	Mean	8.3981E+02	**8.0723E+02**	8.1282E+02	8.2175E+02	8.1867E+02	8.2118E+02	8.4945E+02	8.4519E+02	8.4155E+02	8.4279E+02	8.1738E+02
	Std	5.7153E+00	3.2523E+00	7.2099E+00	5.1092E+00	5.2709E+00	7.5720E+00	7.9061E+00	9.6616E+00	1.1832E+01	7.7615E+00	6.1872E+00
	Best	8.2656E+02	8.0099E+02	8.0365E+02	8.1294E+02	8.0698E+02	8.0705E+02	8.2708E+02	8.3046E+02	8.2491E+02	8.2823E+02	8.0597E+02
	Rank	7	1	2	6	4	5	11	10	8	9	3
F5	Mean	1.0135E+03	9.0007E+02	9.0863E+02	9.1914E+02	1.0058E+03	1.0197E+03	1.2119E+03	1.4424E+03	1.4410E+03	1.3804E+03	**9.0003E+02**
	Std	8.6253E+01	6.8769E-02	1.8700E+01	8.6827E+00	1.1346E+02	1.3978E+02	1.0718E+02	1.9503E+02	4.7517E+02	1.8806E+02	4.1728E-02
	Best	9.4082E+02	9.0000E+02	9.0010E+02	9.0442E+02	9.0029E+02	9.0035E+02	1.0068E+03	1.1390E+03	9.2189E+02	1.0449E+03	9.0000E+02
	Rank	6	2	3	4	5	7	8	11	10	9	1
F6	Mean	2.0009E+06	3.0035E+03	5.5112E+03	2.7500E+04	**2.4834E+03**	1.0137E+04	2.7858E+07	3.0742E+05	3.5273E+03	4.3738E+05	2.9195E+03
	Std	2.1404E+06	1.5267E+03	2.5054E+03	2.1326E+04	1.3577E+03	6.1204E+03	4.9322E+07	8.4490E+05	1.9235E+03	1.1833E+06	1.4793E+03
	Best	1.9151E+04	1.8797E+03	2.0380E+03	2.7960E+03	1.8314E+03	2.8591E+03	4.0323E+05	2.4396E+03	1.8915E+03	1.8999E+03	1.8169E+03
	Rank	10	3	5	7	1	6	11	8	4	9	2
F7	Mean	2.0547E+03	2.0247E+03	2.0285E+03	2.0257E+03	2.0298E+03	2.0408E+03	2.0820E+03	2.1045E+03	2.0678E+03	2.0783E+03	**2.0232E+03**
	Std	6.7447E+00	1.2528E+01	1.2868E+01	4.4982E+00	1.1818E+01	1.2255E+01	1.2146E+01	2.9452E+01	2.9284E+01	2.1819E+01	1.0638E+01
	Best	2.0419E+03	2.0036E+03	2.0053E+03	2.0119E+03	2.0020E+03	2.0211E+03	2.0578E+03	2.0490E+03	2.0303E+03	2.0449E+03	2.0020E+03
	Rank	7	2	4	3	5	6	10	11	8	9	1
F8	Mean	2.2325E+03	2.2228E+03	2.2236E+03	2.2219E+03	2.2230E+03	2.2277E+03	2.2808E+03	2.2445E+03	2.2329E+03	2.2281E+03	**2.2199E+03**
	Std	2.6807E+00	4.8120E+00	5.0763E+00	4.3546E+00	2.4258E+01	2.9985E+00	5.1038E+01	2.0471E+01	9.2896E+00	3.4728E+00	3.7097E+00
	Best	2.2267E+03	2.2012E+03	2.2050E+03	2.2076E+03	2.2016E+03	2.2224E+03	2.2289E+03	2.2152E+03	2.2246E+03	2.2235E+03	2.2004E+03
	Rank	8	3	5	2	4	6	11	10	9	7	1
F9	Mean	2.5628E+03	2.5385E+03	2.5620E+03	2.5328E+03	2.5311E+03	2.5657E+03	2.7634E+03	2.7008E+03	2.5532E+03	2.7335E+03	**2.5290E+03**
	Std	1.3106E+01	7.6089E+00	3.2919E+01	4.2752E+00	7.4751E+00	2.6998E+01	7.0322E+01	5.8269E+01	3.6584E+01	2.6398E+01	5.0241E-01
	Best	2.5403E+03	2.5311E+03	2.5293E+03	2.5296E+03	2.5293E+03	2.5309E+03	2.6162E+03	2.5504E+03	2.5296E+03	2.6794E+03	2.5276E+03
	Rank	7	4	6	3	2	8	11	9	5	10	1
F10	Mean	2.5117E+03	**2.5004E+03**	2.5476E+03	2.5421E+03	2.5290E+03	2.5570E+03	2.5044E+03	2.6915E+03	2.5276E+03	2.6246E+03	2.5163E+03
	Std	3.7256E+01	7.2592E-02	6.4709E+01	5.9318E+01	5.2233E+01	6.0896E+01	3.7553E+00	2.5725E+02	5.4488E+01	1.0170E+02	4.1434E+01
	Best	2.5008E+03	2.5003E+03	2.4286E+03	2.5004E+03	2.5002E+03	2.5008E+03	2.5009E+03	2.5079E+03	2.5004E+03	2.5039E+03	2.5002E+03
	Rank	3	1	8	7	6	9	2	11	5	10	4
F11	Mean	2.7782E+03	2.6501E+03	2.7683E+03	2.6755E+03	2.7045E+03	2.6953E+03	3.0781E+03	3.1789E+03	2.8112E+03	3.3955E+03	**2.6434E+03**
	Std	2.4183E+01	1.0598E+02	1.4869E+02	1.7414E+01	1.4747E+02	1.3027E+02	3.3383E+02	3.5642E+02	1.6356E+02	3.8380E+02	9.8967E+01
	Best	2.7504E+03	2.6000E+03	2.6008E+03	2.6503E+03	2.6024E+03	2.6038E+03	2.7801E+03	2.7801E+03	2.6064E+03	2.8337E+03	2.6000E+03
	Rank	7	2	6	3	5	4	9	10	8	11	1
F12	Mean	2.8695E+03	2.8703E+03	2.8641E+03	2.8637E+03	2.8652E+03	2.8665E+03	2.9165E+03	2.9542E+03	2.8857E+03	3.0294E+03	**2.8550E+03**
	Std	1.9994E+00	8.0850E+00	1.4339E+00	9.9253E-01	3.4019E+00	2.0417E+00	3.0649E+01	6.3088E+01	2.2775E+01	7.7942E+01	1.8733E+00
	Best	2.8645E+03	2.8647E+03	2.8596E+03	2.8611E+03	2.8586E+03	2.8627E+03	2.8759E+03	2.8702E+03	2.8636E+03	2.8921E+03	2.8474E+03
	Rank	6	7	3	2	4	5	9	10	8	11	1

TABLE 4. CEC2022 test results on 20-dimension.

Fun	Value	SCA	ASO	GWO	BWO	POA	AO	BOA	HHO	WOA	OOA	TCIA
F1	Mean	1.4353E+04	6.3878E+03	8.7257E+03	5.4701E+04	6.7133E+03	4.0310E+04	4.8970E+04	6.0739E+04	1.8659E+04	3.8724E+04	**3.0079E+02**
	Std	2.8823E+03	2.3865E+03	4.3359E+03	1.7692E+04	3.1726E+03	9.8132E+03	1.4717E+04	3.2370E+04	1.3890E+04	8.2843E+03	5.6056E-01
	Best	1.0320E+04	2.4617E+03	7.9703E+02	2.1948E+04	2.5135E+03	2.0097E+04	2.9955E+04	2.4649E+04	7.5470E+03	2.0400E+04	3.0006E+02
	Rank	5	2	4	10	3	8	9	11	6	7	1
F2	Mean	7.1053E+02	4.6524E+02	4.9880E+02	1.9759E+03	5.9812E+02	5.0256E+02	2.8551E+03	1.7609E+03	5.3091E+02	2.3896E+03	**4.5182E+02**
	Std	8.7391E+01	7.2331E+00	4.4837E+01	2.9948E+02	8.0963E+01	4.1367E+01	7.3984E+02	4.5882E+02	5.1048E+01	6.0956E+02	1.3873E+01
	Best	5.6044E+02	4.5381E+02	4.4521E+02	1.3624E+03	4.6384E+02	4.3362E+02	1.6675E+03	8.8290E+02	4.7348E+02	1.1722E+03	4.0667E+02
	Rank	7	2	3	9	6	4	11	8	5	10	1
F3	Mean	6.3994E+02	6.0004E+02	6.0342E+02	6.7536E+02	6.4772E+02	6.3547E+02	6.7303E+02	6.7615E+02	6.6493E+02	6.6919E+02	**6.0004E+02**
	Std	6.2727E+00	1.4303E-01	1.9407E+00	6.0182E+00	9.0196E+00	7.8141E+00	7.0472E+00	1.2187E+01	1.2662E+01	9.4721E+00	1.1765E-01
	Best	6.2856E+02	6.0000E+02	6.0050E+02	6.5666E+02	6.2610E+02	6.2069E+02	6.5650E+02	6.5256E+02	6.4552E+02	6.4072E+02	6.0000E+02
	Rank	5	2	2	10	6	4	9	11	7	8	1
F4	Mean	9.4090E+02	8.2073E+02	8.4684E+02	9.6390E+02	8.7751E+02	8.7199E+02	9.6838E+02	9.6452E+02	9.0998E+02	9.6133E+02	**8.1864E+02**
	Std	1.1829E+01	9.5078E+00	2.1627E+01	1.1702E+01	1.3449E+01	1.7217E+01	1.3055E+01	1.8885E+01	3.7698E+01	1.5592E+01	7.6504E+00
	Best	9.1573E+02	8.0796E+02	8.1995E+02	9.3741E+02	8.4897E+02	8.4093E+02	9.3594E+02	9.3088E+02	8.5937E+02	9.3070E+02	8.0590E+02
	Rank	7	2	3	9	5	4	11	10	6	8	1
F5	Mean	2.1730E+03	9.6556E+02	9.8768E+02	3.3245E+03	2.1417E+03	2.1127E+03	3.2278E+03	3.5737E+03	3.7185E+03	3.1029E+03	**9.0069E+02**
	Std	4.0688E+02	2.5437E+02	3.3340E+02	2.2042E+02	2.9070E+02	3.7483E+02	3.6752E+02	4.3232E+02	9.8490E+02	4.1321E+02	4.0216E-03
	Best	1.4379E+03	9.0000E+02	9.0925E+02	2.9514E+03	1.5290E+03	1.4605E+03	2.3728E+03	3.0079E+03	2.2818E+03	2.2747E+03	9.0000E+02
	Rank	6	2	3	9	5	4	8	10	11	7	1
F6	Mean	1.1023E+08	3.7081E+03	4.2388E+06	9.2351E+08	8.2538E+04	9.2605E+04	1.1763E+09	6.7550E+08	1.1860E+05	1.7088E+09	**3.6009E+03**
	Std	8.5636E+07	2.5535E+03	1.2389E+07	2.8540E+08	2.4849E+05	6.0906E+04	6.9206E+08	7.1262E+08	1.2163E+05	9.8745E+08	2.0978E+03
	Best	1.4453E+07	1.9287E+03	2.8332E+03	3.5641E+08	2.3967E+03	1.9290E+04	1.3796E+08	2.8896E+07	4.5924E+03	3.0384E+08	1.9138E+03
	Rank	7	2	6	9	3	4	10	8	5	11	1
F7	Mean	2.1395E+03	**2.0470E+03**	2.0726E+03	2.2137E+03	2.0937E+03	2.1080E+03	2.1806E+03	2.2244E+03	2.1918E+03	2.1804E+03	2.0794E+03
	Std	1.7208E+01	1.4499E+01	4.1520E+01	2.5401E+01	2.5366E+01	2.6100E+01	2.8689E+01	4.1341E+01	5.2118E+01	3.0939E+01	3.2205E+01
	Best	2.1089E+03	2.0247E+03	2.0271E+03	2.1427E+03	2.0482E+03	2.0545E+03	2.1082E+03	2.1646E+03	2.0934E+03	2.1234E+03	2.0233E+03
	Rank	6	1	2	10	4	5	8	11	9	7	3
F8	Mean	2.2634E+03	2.2544E+03	2.2529E+03	2.2706E+03	2.2534E+03	2.2502E+03	3.1489E+03	2.3710E+03	2.3087E+03	2.3197E+03	**2.2265E+03**
	Std	1.3612E+01	5.0236E+01	5.0221E+01	1.1631E+01	4.3184E+01	4.9437E+01	1.5080E+03	1.3957E+02	7.6323E+01	7.0102E+01	2.1898E+01
	Best	2.2418E+03	2.2222E+03	2.2237E+03	2.2501E+03	2.2261E+03	2.2269E+03	2.3272E+03	2.2348E+03	2.2297E+03	2.2351E+03	2.2203E+03
	Rank	6	5	3	7	4	2	11	10	8	9	1
F9	Mean	2.5834E+03	2.4896E+03	2.5090E+03	2.9376E+03	2.5355E+03	2.5382E+03	3.5801E+03	2.8814E+03	2.5304E+03	3.1875E+03	**2.4818E+03**
	Std	2.4880E+01	2.4573E+00	1.9903E+01	1.1844E+02	2.9269E+01	4.1539E+01	3.5075E+02	1.6181E+02	2.8839E+01	2.7144E+02	5.3913E+00
	Best	2.5458E+03	2.4849E+03	2.4812E+03	2.7579E+03	2.4838E+03	2.4927E+03	2.9782E+03	2.6282E+03	2.4890E+03	2.7194E+03	2.4808E+03
	Rank	7	2	3	9	5	6	11	8	4	10	1
F10	Mean	2.6355E+03	**2.5051E+03**	3.3626E+03	3.2290E+03	3.3934E+03	2.8856E+03	2.8871E+03	6.1978E+03	4.4550E+03	5.8996E+03	2.5929E+03
	Std	5.9412E+02	2.4656E+01	6.2371E+02	7.0214E+02	1.0064E+03	8.2832E+02	1.0917E+03	9.7414E+02	1.1773E+03	1.1731E+03	3.7667E+02
	Best	2.5161E+03	2.5004E+03	2.5004E+03	2.6576E+03	2.5012E+03	2.5007E+03	2.5086E+03	2.6091E+03	2.5007E+03	2.5851E+03	2.5003E+03
	Rank	3	1	7	6	8	4	5	11	9	10	2
F11	Mean	4.7143E+03	2.9400E+03	3.3758E+03	7.6632E+03	4.3163E+03	3.0468E+03	8.6488E+03	7.6843E+03	3.1607E+03	8.2693E+03	**2.9149E+03**
	Std	4.7811E+02	8.1368E+01	2.1694E+02	5.2880E+02	6.6515E+02	1.4675E+02	4.5562E+02	1.1300E+03	2.0257E+02	6.9664E+02	1.5103E+02
	Best	3.7177E+03	2.6000E+03	2.9446E+03	6.3687E+03	3.4066E+03	2.6849E+03	7.1454E+03	4.9441E+03	2.7639E+03	6.3296E+03	2.6000E+03
	Rank	7	2	5	8	6	3	11	9	4	10	1
F12	Mean	3.0442E+03	3.0290E+03	**2.9650E+03**	3.2668E+03	3.0229E+03	3.0080E+03	3.2432E+03	3.4373E+03	3.0572E+03	3.8310E+03	2.9688E+03
	Std	3.0364E+01	5.2573E+01	2.1817E+01	8.0952E+01	6.4455E+01	2.4888E+01	8.0552E+01	2.7506E+02	1.0081E+02	2.9256E+02	3.5416E+01
	Best	2.9842E+03	2.9659E+03	2.9414E+03	3.0983E+03	2.9491E+03	2.9597E+03	3.1347E+03	3.0634E+03	2.9595E+03	3.2941E+03	2.9451E+03
	Rank	6	5	1	9	4	3	8	10	7	11	2

In the CEC2022 test set, TCIA ranks first in 9 out of 12 functions each in 10 and 20 dimensions. Convergence FIGURE 5 and FIGURE 6, the results of CEC2022 test illustrate that TCIA is highly convergent. Graphs of test results for three multi-objective functions illustrate that TCIA is also well suited for solving multi-objective problems. The results of box plot analysis demonstrate the strong stability of TCIA in solving constrained function problems. The state tracking results of four random cells show that TCIA has the ability of constantly jumping out of the local optimum and converging to the global optimum. The uniqueness of TCIA is accurately demonstrated after Wilcoxon rank sum test. The results of Friedman rank test and radar chart show that TCIA is highly competitive in solving the CEC2022 test set problems, ranking first compared with other 10 algorithms.

IV. APPLICATION OF ENGINEERING DESIGN PROBLEMS

The strong competitiveness of TCIA in solving the problem of constrained functions has been demonstrated in the third section. To test the ability of TCIA to solve practical problems, it is applied to three kinds of engineering problems. The performance of the TCIA has been compared with 10 optimization algorithms which are known for their strong competitiveness in engineering design problems. To ensure fairness, the initialization parameters of all 11 algorithms are as follows: population size: 50; maximum number of iterations: 1000.

A. PRESSURE VESSEL DESIGN PROBLEM

In this section, TCIA is applied to Pressure vessel design problem. The target of this problem is to minimize the cost associated with manufacturing the pressure vessel, while ensuring that the vessel meets all necessary safety and performance requirements. Both ends of the pressure vessel are capped with a hemispherical cover at one end of the head [53], as shown in FIGURE 16.

There are four design parameters in it, they are thickness of shell $T_s(\alpha_1)$, thickness of head $T_h(\alpha_2)$, inner diameter $R(\alpha_3)$and cylindrical section length $L(\alpha_4)$ of the container [54]. The values of α_1 and α_2 are based on increments of 0.0625 inches, while α_3 and α_4 can take on any continuous value. The problem has four constraints, three of them are linear and the other is nonlinear [55]. The variables and objective functions of this design problem are shown as equations (7) and (8) respectively, and their constraint equations is shown as (9). The results of TCIA and other 10 algorithms for solving this problem are shown in Table 8. When α_1-α_4 are 1.0936, 0, 65.2252 and 10, the best solution obtained by TCIA is 2302.55. Obvious, the TCIA is more excellent than the other comparison algorithms in the design of pressure vessels.

$$\alpha = [\alpha_1, \alpha_2, \alpha_3, \alpha_4] = [T_s, T_h, R, L] \tag{7}$$

$$\begin{aligned} Minf\ (\alpha) &= 0.6224\alpha_1\alpha_3\alpha_4 + 1.7781\alpha_2\alpha_3^2 \\ &\quad + 3.1661\alpha_1^2\alpha_4 + 19.84\alpha_1^2\alpha_3 \end{aligned} \tag{8}$$

$$\begin{cases} g_1(\alpha) = -\alpha_1 + 0.0193\alpha_3 \le 0, \\ g_2(\alpha) = -\alpha_2 + 0.00954\alpha_3 \le 0, \\ g_3(\alpha) = -\pi\alpha_3^2\alpha_4 - \frac{4}{3}\pi\alpha_3^3 + 1296000 \le 0, \\ g_4(\alpha) = \alpha_4 - 240 \le 0, \\ 0 \le \alpha_1 \le 100, 0 \le \alpha_2 \le 100, \\ 10 \le \alpha_3 \le 100, 10 \le \alpha_4 \le 200. \end{cases} \tag{9}$$

B. WELDED BEAM DESIGN PROBLEM

In this section, TCIA is applied to Welded beam design problem. The goal of this question is to minimize its cost under all five kinds of constraints. The five constraints are end deflection of the beam (δ), bending stress of the beam (σ), buckling load of the bar (Pc), shear stress (τ) and lateral constraints [30]. FIGURE 17 shows the design drawings of Welded beam design problem.

The $h(\delta_1)$, $l(\delta_2)$, $t(\delta_3)$, and $b(\delta_4)$ are the design variables of Welded beam design problem. This question has seven inequality constraints, five of them are nonlinear and two are linear. Let $\delta = [\delta_1, \delta_2, \delta_3, \delta_4] = [d, l, t, b]$, then the design problem can be modeled as (10), and the constraint condition is (11). TABLE 9 shows the solutions given by TCIA and other algorithms. When δ_1-δ_4 are 0.20573, 3.25300, 9.03660, 0.20573, the best solution obtained by TCIA is 1.6952. In the optimization of this problem, the TCIA is more excellent than the other comparison algorithms, and has a strong competitiveness.

TCIA can give the best solution to all two kinds of engineering problems and shows outstanding competitiveness. Overall, TCIA is good at solving constrained engineering optimization problems [56].

$$Minf\ (\delta) = 1.10471\delta_1^2\delta_2 + 0.04811\delta_3\delta_4\,(14 + \delta_2) \tag{10}$$

$$\begin{cases} g_1(\delta) = \tau(\delta) + \tau_{max} \le 0, \\ g_2(\delta) = \sigma(\delta) + \sigma_{max} \le 0, \\ g_3(\delta) = \delta(\delta) + \delta_{max} \le 0, \\ g_4(\delta) = \delta_1 - \delta_4 \le 0, \\ g_5(\delta) = P - P_c(\delta) \le 0, \\ g_6(\delta) = 0.125 - \delta_1 \le 0, \\ g_7(\delta) = 0.10471\delta_1^2 + 0.04811\delta_3\delta_4\,(14 + \delta_2) - 5 \le 0, \\ 0.1 \le \delta_1, 0.1 \le \delta_2, \delta_3 \le 10, \delta_4 \le 2. \end{cases} \tag{11}$$

C. PREDICTION OF CONCRETE COMPRESSIVE STRENGTH

Concrete is an important raw material of modern infrastructure, and its strength directly affects the service life of buildings. Therefore, it is very meaningful to predict its strength through known information. The compressive strength of concrete is affected by many factors and is highly non-linear. The concrete compressive strength data set has 1030 sets of data, including eight influencing factors [57]. The characteristic information of the data set is shown in TABLE 10.

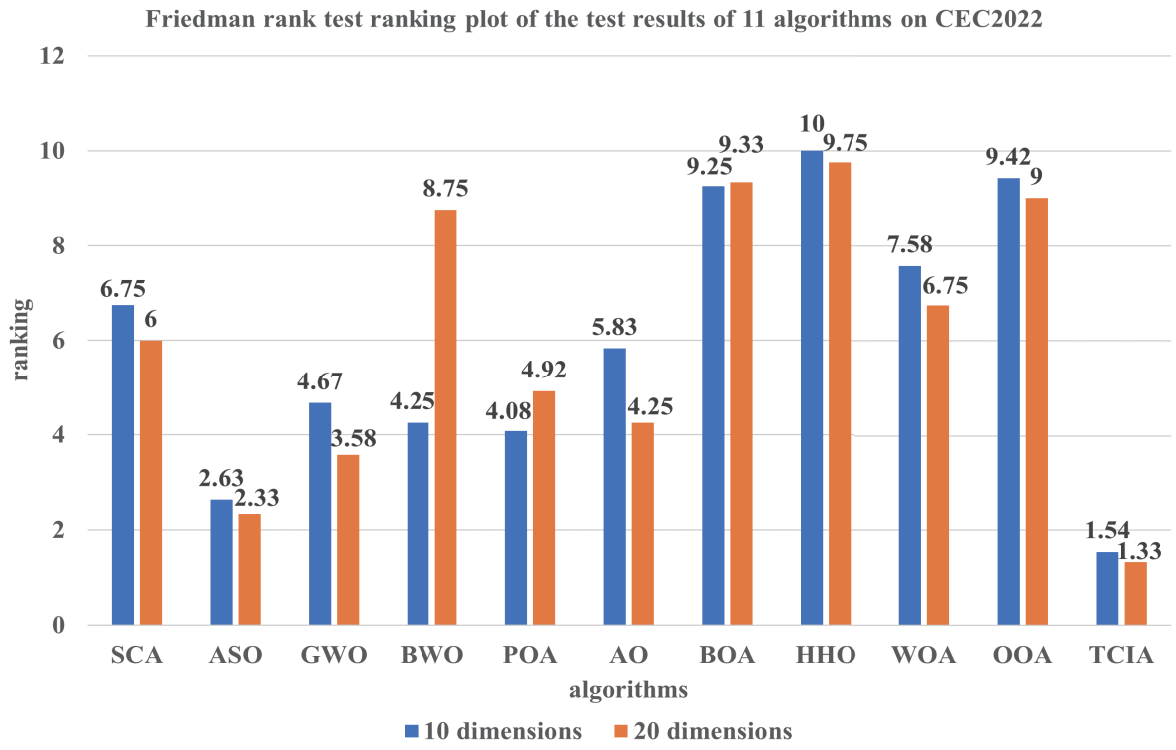


FIGURE 14. Friedman rank test ranking plot of the test results of 11 algorithms on CEC2022.

TABLE 5. Results of Wilcoxon rank sum test (means, 10-dimension).

D(CEC2022)	TCIA vs	p-value	R+	R-	+/=/-
D10	SCA	1.4258E-03	8.3333E+00	4.5667E+02	12/0/0
	ASO	9.1236E-02	1.7033E+02	2.8808E+02	8/3/1
	GWO	4.4059E-02	1.1067E+02	3.5433E+02	9/2/1
	BWO	3.4075E-02	7.0417E+01	4.6500E+02	10/2/0
	POA	3.4333E-02	1.1375E+02	3.5125E+02	10/2/0
	AO	4.0495E-03	5.0083E+01	4.1492E+02	12/0/0
	BOA	3.4333E-02	1.1375E+02	3.5125E+02	12/0/0
	HHO	9.4590E-05	7.0000E+00	4.5800E+02	12/0/0
	WOA	9.5953E-03	4.0250E+01	4.2475E+02	11/1/0
	OOA	5.8152E-04	1.2250E+01	4.5275E+02	12/0/0

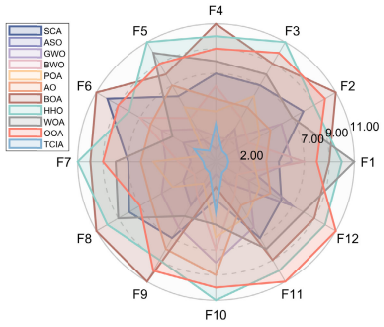
TABLE 6. Results of Wilcoxon rank sum test (means, 20-dimension).

D(CEC2022)	TCIA vs	p-value	R+	R-	+/=/-
D10	SCA	3.7486E-03	1.5917E+01	4.4908E+02	12/0/0
	ASO	1.1553E-01	1.2008E+02	3.4492E+02	8/3/1
	GWO	1.5002E-01	7.5417E+01	3.8958E+02	9/2/1
	BWO	4.1866E-06	3.7500E+00	4.6125E+02	12/0/0
	POA	3.0278E-03	2.7917E+01	4.3708E+02	12/0/0
	AO	1.7674E-03	3.5083E+01	4.2992E+02	12/0/0
	BOA	3.6746E-04	8.1667E+00	4.5683E+02	12/0/0
	HHO	1.7499E-06	8.3333E-02	4.6492E+02	12/0/0
	WOA	9.7532E-06	8.0000E+00	4.5700E+02	12/0/0
	OOA	1.7344E-06	0.0000E+00	4.6500E+02	12/0/0

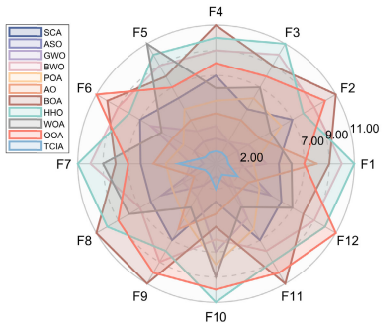
The mapping relationship between the compressive strength of concrete and the eight influencing factors is unknown and difficult to be represented by mathematical models. Therefore, the BP neural network model is used to predict the compressive strength of concrete. BP neural network is a multi-layer feedforward neural network, which

TABLE 7. Friedman rank test tests the average ranking of 11 algorithms on 12 functions.

Method	Ranked in different dimensions	
	10	20
SCA	6.75	6.00
ASO	2.63	2.33
GWO	4.67	3.58
BWO	4.25	8.75
POA	4.08	4.92
AO	5.83	4.25
BOA	9.25	9.33
HHO	10.00	9.75
WOA	7.58	6.75
OOA	9.42	9.00
TCIA	**1.54**	**1.33**
p-value	1.05E-14	4.20E-16



(a) Radar chart on 10 dimensions



(b) Radar chart on 20 dimensions

FIGURE 15. Radar chart.

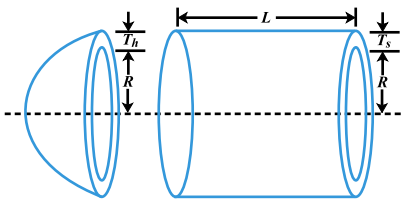


FIGURE 16. Schematic diagram of pressure vessel.

includes three parts: input layer, hidden layer and output layer, and its structure diagram is shown in FIGURE 18. Among them, $X_1, X_2, X_3, \cdots, X_n$ are the input variables, n is the number of input nodes, W_{ij} is the connection weight of

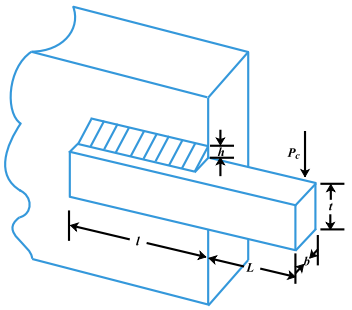


FIGURE 17. Schematic diagram of welded beam.

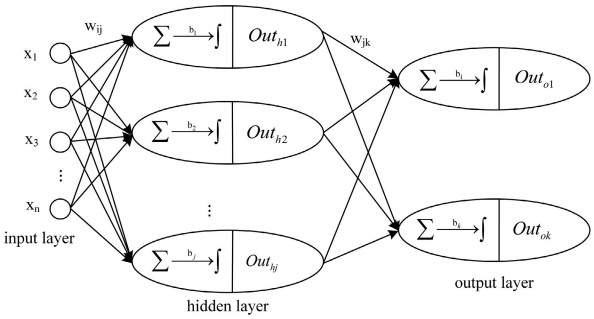


FIGURE 18. BP structure diagram.

the input layer and the hidden layer, j is the number of hidden layer nodes. $\int$ is the operation of the activation function, $Out_{h1}, Out_{h2}, \cdots, Out_{hj}$ are the output results obtained by the hidden layer. W_{jk} is the connection weight of the hidden layer and the output layer, k is the number of output layer nodes, $Out_{o1}, \cdots, Out_{ok}$ are the final output result obtained by the output layer.

The learning process of BP neural network is divided into two stages: signal forward transmission and error back propagation. The forward signal transmission includes two parts: input layer to hidden layer and hidden layer to output layer, and the calculation formula is shown in equation (12) and equation (13), respectively. When the error does not reach the accuracy requirement, the error back propagation phase starts. The network parameters are adjusted by calculating the error between the output layer and the desired value, thus making the error smaller. The calculation error formula is given in equation (14). Where Num is the number of training samples, y_i and y_{pre} are the actual input and predicted values, respectively. The main parameters of BP neural network are weights and thresholds. Through the two stages of cyclic training, the weights and thresholds will be constantly corrected, and finally the actual output that meets the error requirements will be obtained. BP network is sensitive to the initial weight and threshold setting, unreasonable initial weight and threshold will reduce the performance of the network prediction. In this paper, TCIA is used to calculate the error between the predicted value and the true value of BP network to find the best weight and threshold for subsequent network prediction experiments.

TABLE 8. Test results of pressure vessel design problems.

Method	Parameters				Best cost
	α_1	α_2	α_3	α_4	
SCA	1.0000	0.0000E+00	40.3261	200.00	6057.40
ASO	1.0936	0.0000E+00	65.2252	10.00	2302.55
GWO	1.0926	3.6765E-04	65.2262	10.00	2305.43
BWO	1.1283	0.0000E+00	65.2285	10.00	2316.46
POA	1.0936	8.6783E-20	65.2252	10.00	2302.55
AO	1.0978	0.0000E+00	41.3120	187.15	3642.60
BOA	1.3858	0.0000E+00	66.3504	10.00	3161.05
HHO	1.0021	5.2579E-01	65.2252	10.00	6739.71
WOA	1.1008	0.0000E+00	65.2252	10.00	2303.14
OOA	3.2945	2.2612E+01	55.9342	60.54	4307.90
TCIA	1.0936	0.0000E+00	65.2252	10.00	2302.55

TABLE 9. Test results of design problems of welded beams.

Method	Parameters				Best cost
	δ_1	δ_2	δ_3	δ_4	
SCA	0.19330	3.69270	9.37570	0.21061	1.8332
ASO	0.20570	3.20530	9.13740	0.20575	1.7060
GWO	0.20538	3.26080	9.03910	0.20572	1.6961
BWO	0.19877	3.51230	9.18800	0.20741	1.7589
POA	0.20574	3.25300	9.03660	0.20573	1.6952
AO	0.12485	7.77410	9.17700	0.20806	2.1363
BOA	0.17179	4.20040	8.92530	0.22345	1.8832
HHO	0.35060	3.15900	4.25230	0.92909	3.6904
WOA	0.23011	2.76410	9.47770	0.23656	1.9699
OOA	0.20358	7.86950	6.56750	0.38955	3.0521
TCIA	0.20573	3.25300	9.03660	0.20573	1.6952

TABLE 10. Variable description of concrete compressive strength data set.

variable	description	unit
x_1	cement	kg in m^3
x_2	blast furnace slag	kg in m^3
x_3	coal ash	kg in m^3
x_4	water	kg in m^3
x_5	super plasticizer	kg in m^3
x_6	coarse aggregate	kg in m^3
x_7	fine aggregate	kg in m^3
x_8	age	day
y	compressive strength	Mpa

TABLE 11. Comparison of prediction performance of concrete compressive strength.

Evaluation indicator	BP	ASO-BP	GWO-BP	TCIA-BP
MAE	4.61600180	4.44072158	4.61797744	4.20073472
RMSE	6.29780794	6.14329448	6.37196659	5.61644297
R^2	0.85329176	0.86519661	0.85694307	0.88506112

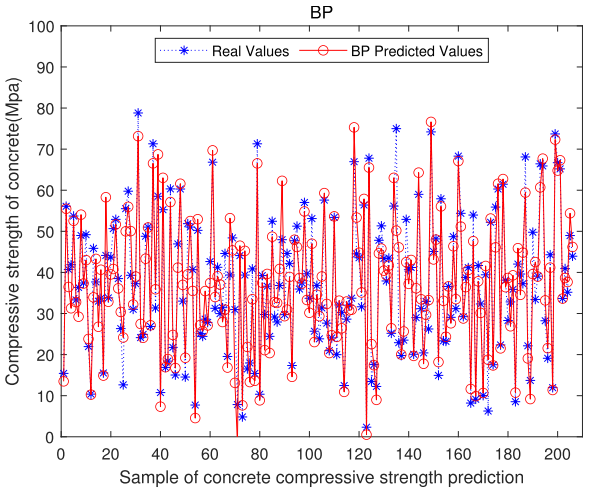


FIGURE 19. Plot of BP relative prediction error.

The 1030 groups of concrete compressive strength degrees were divided into training set and test set according to the ratio of 8:2. The BP neural network model optimized by TCIA is used to predict the compressive strength of concrete. The test results of the BP neural network optimized by TCIA and several other networks are shown in TABLE 11, and the comparison results between the true value and the predicted value are shown in FIGURE 19, FIGURE 20, FIGURE 21 and FIGURE 22, respectively. The evaluation metrics include mean absolute error (MAE), root mean square error (RMSE), and coefficient of fitting determination (R^2). The mean absolute error gives a visual representation of the average difference between the predicted and true values. The root mean square error represents the error between the predicted and true values. The coefficient of fit determination

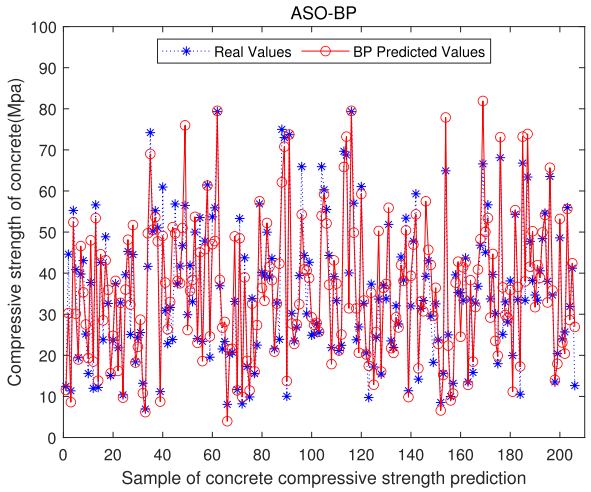


FIGURE 20. Plot of ASO-BP relative prediction error.

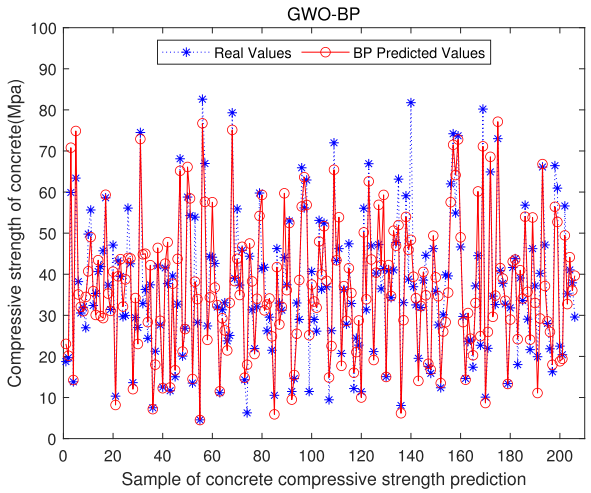


FIGURE 21. Plot of GWO-BP relative prediction error.

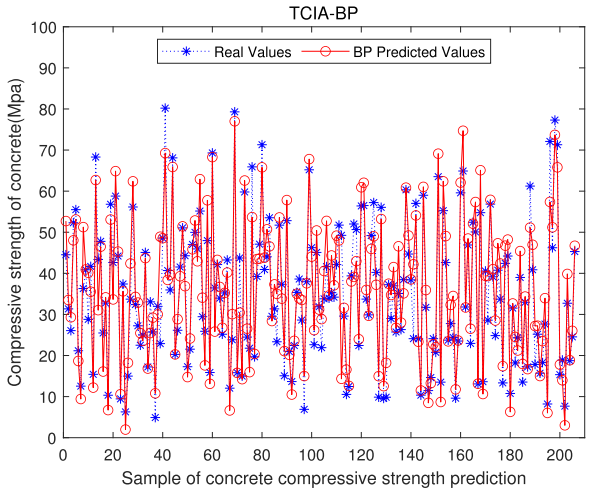


FIGURE 22. Plot of TCIA-BP relative prediction error.

represents the degree of fit between the current model and the problem to be solved. Through the above three indicators, it can be seen that the parameters of BP neural network are more accurate in the prediction of concrete compressive strength after TCIA optimization.

$$Out_{hj} = f(\sum_{i=1}^{n} X_i W_{ij} + b_j) \tag{12}$$

$$Out_{ok} = f(\sum_{j=1}^{k} Out_{hj} W_{jk} + b_k) \tag{13}$$

$$Error = \frac{1}{Num} \sum_{i=1}^{Num} (y_i - y_{pre})^2 \tag{14}$$

V. CONCLUSIONS AND PROSPECTS

A novel meta-heuristic algorithm named T cell immune algorithm(TCIA) is proposed in this paper. The TCIA draws inspiration from the T-cell immune process and can be used to solve a number of practical problems. TCIA imitates the behavior of T cell immune process in recognizing antigens, activating cells and attacking pathogens, thus enabling the modeling of the three main steps of recognition, activation and attack. The recognition process is to evaluate the fitness value of the current cell. The activation process helps the population to explore in a wider range. Attacks are divided into random attack and directed attack, where random attack can help population jump out of the local optimum and directed attack can better exploit and utilize the current optimal solution. Benefiting from the above three processes, TCIA balances exploration and exploitation well. TCIA is compared with the other 10 algorithms on 12 functions of CEC2022. The results and convergence curves show that TCIA is better than the comparison algorithms in terms of convergence speed and solution quality. TCIA is tested on three multi-objective functions, which verifies its good ability in solving multi-objective functions. The box plot test and the location analysis of four random cells reveal that TCIA has strong stability in solving the optimization problems. The superiority of TCIA in solving constraint problems is visualized by Wilcoxon rank sum test, Friedman rank test and Radar chart statistical analysis. Finally, through three practical engineering cases, TCIA demonstrates its strong competency in solving real-world problems. In the future, TCIA can be improved to solve more realistic applications.

REPLICATION OF RESULTS

TCIA, freely available at https://github.com/xiaozhi1124/TCIA.git, was used to obtain the results above.

REFERENCES

[1] H. Su, D. Zhao, A. A. Heidari, L. Liu, X. Zhang, M. Mafarja, and H. Chen, "RIME: A physics-based optimization," *Neurocomputing*, vol. 532, pp. 183–214, May 2023.

[2] B. Das, V. Mukherjee, and D. Das, "Student psychology based optimization algorithm: A new population based optimization algorithm for solving optimization problems," *Adv. Eng. Softw.*, vol. 146, Aug. 2020, Art. no. 102804. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0965997820301484

[3] B. Abdollahzadeh, F. S. Gharehchopogh, N. Khodadadi, and S. Mirjalili, "Mountain gazelle optimizer: A new nature-inspired metaheuristic algorithm for global optimization problems," *Adv. Eng. Softw.*, vol. 174, Dec. 2022, Art. no. 103282. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0965997822001831

[4] M. A. Elaziz, D. Yousri, and S. Mirjalili, "A hybrid Harris hawks-moth-flame optimization algorithm including fractional-order chaos maps and evolutionary population dynamics," *Adv. Eng. Softw.*, vol. 154, Apr. 2021, Art. no. 102973. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0965997821000028

[5] J. Chen, M. Chen, J. Wen, L. He, and X. Liu, "A heuristic construction neural network method for the time-dependent agile earth observation satellite scheduling problem," *Mathematics*, vol. 10, no. 19, p. 3498, Sep. 2022.

[6] A. Seyyedabbasi, "A reinforcement learning-based metaheuristic algorithm for solving global optimization problems," *Adv. Eng. Softw.*, vol. 178, Apr. 2023, Art. no. 103411. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0965997823000030

[7] N. Moradi, V. Kayvanfar, and M. Rafiee, "An efficient population-based simulated annealing algorithm for 0–1 knapsack problem," *Eng. Comput.*, vol. 38, pp. 2771–2790, Jan. 2021, doi: 10.1007/s00366-020-01240-3.

[8] J. F. Gonçalves and M. G. C. Resende, "Biased random-key genetic algorithms for combinatorial optimization," *J. Heuristics*, vol. 17, no. 5, pp. 487–525, Oct. 2011, doi: 10.1007/s10732-010-9143-1.

[9] J. Liu, J. Yang, H. Liu, X. Tian, and M. Gao, "An improved ant colony algorithm for robot path planning," *Soft Comput.*, vol. 21, no. 19, pp. 5829–5839, Oct. 2017, doi: 10.1007/s00500-016-2161-7.

[10] S.-C. Chu, J. F. Roddick, C.-J. Su, and J.-S. Pan, "Constrained ant colony optimization for data clustering," in *Proc. PRICAI Trends Artif. Intell., 8th Pacific Rim Int. Conf. Artif. Intell.* Auckland, New Zealand: Springer, Aug. 2004, pp. 534–543.

[11] I. C. Trelea, "The particle swarm optimization algorithm: Convergence analysis and parameter selection," *Inf. Process. Lett.*, vol. 85, no. 6, pp. 317–325, Mar. 2003.

[12] X. Li, "A new intelligent optimization-artificial fish swarm algorithm," Doctor thesis, Dept. Control Sci. Eng., Zhejiang Univ., Zhejiang, China, vol. 27, 2003.

[13] D. T. Pham, A. Ghanbarzadeh, E. Koc, S. Otri, S. Rahim, and M. Zaidi, "The bees algorithm—A novel tool for complex optimisation problems," in *Intelligent Production Machines and Systems*. Amsterdam, The Netherlands: Elsevier, 2006, pp. 454–459.

[14] W.-T. Pan, "A new fruit fly optimization algorithm: Taking the financial distress model as an example," *Knowl.-Based Syst.*, vol. 26, pp. 69–74, Feb. 2012. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705111001365

[15] J. Xue and B. Shen, "A novel swarm intelligence optimization approach: Sparrow search algorithm," *Syst. Sci. Control Eng.*, vol. 8, no. 1, pp. 22–34, Jan. 2020, doi: 10.1080/21642583.2019.1708830.

[16] F. Sadeghi, A. Larijani, O. Rostami, D. Martín, and P. Hajirahimi, "A novel multi-objective binary chimp optimization algorithm for optimal feature selection: Application of deep-learning-based approaches for SAR image classification," *Sensors*, vol. 23, no. 3, p. 1180, Jan. 2023.

[17] C.-F. Juang, "A hybrid of genetic algorithm and particle swarm optimization for recurrent network design," *IEEE Trans. Syst., Man Cybern., B Cybern.*, vol. 34, no. 2, pp. 997–1006, Apr. 2004.

[18] Z. A. Khan, A. Zafar, S. Javaid, S. Aslam, M. H. Rahim, and N. Javaid, "Hybrid meta-heuristic optimization based home energy management system in smart grid," *J. Ambient Intell. Humanized Comput.*, vol. 10, no. 12, pp. 4837–4853, Dec. 2019.

[19] L. A. Moncayo-Martínez and D. Z. Zhang, "Multi-objective ant colony optimisation: A meta-heuristic approach to supply chain design," *Int. J. Prod. Econ.*, vol. 131, no. 1, pp. 407–420, May 2011.

[20] M. Kaveh and M. S. Mesgari, "Application of meta-heuristic algorithms for training neural networks and deep learning architectures: A comprehensive review," *Neural Process. Lett.*, vol. 55, pp. 4519–4622, Oct. 2022.

[21] H. Li, S. Guo, H. Zhao, C. Su, and B. Wang, "Annual electric load forecasting by a least squares support vector machine with a fruit fly optimization algorithm," *Energies*, vol. 5, no. 11, pp. 4430–4445, Nov. 2012.

[22] H. Jafari and N. Salmasi, "Maximizing the nurses' preferences in nurse scheduling problem: Mathematical modeling and a meta-heuristic algorithm," *J. Ind. Eng. Int.*, vol. 11, no. 3, pp. 439–458, Sep. 2015.

[23] M. Abdel-Basset, G. Manogaran, L. Abdel-Fatah, and S. Mirjalili, "An improved nature inspired meta-heuristic algorithm for 1-D bin packing problems," *Pers. Ubiquitous Comput.*, vol. 22, nos. 5–6, pp. 1117–1132, Oct. 2018.

[24] A. H. Halim and I. Ismail, "Combinatorial optimization: Comparison of heuristic algorithms in travelling salesman problem," *Arch. Comput. Methods Eng.*, vol. 26, no. 2, pp. 367–380, Apr. 2019.

[25] M. AkbaiZadeh, T. Niknam, and A. Kavousi-Fard, "Adaptive robust optimization for the energy management of the grid-connected energy hubs based on hybrid meta-heuristic algorithm," *Energy*, vol. 235, Nov. 2021, Art. no. 121171.

[26] S. Khalifehzadeh, M. Seifbarghy, and B. Naderi, "Solving a fuzzy multi objective model of a production–distribution system using meta-heuristic based approaches," *J. Intell. Manuf.*, vol. 28, no. 1, pp. 95–109, Jan. 2017.

[27] H. Zhu, Y. Wang, K. Wang, and Y. Chen, "Particle swarm optimization (PSO) for the constrained portfolio optimization problem," *Exp. Syst. Appl.*, vol. 38, no. 8, pp. 10161–10169, Aug. 2011.

[28] S. Kumar, G. G. Tejani, N. Pholdee, and S. Bureerat, "Improved metaheuristics through migration-based search and an acceptance probability for truss optimization," *Asian J. Civil Eng.*, vol. 21, no. 7, pp. 1217–1237, Nov. 2020.

[29] S. Kumar, P. Jangir, G. G. Tejani, and M. Premkumar, "A decomposition based multi-objective heat transfer search algorithm for structure optimization," *Knowl.-Based Syst.*, vol. 253, Oct. 2022, Art. no. 109591. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705122008024

[30] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 67–82, Apr. 1997.

[31] H. L. Robinson and R. R. Amara, "T cell vaccines for microbial infections," *Nature Med.*, vol. 11, no. S4, pp. S25–S32, Apr. 2005, doi: 10.1038/nm1212.

[32] V. P. Badovinac, K. A. N. Messingham, A. Jabbari, J. S. Haring, and J. T. Harty, "Accelerated CD8+ T-cell memory and prime-boost response after dendritic-cell vaccination," *Nature Med.*, vol. 11, no. 7, pp. 748–756, Jul. 2005, doi: 10.1038/nm1257.

[33] Q. Xu, G. Liu, X. Yuan, M. Xu, H. Wang, J. Ji, B. Konda, K. L. Black, and J. S. Yu, "Antigen-specific T-cell response from dendritic cell vaccination using cancer stem-like cell-associated antigens," *Stem Cells*, vol. 27, no. 8, pp. 1734–1740, Aug. 2009.

[34] A. Lanzavecchia and F. Sallusto, "Regulation of T cell immunity by dendritic cells," *Cell*, vol. 106, no. 3, pp. 263–266, 2001, doi: 10.1016/S0092-8674(01)00455-X.

[35] K. Choudhuri, D. Wiseman, M. H. Brown, K. Gould, and P. A. van der Merwe, "T-cell receptor triggering is critically dependent on the dimensions of its peptide-MHC ligand," *Nature*, vol. 436, no. 7050, pp. 578–582, Jul. 2005, doi: 10.1038/nature03843.

[36] H. Jonuleit, G. Adema, and E. Schmitt, "Immune regulation by regulatory T cells: Implications for transplantation," *Transplant Immunology*, vol. 11, nos. 3–4, pp. 267–276, Jul. 2003. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0966327403000571

[37] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014.

[38] S. Mirjalili, "SCA: A sine cosine algorithm for solving optimization problems," *Knowl.-Based Syst.*, vol. 96, pp. 120–133, Mar. 2016.

[39] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Adv. Eng. Softw.*, vol. 95, pp. 51–67, May 2019.

[40] S. Arora and S. Singh, "Butterfly optimization algorithm: A novel approach for global optimization," *Soft Comput.*, vol. 23, no. 3, pp. 715–734, Feb. 2019.

[41] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: Algorithm and applications," *Future Gener. Comput. Syst.*, vol. 97, pp. 849–872, Aug. 2019.

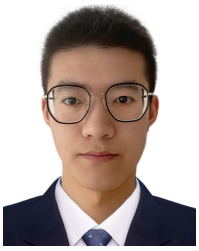
[42] W. Zhao, L. Wang, and Z. Zhang, "Atom search optimization and its application to solve a hydrogeologic parameter estimation problem," *Knowl.-Based Syst.*, vol. 163, pp. 283–304, Jan. 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705118304271

[43] L. Abualigah, D. Yousri, M. A. Elaziz, A. A. Ewees, M. A. A. Al-Qaness, and A. H. Gandomi, "Aquila optimizer: A novel meta-heuristic optimization algorithm," *Comput. Ind. Eng.*, vol. 157, Jul. 2021, Art. no. 107250.

[44] C. Zhong, G. Li, and Z. Meng, "Beluga whale optimization: A novel nature-inspired metaheuristic algorithm," *Knowl.-Based Syst.*, vol. 251, Sep. 2022, Art. no. 109215.

[45] P. Trojovský and M. Dehghani, "Pelican optimization algorithm: A novel nature-inspired algorithm for engineering applications," *Sensors*, vol. 22, no. 3, p. 855, Jan. 2022.

[46] M. Dehghani and P. Trojovský, "Osprey optimization algorithm: A new bio-inspired metaheuristic algorithm for solving engineering optimization problems," *Frontiers Mech. Eng.*, vol. 8, Jan. 2023, Art. no. 1126450.
[47] W. Luo, X. Lin, C. Li, S. Yang, and Y. Shi, "Benchmark functions for CEC 2022 competition on seeking multiple optima in dynamic environments," 2022, *arXiv:2201.00523*.
[48] S. Kumar, P. Jangir, G. G. Tejani, and M. Premkumar, "MOTEO: A novel physics-based multiobjective thermal exchange optimization algorithm to design truss structures," *Knowl.-Based Syst.*, vol. 242, Apr. 2022, Art. no. 108422. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S095070512200171X
[49] S. Kumar, P. Jangir, G. G. Tejani, M. Premkumar, and H. H. Alhelou, "MOPGO: A new physics-based multi-objective plasma generation optimizer for solving structural optimization problems," *IEEE Access*, vol. 9, pp. 84982–85016, 2021.
[50] R. Mcgill, J. W. Tukey, and W. A. Larsen, "Variations of box plots," *Amer. Statistician*, vol. 32, no. 1, pp. 12–16, Feb. 1978.
[51] J. Cuzick, "A Wilcoxon-type test for trend," *Statist. Med.*, vol. 4, no. 1, pp. 87–90, Jan. 1985.
[52] G. A. Mack and J. H. Skillings, "A friedman-type rank test for main effects in a two-factor ANOVA," *J. Amer. Stat. Assoc.*, vol. 75, no. 372, pp. 947–951, Dec. 1980.
[53] X.-S. Yang, C. Huyck, M. Karamanoglu, and N. Khan, "True global optimality of the pressure vessel design problem: A benchmark for bio-inspired optimisation algorithms," *Int. J. Bio-Inspired Comput.*, vol. 5, no. 6, pp. 329–335, 2013.
[54] E. Trojovská, M. Dehghani, and P. Trojovský, "Zebra optimization algorithm: A new bio-inspired optimization algorithm for solving optimization algorithm," *IEEE Access*, vol. 10, pp. 49445–49473, 2022.
[55] S. Q. Salih, A. A. Alsewari, and Z. M. Yaseen, "Pressure vessel design simulation: Implementing of multi-swarm particle swarm optimization," in *Proc. 8th Int. Conf. Softw. Comput. Appl.*, 2019, pp. 120–124.
[56] F. A. Hashim, K. Hussain, E. H. Houssein, M. S. Mabrouk, and W. Al-Atabany, "Archimedes optimization algorithm: A new metaheuristic algorithm for solving optimization problems," *Appl. Intell.*, vol. 51, pp. 1531–1551, Sep. 2021.
[57] I.-C. Yeh, "Analysis of strength of concrete using design of experiments and neural networks," *J. Mater. Civil Eng.*, vol. 18, no. 4, pp. 597–604, Aug. 2006.



HONGZHI ZHANG was born in Jilin, China. He received the bachelor's degree in automation from the University of Science and Technology Liaoning, where he is currently pursuing the master's degree in control science and engineering. His research interests are modeling and optimization of complex industrial systems.

YONG ZHANG was born in Liaoning, China. He is currently a Doctoral Supervisor with the University of Science and Technology Liaoning. He has published more than 60 academic articles, ten indexed by SCI and more than 20 indexed by EI. His research interests include control theory and control engineering, production scheduling and integrated optimization in process industries, and network security and fault analysis. He has presided over the National Natural Science Foundation projects: research and application of optimal scheduling method for cold rolling production process, disturbance rejection identification, and robust iterative learning control with constraints for batch production process.

YIXING NIU was born in Shandong, China. He received the bachelor's degree in computer science and technology. He is currently pursuing the master's degree in electronic information with the University of Science and Technology Liaoning. His research interests are image processing on deep learning and intelligent controls.

KAI HE was born in Anhui, China. He received the bachelor's degree in mechanical design manufacture and automation from Bengbu University and the master's degree in electrical engineering from the Anhui University of Science and Technology. He is currently pursuing the Ph.D. degree in control science and engineering with the University of Science and Technology Liaoning. His research interests are control optimization and evolutionary algorithms.

YUKUN WANG was born in Hebei, China. He is currently an Associate Professor with the University of Science and Technology Liaoning. His research interests include machine learning, neural networks, and intelligent optimization algorithm.