RESEARCH ARTICLE

Self-Adaptive Forensic-Based Investigation Algorithm with Dynamic Population for Solving Constraint Optimization Problems

Pengxing Cai[1] · Yu Zhang[1] · Ting Jin[2] · Yuki Todo[3] · Shangce Gao[1]

Abstract
The Forensic-Based Investigation (FBI) algorithm is a novel metaheuristic algorithm. Many researches have shown that FBI is a promising algorithm due to two specific population types. However, there is no sufficient information exchange between these two population types in the original FBI algorithm. Therefore, FBI suffers from many problems. This paper incorporates a novel self-adaptive population control strategy into FBI algorithm to adjust parameters based on the fitness transformation from the previous iteration, named SaFBI. In addition to the self-adaptive mechanism, our proposed SaFBI refers to a novel updating operator to further improve the robustness and effectiveness of the algorithm. To prove the availability of the proposed algorithm, we select 51 CEC benchmark functions and two well-known engineering problems to verify the performance of SaFBI. Experimental and statistical results manifest that the proposed SaFBI algorithm performs superiorly compared to some state-of-the-art algorithms.

Keywords Metaheuristic algorithm · Evolutionary algorithm · Optimization problem · Self-adaptive mechanism · Engineering problem

1 Introduction

Although deterministic algorithm can effectively find the optimal solution for some problems, in general, it has high-level time and space complexity, in addition to that, deterministic algorithms are dependent on the property of specific problem. Stochastic algorithms are more efficient and easier to understand in comparison with traditional deterministic algorithms. With the development of stochastic algorithms for several decades, meta-heuristic algorithms (MHAs) emerged and developed rapidly, which have been successfully applied to existing optimization problem due to their effectiveness and robustness.

MHAs are inspired by physical phenomena, ecosystems or immune systems, whose principles cause their intelligibility and interpretability of them. In accordance with distinct properties, MHAs include the algorithms which draw inspiration from swarm behaviour, such as particle swarm optimization [1, 2], ant colony optimization [3], artificial bee colony algorithm [4], bat algorithm [5], dynamic hunting leadership optimization [6], slime mould algorithm [7]. The algorithms inspired by natural (physical) phenomena contain (but are not limited to) a simulated annealing algorithm [8], gravitational search algorithm [9], plasma generation optimization [10], and sine cosine algorithm [11]. Except for these algorithms above, a vitally important part of MHAs is genetic algorithms, which involve differential evolution [12], genetic algorithm [13], and artificial immune system [14]. Not only that, but most novel biology-inspired evolutionary algorithm or variant of classical MHAs have been successfully applied

✉ Shangce Gao
gaosc@eng.u-toyama.ac.jp

Pengxing Cai
d2172005@ems.u-toyama.ac.jp

Yu Zhang
d2272005@ems.u-toyama.ac.jp

Ting Jin
tingjin@njfu.edu.cn

Yuki Todo
yktodo@se.kanazawa-u.ac.jp

[1] Faculty of Engineering, University of Toyama, Toyama-shi 930-8555, Japan

[2] School of Science, Nanjing Forestry University, Nanjing 210037, China

[3] Faculty of Electrical, Information and Communication Engineering, Kanazawa University, Kanazawa, Ishikawa 920-1192, Japan

to optimization problems in various fields. Such as protein energy landscapes mapping [15], droplet routing automation [16], road map partitioning [17], digital IIR filter optimization [18], and so on.

Although the novel MHAs emerge endlessly in recent years, the pursuit of high-performance algorithms has never stagnated on the basis of exploration and exploitation in organizational learning [19]. Achieving a better balance between the exploration ability and exploitation ability of the proposed algorithm is crucial to find the optimal solution for different problems. The concept of exploration and exploitation represent the global search process and local search operator in MHAs, respectively. Since the mechanism and implementation of earlier MHAs were relatively simple, most of them have only a global search mechanism. It stands to reason that the mix of global search algorithm and local search strategy is an available method to improve the performance of an original algorithm, the common and effective local search such as chaotic local search (CLS) [20, 21], hill-climbing algorithm (HC) [22, 23], local optima topology (LOT) [24].

Whereas with the rapid development of MHAs and hybrid algorithms, current mechanisms contain multiple global and local search operators. Meanwhile, new shortcomings appear that the fixed calling mechanism of multiple search operators is unable to fit distinct characteristics of optimization problems. Therefore, adaptive and self-adaptive parameters (or judgment mechanisms) are incorporated into original algorithms, for example, aggregative learning gravitational search algorithm [25], self-adaptive bat algorithm [26], self-adaptive strategy based firefly algorithm [27] and self-adaptive hybrid self-learning teaching learning based optimization (SHSLTLBO) [28]. Self-adaptive MHAs perform superior to origin in that they can dynamically obtain information about the current population and the specificity of objective problems.

Despite the fact that according to No Free Lunch theory, none of global or local optimization algorithms can solve various existing kinds of problems perfectly on account of the randomness initialization and non-deterministic search mechanism. Nevertheless, it is also extremely significant to design a metaheuristic algorithm that can perform well in most optimization problems and keep both time complexity and space complexity low. Recently, a novel FBI inspired meta-optimization algorithm was proposed [29]. Different from most of other MHAs, FBI is inspired from the system of human work and refer to a double population mechanism. Then some variants have been applied to the optimal design of frequency-constrained dome structures and Parameter Extraction of Solar Cell Models [30, 31]. However, both the original FBI and its variants are only suitable for some specific optimization problems, and most of them perform very ordinary or even poorly on other types

of problems. According to the previous research, it can be concluded that MHAs with multiple population types have a good performance in population diversity, but most of the time consumption is much more than that of ordinary meta-heuristic algorithms. Therefore, we consider that the information between two different populations, and design a general dynamic population mechanism. When dealing with optimization problems, our proposed dynamic population mechanism adapts to the problem by changing the composition of population according to the complexity of problem. To demonstrate the robustness and effectiveness, experimental datasets involve 29 CEC2017 benchmark functions [32], 22 CEC2011 real-world optimization problems [33] and two classical engineering problems [34]. Each dataset has different complexity and number of dimension, additionally, each of those dimensions has different upper and lower bounds on its value.

This work gives the following contributions and originality:

1) We design a novel dynamic population architecture to regulate the proportion of each individual type. When a large number of individuals are in a state of stagnation, the proportion of explorers (Investigation team) in the population will increase based on evaluation value, whereas conversely, the proportion of exploiters (Pursuit team) will rise. This is a guide to the improvement of some existing evolutionary algorithms with multiple population.

2) We put forward a effective updating operator to expand the search space based on two gaussian random numbers and the current iteration number, providing a promising operator for SaFBI and other MHAs.

3) Experimental results demonstrate that SaFBI can be applied to various optimization problems with different dimension. In addition to the chart data, we illustrate the convergence graph, box-and-whisker plot, diversity graph, and three-dimensional contour map to reflect the effectiveness of SaFBI in terms of diverse metrics.

The rest of this paper can be divided into the following sections. Table 1 lists the explanations of frequent symbols. Section 2 briefly introduces FBI algorithm. Section 3 depicts our proposed SaFBI algorithm. The experimental results and statistical analyses are shown in Sect. 4. Section 5 exhibit the conclusion of this work.

2 The introduction of FBI

This section briefly introduces the whole system of FBI algorithm. In the whole law enforcement, forensic investigation process is high-frequency and critical. A standard forensic investigation consists of five steps:

1) The start of the investigation: An investigation begins with information found by the first police officer who arrives

Table 1 Nomenclatures

Symbol	Description
X	The total population
X_i	The ith individual of population X
X_{ij}	The jth dimension of ith individual in population X
$f(X_i)$	The fitness of objective function of individual X_i
X_{Best}	The optimal individual in the current population X
X_{Worst}	The worst individual in the current population X
$rand()$	A random number which obeys uniform distribution between [0,1]
g	The current number of iteration
G	The maximum number of iteration
D	The dimension of each individual for objective problem
$STATE_i$	The search tendency of ith individual X_i
NP	The scale of whole population X
NP_A	The scale of individuals in investigation team
NP_B	The scale of individuals in pursuit team
Lb_j	The upper limit of jth dimension of objective problem
Ub_j	The lower limit of jth dimension of objective problem

Algorithm 1 Pseudo-code of original FBI.

1: **for** i from 1 to NP_A **do**
2:     Generate individuals X_i^A.
3: **end for**
4: **for** i from 1 to NP_B **do**
5:     Generate individuals X_i^B.
6: **end for**
7: **repeat**
8:     **for** i from 1 to NP_A **do**
9:         Randomly select two individuals X_{A_1} and X_{A_2}
10:         Calculate new individual $X_i^{A,new}$ by Eq. 1
11:         Evaluate new location and choose the offspring
12:         **if** rand<0.5 **then**
13:             Calculate new individual $X_i^{A,new2}$ by Eq. 3
14:         **else**
15:             Restore to the initial X_i^A
16:         **end if**
17:         Evaluate new location and choose the offspring
18:     **end for**
19:     **for** i from 1 to NP_B **do**
20:         Calculate new individual $X_i^{B,new1}$ by Eq. 4
21:         Evaluate new location and choose the offspring
22:         Calculate new individual $X_i^{B,new2}$ by decision statement Eq. 5
23:         Evaluate new location and choose the offspring
24:     **end for**
25:     Sort and obtain the optimal solution in two populations
26: **until** Reach Terminal Condition

---

at the crime scene, this information determines the main direction of investigation for team members.

2) The interpretation of the findings: The investigation team interprets existing survey results and shares information with other team members. At the same time, team members will attempt to connect the new information with their existing impressions of the case.

3) Directions of inquiry: Based on the interpretation of survey results, team members construct several lines of inquiry (scenarios, criminal motives, and lines of inquiry) and try to reach a consensus when discussing research directions in briefings or brainstorming sessions.

4) Actions: After determining the survey direction and priorities of this investigation, the total team decides on further actions about the criminal investigation. Team members weigh the cost and predicted earnings of a survey direction, and carry out a promising scheme.

5) Prosecution: Until a key suspect has been found, the whole case is almost over.

To reflect the distinct survey direction of team members, FBI divided the whole population into an investigation team and a pursuit team based on the behavior of the police. The members of the investigation team attempt to analyze the possible position of the prime suspect (global optimal) in the whole search space, which can be called 'suspected locations'. After determining the suspected locations with a high possibility, the pursuit team acts on the orders from headquarters and gradually closes to the current suspected locations. In FBI algorithm, the scale of these two team is set to NP_A and NP_B (NP_A=NP_B), each sub-population executes its respective search operations. The details of investigation team and pursuit team can be described as follows.

2.1 Investigation Team

The members of investigation team X^A mainly implement the process of step(2) and step(3). Each individual X_i^A ($i = 1, 2, .., NP_A$) concludes the possible position of the suspect based on two random individuals $X_{r_1}^A$ and $X_{r_2}^A$, the first step

can be expressed as:

$$X_{i,j}^A = X_{i,j}^A + rand * (X_{i,j}^A - F \times (X_{r_1,j}^A + X_{r_2,j}^A)) \tag{1}$$

where F is the positive constant, $i = 1, 2, ..., NP_A$, and j is a random positive integer between $[1, D]$, which represents the position to be modified, the random numbers of individual satisfy uniform distribution between $[1, NP_A]$ and $r_1 \neq r_2 \neq i$. This equation reflects the process 'interpretation of the findings', where the random individual can stimulate disorganized and random information collected by investigation members. The second search model of investigators is 'directions of inquiry'. Team members move towards some promising survey direction with selected dimensions based on the results of group discussion or brainstorming. To embody the feasibility of selected survey direction, a probability parameter is formulated as:

$$Prob(X_i) = \frac{f_{Worst} - f(X_i)}{f_{Worst} - f_{Best}} \tag{2}$$

where f_{Best} and f_{Worst} respectively represents current individual with the best and worst fitness, $f()$ represents the fitness of the individual to be calculated. According to the normalization method, each probability to be optimized of individual is determined by a decision tree and a random number. The selected individual will be calculated by formulation:

$$X_{i,j}^A = X_{Best,j} + X_{r_3,j}^A + rand \times (X_{r_4,j}^A - X_{r_5,j}^A) \tag{3}$$

where r_3, r_4, and r_5 are the random positive integer between $[1, NP_A]$ $(r_3 \neq r_4 \neq r_5 \neq i)$, the whole investigation team respectively carries out two operators one by one.

2.2 Pursuit Team

The pursuit team concludes promising positions based on the population and investigation team's information and move directly toward the speculative location. Different from the investigation team, each dimension of the individual in the pursuit team will be improved by different information between the current optimal individual X_{Best} and the selected individual X_i^B, the process can be formulated as:

$$X_{i,j}^B = rand \times X_{i,j}^B + rand \times (X_{Best,j} - X_{i,j}^B) \tag{4}$$

The pursuit team considers that if the fitness of the updated position according to Eq. 4 is better than the previous state, however, the whole algorithm may converge fastly and fall into stagnation due to the new position mainly depending on the present best individual. Therefore, the second step of the

---

Algorithm 2 Pseudo-code of SaFBI.

1: **for** i from 1 to NP **do**
2:    Randomly generate individuals X_i^1.
3:    Randomly initialize the count $n_i^1 \leftarrow n_0$.
4: **end for**
5: **for** g from 2 to G **do**
6:    **for** i from 1 to NP **do**
7:       **if** $n_i^{g-1} = 1$ **then**
8:          **if** rand<0.35 **then**
9:             Calculate new individual X_i^g by Eq. 4
10:          **else**
11:             Select a random individual X_{r_1}
12:             Calculate new individual X_i^g by Eq. 5
13:          **end if**
14:       **else**
15:          **if** $n_i^{g-1} \leq$ -3 **then**
16:             Select two random individuals X_{r_2} and X_{r_3}.
17:             Calculate new individual X_i^g by Eq. 8
18:          **else**
19:             select a random dimension d between $[1, D]$
20:             **for** j from 1 to D **do**
21:                **if** rand<0.5 or $j = d$ **then**
22:                   Calculate new element $X_{i,j}^g$ by Eq. 3
23:                **else**
24:                   $X_{i,j}^g \leftarrow X_{i,j}^{g-1}$
25:                **end if**
26:             **end for**
27:          **end if**
28:       **end if**
29:       Update the state n_i^g by Eq. 6
30:    **end for**
31:    Evaluation new individuals and select the offspring with better fitness
32: **end for**

---

pursuit team need to strengthen the diversity of the whole population. Except for X_{Best}, team member should take the pursuit information from other members into account. In the course of pursuit action, headquarters constantly receive position information from members and update the current optimal location. Members cooperate with each other according to the instruction of headquarters, which can be depicted by:

$$\begin{aligned}
&if\ f(X_i^B) > f(X_{r_6}^B) \\
&\quad X_i^B = X_i^B + rand(D) \times (X_{r_6}^B - X_i^B) \\
&\qquad\qquad + rand(D) \times (X_{Best} - X_{r_6}^B) \\
&else \\
&\quad X_i^B = X_i^B + rand(D) \times (X_i^B - X_{r_6}^B) \\
&\qquad\qquad + rand(D) \times (X_{Best} - X_i^B)
\end{aligned} \tag{5}$$

where $rand(D)$ represent a D-dimensional vector which is constituted by random numbers between $[0, 1]$, r_6 is a random number with the range of $[1, NP_B]$. In general, the cyclic invocation of investigation and pursuit forms a complete forensic investigation process, which can be shown in Algorithm 1, where the original FBI respectively executes

two updating operators in different populations, then select the individual with better fitness as the offspring in comparison to the newly produced individual. According to the differing characteristics of two populations, we have identified issues with the original FBI and thus made the decision to improve it in regards to the overall architecture and updating operator.

3 Proposed Algorithm SaFBI

3.1 Issue of FBI and Motivation

Although the mechanism of the original FBI is reliable and efficient in solving optimization problems, its ability of exploration and exploitation are still limited under some circumstances. Firstly, updating double populations will slow down the convergence speed of the whole mechanism, which causes a feasible solution that can't be obtained under the standard numbers of iterations. Secondly, although FBI expands a wider search space than conventional algorithms due to a larger population, its exploration still suffers from poor information exchange between two populations. Specifically, in the process of optimizing population A (Investigation team) to generate new individual, the information in population B (Pursuit team) are not considered. Therefore, it is easy for both populations to fall into stagnation. Finally, it is difficult to reflect the different emphasis of two populations on exploration and exploitation due to the updating operators.

For disposing of the specific problems with no prior information, a self-adaptive mechanism is introduced to achieve a better balance between exploration and exploitation. Self-adaptive optimization algorithms dynamically adjust the population structure or parameters to react to some particulars of the problem to be solved. According to the persistent progress and immediate situation of optimization, the adaptive mechanism concentrates on exploring new search space when the population is trapped in local optima. On the contrary, the emphasis of the algorithm tends to exploit the current space when the diversity of the population is high-level. To achieve an excellent balance between exploration and exploitation, we incorporate a dynamic population structure based on self-adaptive parameters into the original FBI, named SaFBI.

3.2 Self-Adaptive Population Mechanism

The original FBI algorithm generates two populations with the same scales, which updates the offspring by different operators respectively. Nevertheless, double population size has no advantage in producing offspring due to poor utilization of individuals from a different population. When the scale of each population is set to less than the standard value, each of them is prone to fall into a local optima in search space, respectively. On the contrary, the convergence speed of the algorithm is not sufficient to obtain a feasible solution within the specified time consumption. Therefore, our improvement is primarily founded on the population structure. Instead of creating two distinct populations and using separate operators for each, we label all individuals in various states to differentiate between the investigation team and the pursuit team. If an individual's evaluation indicator remains unpromising or unchanged for multiple iterations, it signifies stagnation. Thus, operators with robust exploration are utilized to evade local optima, and the individual tag is converted to a member of the investigation team. Alternatively, as an individual steadily improves their fitness or explores new areas, they should be integrated into a pursuit team. To accomplish the self-adaptive population mechanism, a variable parameter n_i^g is produced as the count number to reflect the state of the individual in the continuous optimization process, which can be formulated as

$$n_i^g = \begin{cases} n_i^{g-1} - \tau, & if \ f(X_i^{g-1}) <= f(X_i^g) \\ n_0, & otherwise \end{cases} \qquad (6)$$

where τ represent the strade, and n_0 is the initial value. Different state of individual is dynamically classified as investigation team or pursuit team. The state's change of individual X_i can be represented as the selection of two different updating operator, which can be concretely embodied as

$$X_i^g = \begin{cases} X_i^{g-1} \rightarrow Eq.8, & if \ n_i^{g-1} \le -3 \\ X_i^{g-1} \rightarrow Eq.3, & otherwise \end{cases} \qquad (7)$$

where parameter n satisfies the inequality then the ith individual acts as an investigation team and conversely as a pursuit team. Based on the complexity and difficulty of various optimization functions, individuals deliberately select and implement operators for exploration and exploitation. Our proposed self-adaptive population structure's advantage can be observed in Figs. 1 and 2. The SaFBI algorithm for unimodal functions utilizes a subset of individuals as an investigation team to explore the entire search space, while the rest of the individuals form the pursuit team and exploit the areas that have been investigated. Consequently, SaFBI outperforms the original FBI algorithm in locating the optimal solution swiftly on unimodal problems. In complex optimization problems with multiple modes, SaFBI will continue to explore the entire area with a significant portion of individuals after obtaining local optima. As a result, our proposed self-adaptive population structure can effectively prevent the FBI algorithm from falling into local optima.
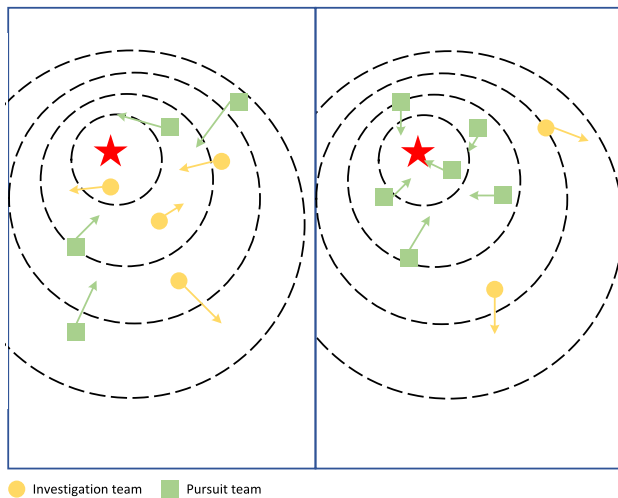
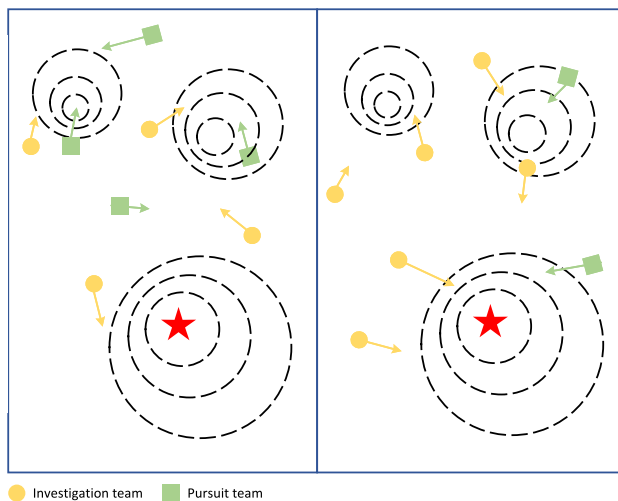**Fig. 1** FBI (left) and SaFBI (right) on unimodal function

Fig. 2 FBI (left) and SaFBI (right) on complex function

In addition to the terrible exploitation ability caused by the population structure, several operators of FBI greatly utilize the location information of current optima so that the exploration ability of each population is low-level. In order to enhance convergence speed when the current optimal solution is slowly optimized in the search space adjacent to the global optima, we modified part of operators for better balance between exploration and exploitation. The investigation individuals tend to seek for an undeveloped region when the algorithm is stuck, so we adjust Eq. 1 to expand the search space.

$$X_i^A = X_i^A + (rand - 0.5) * (1 - \sqrt{g/G}) * (X_i^A - X_{r_1}^A)$$
$$+ (rand - 0.5) * (1 - \sqrt{g/G}) * (X_i^A - X_{r_2}^A) \quad (8)$$

According to the changing weight based on the proportion of g in G, we can obtain a large step to escape from local

optima in the early generation. In the final phase of optimization, the operator will tend to explore the found search space to ensure excellent convergence results. We incorporate self-adaptive population mechanism into original FBI and fine-tune an operator. The pseudo-code of proposed SaFBI is shown in Algorithm 2.

3.3 Time Complexity

We can determine the time complexity of the entire algorithm by following the pseudo-code as a guide. This section lists the computational complexity of SaFBI:

1) The initialization of population needs O(N);

2) Calculating the fitness of population costs O(N);

3) The time complexity of updating population needs O(N);

4) The time complexity of boundary detection is O(N);

5) Selecting the random individuals costs O(N);

6) Replacing or retaining the current individuals needs O(N);

Therefore, the total time complexity of SaFBI is O(N). It is worth mentioning that SaFBI is faster than original FBI on account of the fewer judgment statements based on self-adaptive population mechanism. The actual time cost will be analyzed in detail under the repeated experiments.

4 Experimental Results and Statistical Analysis

To testify the efficiency of our proposed SaFBI in disposing optimization problems with different specialties, the experimental results of CEC2017 benchmark functions are summarized as the comparison between SaFBI and other state-of-the-art algorithms. At first, CEC2017 and CEC2011 benchmark functions are clarified. Secondly, the total results of experiment are listed in table on two function groups. Then, four different illustrations are depicted to verify the performance of the proposed algorithm. Finally, multiple variants of SaFBI with distinct parameter or structure are compared with proposed SaFBI. All experiments are executed by MATLAB on a desktop with Intel(R) Core(TM) i7-9700 CPU @ 3.00 GHz and 8.00 GB of RAM.

4.1 Optimization Problems

We selected 51 benchmark functions from IEEE CEC to assess SaFBI, our proposed algorithm's viability. While 29 functions are from CEC2017, we derived the remaining ones from CEC2011. The CEC2017 functions include two unimodal functions (F1, F2), seven multimodal functions (F3 to F9), ten hybrid functions (F10 to F19), and ten composition functions (F20 to F29). Notably, we discarded the second

Table 2 The experimental results on 29 function of CEC 2017

|	SaFBI MEAN ± STD	FBI MEAN ± STD	RSA MEAN ± STD	INFO MEAN ± STD	DEGLS MEAN ± STD	OLBSO MEAN ± STD	MPSO MEAN ± STD
F1	1.000E+02±1.952E-02	4.530E+03±4.760E+05	1.149E+05±2.100E+05	**1.000E+02±4.017E-09**	7.280E+09±4.194E+09	2.245E+03±2.565E+03	2.225E+08±6.974E+07
F2	3.000E+02±1.238E-05	1.752E+04±2.816E+03	2.469E+04±4.282E+03	**3.000E+02±2.415E-07**	3.004E+02±4.821E-01	3.005E+02±4.197E-01	8.093E+03±2.052E+03
F3	**4.096E+02±2.104E+01**	4.604E+02±3.737E+01	4.877E+02±1.755E+01	4.148E+02±2.490E+01	1.215E+03±4.017E+02	4.715E+02±1.512E+01	6.644E+02±5.783E+01
F4	**5.659E+02±1.667E+01**	5.753E+02±1.000E+01	5.903E+02±1.903E+01	6.399E+02±3.567E+01	6.565E+02±4.479E+01	6.469E+02±3.485E+01	6.771E+02±2.524E+01
F5	**6.000E+02±1.036E-01**	6.002E+02±3.885E-01	6.084E+02±3.546E+00	6.159E+02±8.919E+00	6.306E+02±8.656E+00	6.051E+02±2.604E+00	6.090E+02±2.071E+00
F6	**8.026E+02±1.592E+01**	8.475E+02±1.234E+01	8.303E+02±2.445E+01	9.209E+02±4.404E+01	1.021E+03±8.154E+01	9.014E+02±3.499E+01	9.660E+02±1.993E+01
F7	**8.658E+02±2.104E+01**	8.770E+02±9.911E+00	8.880E+02±2.403E+01	9.185E+02±2.779E+01	9.420E+02±2.683E+01	9.137E+02±2.401E+01	9.698E+02±2.204E+01
F8	**9.102E+02±1.894E+01**	9.671E+02±2.006E+02	1.453E+03±4.145E+02	2.368E+03±6.635E+02	3.373E+03±1.069E+03	3.477E+03±7.127E+02	1.175E+03±1.316E+02
F9	4.918E+03±4.481E+02	**3.895E+03±2.637E+02**	5.659E+03±6.825E+02	5.007E+03±7.947E+02	4.762E+03±6.686E+02	4.702E+03±4.859E+02	7.163E+03±4.857E+02
F10	**1.150E+03±2.952E+01**	1.224E+03±3.471E+01	1.228E+03±4.888E+01	1.243E+03±5.364E+01	1.411E+03±1.190E+02	1.232E+03±3.945E+01	1.347E+03±5.359E+01
F11	2.124E+04±9.917E+03	1.146E+05±8.247E+04	6.602E+05±5.563E+05	**1.417E+04±1.088E+04**	1.283E+08±1.280E+08	1.789E+06±1.274E+06	1.683E+07±1.097E+07
F12	1.297E+04±1.495E+04	1.978E+04±1.905E+04	3.283E+04±3.344E+04	**2.135E+03±5.175E+02**	3.206E+06±2.227E+07	1.993E+04±1.391E+04	2.685E+05±5.572E+05
F13	1.865E+03±1.097E+03	6.404E+03±5.287E+03	1.366E+04±1.050E+04	**1.631E+03±7.177E+01**	2.917E+03±3.695E+03	4.978E+03±2.476E+03	2.132E+04±2.188E+04
F14	3.370E+03±2.872E+03	3.107E+03±2.183E+03	2.029E+04±1.632E+04	**1.909E+03±7.623E+02**	1.227E+04±2.689E+04	6.009E+03±4.882E+03	4.484E+04±1.196E+05
F15	**1.935E+03±2.129E+02**	2.403E+03±1.513E+02	2.563E+03±2.770E+02	2.675E+03±2.507E+02	2.709E+03±2.217E+02	2.872E+03±2.430E+02	2.737E+03±2.994E+02
F16	**1.825E+03±1.104E+02**	1.896E+03±7.763E+01	2.089E+03±1.528E+02	2.214E+03±1.865E+02	2.212E+03±2.056E+02	2.274E+03±2.359E+02	1.908E+03±1.048E+02
F17	4.049E+04±2.415E+04	2.736E+05±1.462E+05	2.794E+05±2.581E+05	8.396E+03±1.044E+04	**7.039E+03±8.185E+03**	1.269E+05±1.077E+05	5.040E+05±3.035E+05
F18	7.544E+03±7.006E+03	6.798E+03±4.566E+03	1.354E+04±1.494E+04	2.042E+03±7.021E+01	9.593E+04±2.341E+05	8.839E+03±8.767E+03	2.515E+04±4.406E+04
F19	**2.154E+03±9.279E+01**	2.180E+03±5.747E+01	2.418E+03±1.488E+02	2.490E+03±1.978E+02	2.450E+03±1.623E+02	2.500E+03±1.687E+02	2.285E+03±9.075E+01
F20	**2.356E+03±1.544E+01**	2.384E+03±9.282E+00	2.394E+03±2.477E+01	2.407E+03±4.054E+01	2.442E+03±3.737E+01	2.460E+03±3.842E+01	2.473E+03±2.293E+01
F21	2.300E+03±8.681E-01	**2.300E+03±8.668E-01**	4.843E+03±2.572E+03	3.788E+03±2.085E+03	4.998E+03±1.706E+03	4.187E+03±1.977E+03	2.382E+03±1.288E+01
F22	**2.692E+03±1.472E+01**	2.749E+03±1.393E+01	2.756E+03±2.746E+01	2.795E+03±4.095E+01	2.892E+03±6.950E+01	2.904E+03±7.282E+01	2.876E+03±3.664E+01
F23	**2.872E+03±1.616E+01**	2.958E+03±1.991E+01	2.928E+03±3.367E+01	2.954E+03±3.742E+01	3.084E+03±6.802E+01	3.061E+03±5.596E+01	3.043E+03±3.259E+01
F24	2.886E+03±3.191E+00	2.885E+03±1.796E+00	2.887E+03±2.841E+00	2.893E+03±1.477E+02	3.206E+03±1.681E+02	**2.881E+03±1.121E+01**	2.951E+03±1.783E+01
F25	3.557E+03±7.431E+02	**3.184E+03±6.993E+02**	4.651E+03±4.503E+02	5.281E+03±9.976E+02	6.257E+03±7.163E+02	6.145E+03±1.541E+03	3.417E+03±7.204E+02
F26	3.218E+03±1.370E+01	3.238E+03±1.103E+01	**3.205E+03±1.263E+01**	3.242E+03±2.837E+01	3.326E+03±4.400E+01	3.336E+03±8.577E+01	3.364E+03±2.939E+01
F27	**3.112E+03±3.360E+01**	3.189E+03±4.718E+01	3.225E+03±2.140E+01	3.119E+03±4.755E+01	3.945E+03±3.757E+02	3.184E+03±5.186E+01	3.332E+03±3.423E+01
F28	**3.404E+03±9.110E+01**	3.597E+03±8.091E+01	3.775E+03±1.688E+02	3.946E+03±2.596E+02	4.128E+03±2.891E+02	4.346E+03±3.047E+02	3.786E+03±1.641E+02
F29	7.085E+03±2.320E+03	2.395E+04±1.565E+04	2.748E+04±2.588E+04	**5.676E+03±8.346E+02**	3.255E+06±6.101E+06	3.937E+05±2.336E+05	4.594E+05±4.220E+05
W/T/L	–	24/3/2	27/1/1	18/3/8	26/0/3	26/1/2	28/1/0
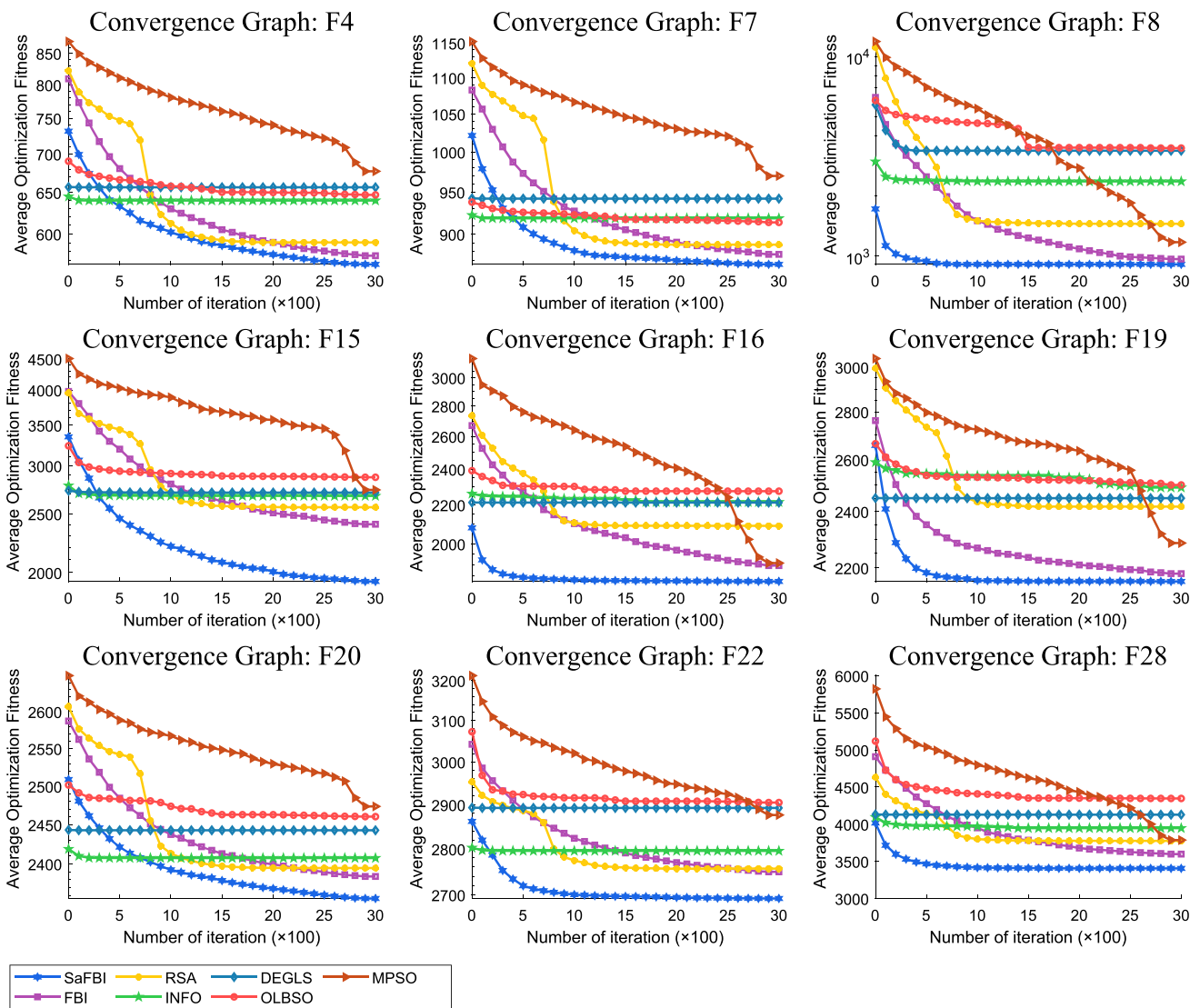
Fig. 3 Convergence graphs of SaFBI and six comparatives on nine CEC functions

original function due to its instability. To further underscore the robustness and flexibility of our proposed algorithm, we have selected some intricate problems as additional evaluation criteria. 14 real-world problems are modeled as 22 optimization functions in CEC2011 with different dimensions and constraints, which consist of parameter estimation problem of frequency-modulated sound waves (F30), two potential energy minimization problems (F31, F34 and F35), two optimal control problems (F32, F33), spread spectrum radar polly-phase code design (F36), transmission network expansion planning problem (F37), large scale transmission pricing problem (F38), circular antenna array design (F39), two hourly dispatch problems (F40 to F46), hydrothermal scheduling problem (F47 to F49), and two spacecraft trajectory design problems (F50, F51). Except for 51 CEC benchmark functions, we attach two complete engineering

optimization problems to further reflect the high-efficiency and robustness of SaFBI.

4.2 Experimental Result on CEC2017

For the sake of fairness, parameters of each function are set to the same standard in CEC2017. The initial population size is fixed as 100, the dimension of each individual is set to 30, the lower and upper bounds of element in individual are set to -100 and 100 respectively, the maximal number of function evaluation is 3×10^5, and the run time for functions are fixed as 51 due to the randomicity of meta-heuristic algorithm. Except for original FBI, two state-of-the-art algorithms and three first-class variants of classical algorithm are selected as competitors. We compare SaFBI with other six algorithms, namely, FBI, reptile search algorithm (RSA) [35], innovative
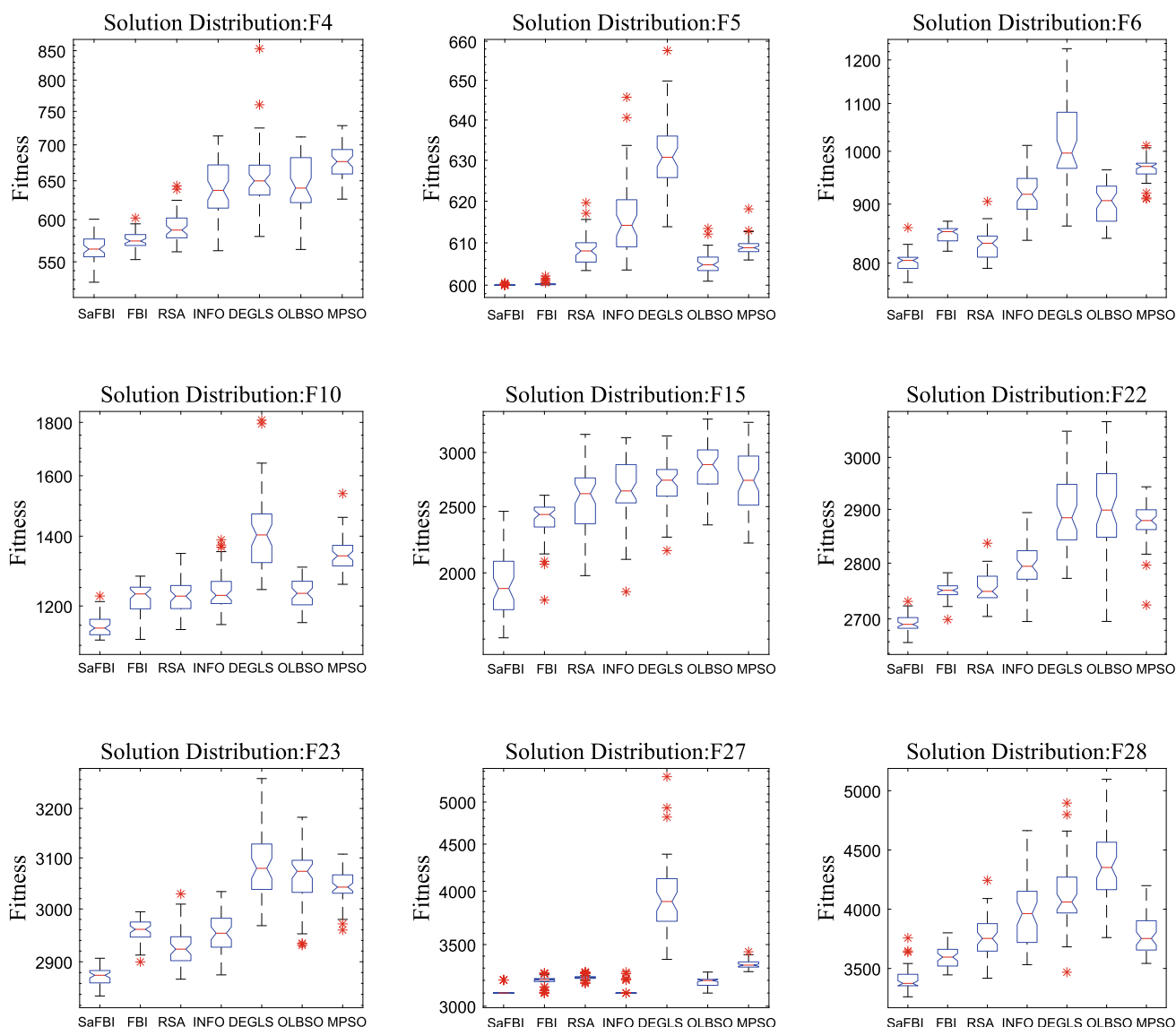
Fig. 4 Box-and-whisker plots of SaFBI and six competitors on nine CEC functions

optimizer based on weighted mean of vectors (INFO) [36], global optimum-based search differential evolution(DEGLS) [37], orthogonal Learning-based brain storm optimization (OLBSO) [38], and modified particle swarm optimization (MPSO) [39] to verify the performance of proposed SaFBI. The experimental results are shown in Table 2, where each set of data is formed from arithmetic mean and standard deviation with scientific notation, which is obtained by 51 identical tests. Note that the best value among all compared algorithms in these tables are shown in bold. All functions belong to minimum optimization and the global optimal value of Fi are set to $i \times 100$, and the value in bold is the optimum in all algorithms. W, T, L denotes the numbers of 29 functions that our proposed SaFBI performs better, tied, worse than the competitor from the result of Wilcoxon-sign-test, respectively. It

is observed that the performance of SaFBI dominates other competitors in solving optimization function with different types.

4.3 Convergence Graphs and Box-and-Whisker Plots

In order to specify the entire convergence process of each algorithm and the comparison results between SaFBI and other competitors, convergence curves of different functions are illustrated in Fig. 3, where X-axis denotes the number of current iteration ($\times 100$) and Y-axis represents the mean evaluation value of 51 independent experiments. Obviously, SaFBI is still hard to trap into stagnation in despite of a fast convergence speed, and the optimal solutions under 3000 iterations are superior to other competitors. In addition to

Table 3 The experimental results on 22 real-world functions

Algorithm	F30	F31	F32	F33	F34	F35
SaFBI	**8.064E+00±6.592E+00**	-2.506E+01±2.262E+00	**1.151E-05±2.520E-019**	1.607E+01±2.848E+00	-3.437E+01±1.313E+00	-2.597E+01±2.566E+00
FBI	9.880E+00±6.214E+00	-1.871E+01±1.583E+00	1.151E-05±4.960E-019	1.590E+01±2.081E+00	**-3.514E+01±7.237E-01**	**-2.898E+01±3.132E-01**
RSA	1.566E+01±6.231E+00	-2.358E+01±4.613E+00	1.151E-05±3.954E-019	1.624E+01±3.144E+00	-3.362E+01±1.619E+00	2.000E+030±7.611E+030
HGS	1.406E+01±7.284E+00	-1.599E+01±6.161E+00	1.151E-05±1.287E-09	**1.425E+01±1.892E-01**	-3.272E+01±2.504E+00	-2.653E+01±3.038E+00
SMA	1.746E+01±6.830E+00	-2.151E+01±3.024E+00	1.151E-05±3.641E-018	1.491E+01±2.055E+00	-2.849E+01±4.124E+00	-1.975E+01±3.059E+00
INFO	1.408E+01±6.154E+00	**-2.638E+01±3.394E+00**	1.151E-05±3.328E-019	1.435E+01±1.585E-01	-3.351E+01±2.561E+00	-2.096E+01±2.256E+00
DEGLS	1.560E+01±5.285E+00	-1.125E+01±2.691E+00	1.151E-05±2.818E-019	1.685E+01±3.419E+00	-2.815E+01±3.876E+00	6.000E+030±1.220E+031
MPSO	1.544E+01±4.372E+00	-1.335E+01±1.636E+00	1.151E-05±1.031E-013	1.486E+01±5.950E-01	-2.998E+01±1.889E+00	-2.201E+01±2.268E+00

Algorithm	F36	F37	F38	F39	F40	F41
SaFBI	1.176E+00±1.037E-01	**2.200E+02±0.000E+00**	**2.212E+03±5.948E+02**	**-2.130E+01±5.852E-01**	5.308E+04±1.787E+03	1.743E+07±5.028E+04
FBI	1.287E+00±8.106E-02	**2.200E+02±0.000E+00**	6.082E+03±1.294E+04	-2.117E+01±1.126E+00	5.283E+04±2.697E+02	1.788E+07±4.674E+04
RSA	1.277E+00±1.917E-01	2.274E+02±1.199E+01	6.604E+04±5.002E+04	-1.307E+01±2.391E+00	**5.217E+04±7.563E+02**	1.758E+07±8.604E+04
HGS	9.305E-01±1.400E-01	**2.200E+02±0.000E+00**	3.908E+03±1.953E+03	-1.382E+01±2.927E+00	5.247E+04±5.157E+02	**1.737E+07±4.556E+04**
SMA	1.009E+00±1.683E-01	2.431E+02±1.796E+01	9.954E+03±4.689E+03	-1.325E+01±3.229E+00	6.663E+04±1.824E+04	1.880E+07±2.045E+05
INFO	1.035E+00±1.745E-01	**2.200E+02±0.000E+00**	4.501E+03±2.499E+03	-1.520E+01±4.157E+00	5.244E+04±4.770E+02	1.743E+07±5.293E+04
DEGLS	**9.238E-01±1.166E-01**	3.017E+02±1.701E+02	1.590E+06±1.498E+05	-1.488E+01±3.917E+00	7.369E+06±6.418E+06	4.409E+07±1.362E+06
MPSO	1.580E+00±1.721E-01	**2.200E+02±0.000E+00**	1.272E+06±1.313E+05	-1.796E+01±8.983E-01	2.511E+05±9.821E+04	3.256E+07±6.270E+05

Algorithm	F42	F43	F44	F45	F46	F47
SaFBI	1.545E+04±1.400E+01	**1.830E+04±7.451E+01**	**3.289E+04±3.576E+01**	**1.306E+05±1.826E+03**	**1.912E+06±1.185E+04**	**9.436E+05±2.911E+03**
FBI	**1.545E+04±6.605E+00**	1.916E+04±1.454E+02	3.303E+04±5.109E+01	1.357E+05±1.862E+03	1.966E+06±3.958E+04	3.286E+06±1.475E+06
RSA	1.553E+04±4.408E+01	1.922E+04±1.843E+02	3.302E+04±1.475E+02	1.394E+05±7.538E+03	1.948E+06±3.075E+04	9.548E+05±5.509E+03
HGS	1.548E+04±3.542E+01	1.914E+04±1.988E+02	3.293E+04±1.015E+02	1.342E+05±2.788E+03	1.948E+06±2.504E+04	9.522E+05±3.581E+03
SMA	1.549E+04±3.104E+01	1.881E+04±1.105E+02	3.312E+04±9.764E+01	1.434E+05±7.578E+03	2.158E+06±2.848E+05	1.006E+06±1.602E+05
INFO	1.548E+04±2.730E+01	1.898E+04±2.055E+02	3.295E+04±6.907E+01	1.378E+05±6.398E+03	1.931E+06±2.419E+04	9.824E+05±1.130E+05
DEGLS	1.548E+04±2.211E+01	1.855E+04±1.766E+02	3.289E+04±1.104E+02	1.340E+05±4.518E+03	4.921E+07±2.586E+08	1.051E+07±1.026E+07
MPSO	1.547E+04±2.574E+01	1.936E+04±1.931E+02	3.342E+04±4.621E+02	1.405E+05±2.608E+03	2.232E+06±3.123E+05	8.441E+06±4.489E+06

Algorithm	F48	F49	F50	F51	W/T/L
SaFBI	1.216E+06±1.332E+05	**9.475E+05±2.104E+04**	**1.281E+01±2.650E+00**	1.975E+01±2.321E+00	-
FBI	4.205E+06±1.697E+06	3.344E+06±1.490E+06	2.180E+01±2.428E+00	2.043E+01±2.923E+00	14/8/0
RSA	**1.018E+06±7.855E+04**	9.544E+05±4.573E+03	2.150E+01±6.223E+00	2.221E+01±6.994E+00	17/3/2
HGS	1.133E+06±9.932E+04	9.512E+05±5.682E+03	1.607E+01±1.753E+00	**1.822E+01±4.806E+00**	16/4/2
SMA	1.441E+06±2.400E+05	1.016E+06±1.635E+05	2.872E+01±6.531E+00	2.658E+01±4.531E+00	17/4/0
INFO	1.251E+06±2.157E+05	1.023E+06±3.268E+05	1.649E+01±3.090E+00	2.095E+01±3.431E+00	14/7/1
DEGLS	1.248E+07±1.057E+07	9.553E+06±1.107E+07	2.065E+01±5.538E+00	2.662E+01±4.689E+00	19/2/1
MPSO	1.082E+07±5.529E+06	9.580E+06±4.836E+06	1.981E+01±3.707E+00	2.358E+01±1.875E+00	17/5/0

the convergence results, we can find that SaFBI is promising during the following optimization process after constrained number of iterations for disposing functions F4, F7, F15, F20, and so on. The better optimum can be obtained under the greater number of iterations.

Compared with the characteristics of convergence curve, box-and-whisker plot tends to judge performance by several indicators, which is shown in Fig. 4. In this figure, the plus signs are outliers which can be utilized to reflect the stability of algorithm for many repeated tests under the same conditions, the upper and lower black lines denote the maxima and minima, respectively. The top and bottom sides of a polygon are the quantiles, and the medium line is median. When the box-plot is drawn as a line, it means that the corresponding algorithm is stable and determinate. It can be noted that the plots of SaFBI perform outstanding with the lowest region and the smallest span, which manifest that our proposed mechanism alleviates the randomness of metaheuristic algorithms under the premise of superb convergence results. Excellent performance testifies the high-level balance between exploration and exploitation of our proposed SaFBI.

4.4 Experimental Result on CEC2011

Since the dimensions of the CEC2017 standard dataset are mostly fixed and do not reflect the robustness of proposed SaFBI. We choose 22 real-world optimization problems with unique dimensions from CEC2011. For each function of CEC2011, the elements of each dimension in an individual are optimized under different constraints. Therefore, a distinguished performance in dealing with the real-world problems of CEC2011 can better reflect the effectiveness and extensibility of algorithm. Taking the characteristic into consideration, the parameter are set to follows: population size is fixed as 100; the maximal number of ith function evaluation is $D_i \times 10^4$, where D_i denotes the dimension of ith function and each algorithm used for comparison run 30 times independently. It is worth mentioning that OLBSO is excluded in testing CEC2011 functions because of the poor performance, which is caused by the segmentation mechanism in OLBSO limited in real-world optimization with different dimensions. To reflect the distinguished performance of proposed SaFBI, two state-of-the-art algorithms for engineering optimization are appended to experiment. The experimental results are listed as Table 3, where HGS and SMA respectively denotes hunger games search [40] and slime mould algorithm [41]. From this table, it is obvious that proposed SaFBI performs excellent in all the real-world optimization problems. According to the mean, variance and result of Wilcoxon's sign rank test, we can conclude that the performance of SaFBI is superior to other competitors.

4.5 Application for Engineering Optimization Problem

Different from the simple CEC benchmark optimization problems, the complete real-world engineering problems have numerous complex constraints, such as range and type of value, combination of values from different dimensions and so on. Therefore, it is crucial for a newly proposed algorithm to perform excellent in real-world engineering problems. In this paper, two well-known optimization design are added to verify the performance of SaFBI. These engineering problems can be illustrated as Figs. 5 and 6 named pressure vessel design problem and welded beam design problem, respectively.

4.5.1 Pressure Vessel Design Problem

A pressure vessel (PV) refers to a sealed equipment containing gas or liquid under the determinate load, which plays a significant role in civil, industry, military and chemistry fields. However, the pressure difference between the interior and exterior of PV is exceeding dangerous, the design of PV is subject to strict standard control. In this paper, a modeling of cylindrical vessel is selected as the additional experiment to demonstrate the effectiveness of SaFBI for engineering optimization problem [42]. The optimization formulation of PV design can be depicted as:

$$
\begin{aligned}
F_{PV} &= 0.6224 T_s R L + 1.7781 T_h R^2 + 3.1661 T_s^2 L + 19.84 T_s^2 R \\
&\quad Subject\ to: \\
&\qquad\qquad T_s \geq 0.0193 R \\
&\qquad\qquad T_h \geq 0.00954 R \\
&\qquad R^2 L + 4\pi R^3/3 \geq 1296000 \\
&\quad 0.0625 \times 1 \leq T_s, T_h \leq 0.0625 \times 99 \\
&\qquad\qquad 10 \leq R, L \leq 200
\end{aligned}
\tag{9}
$$

where four parameters thickness T_s, thickness of head T_h, inner radius R and length of cylindrical section L_c are considered as 4-dimensional vectors for input, and F_{PV} denotes the cost of solution.

4.5.2 Welded Beam Design Problem

The other engineering optimization we chose is a minimization problem welded beam (WB) design [45], which obtains the minimum cost of WB. During optimization process, all the design variables can be modeled as four parameters containing thickness of weld T_w, length of welded joint L_w, width of beam W, and thickness of beam T_b. The optimization objective and constraint condition can be formulated as:

Table 4 The Experimental Results of Pressure Vessel Design Problem

Algorithm	T_s	T_h	R	L	Optimal cost	Time	Rank
FBI	0.7500	0.3750	38.8587	221.4117	5.8511E+03	0.93s	4
SaFBI	0.7500	0.3750	38.8601	221.3655	**5.8504E+03**	**0.84s**	1
RSA	0.8750	0.4375	44.4173	160.3180	6.4761E+03	6.15s	8
HGS	0.8750	0.4375	45.2361	141.2957	6.1024E+03	1.55s	6
SMA	0.7500	0.3750	38.8573	221.4362	5.8513E+03	7.08s	3
INFO	0.7500	0.3750	38.8601	221.3655	**5.8504E+03**	5.95s	2
OPA [43]	0.8125	0.4375	42.0984	176.6365	6.0597E+03	1.44s	7
CSA [44]	0.7500	0.3750	38.8525	221.4818	5.8514E+03	0.96s	5
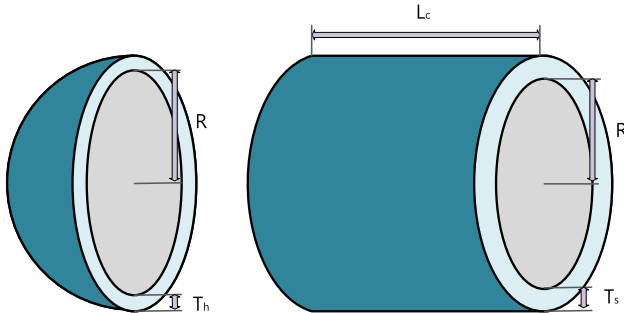
Fig. 5 The illustration of pressure vessel design problem
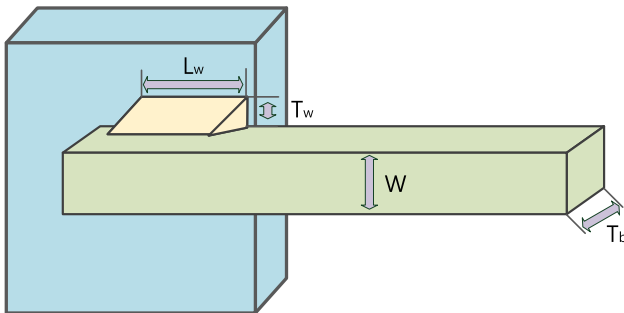
Fig. 6 The illustration of welded beam design problem

$$F_{WB} = 1.10471 T_w^2 L_w + 0.04811 W T_b (L_w + 14)$$
$$Subject\ to:$$
$$W^2 T_b \geq 168$$
$$T_w \leq T_b$$
$$\frac{2.1952}{W^3 T_b} \leq \frac{1}{4}$$
$$(1 - 0.0282346W)W T_b^3 \geq \frac{1000}{10791}$$
$$(\tau')^2 + (\tau'')^2 + T_w \frac{\tau' \tau''}{\sqrt{0.25((T_w + W)^2 + L_w^2)}} \leq 13600^2$$

$$(10)$$

where

$$\tau' = \frac{6000}{\sqrt{2} T_w L_w}$$

$$\tau'' = 6000(14 + 0.5L_w)\frac{\sqrt{0.25((T_w + W)^2 + L_w^2)}}{\sqrt{2} T_w L_w \left(\frac{L_w^2}{12} + \left(\frac{T_w + L_w}{2}\right)^2\right)}$$

$$(11)$$

In addition, $0.125 \leq T_w \leq 5, 0.1 \leq L_w, W, T_b \leq 10$, it is worth mentioning that both T_w and L_w are multiples of 0.0065.

4.5.3 Experimental Results on Engineering Problems

In this experiment, we testify the performance of proposed SaFBI in comparison with several state-of-the-art algorithms designed to solve engineering problems. Each algorithm runs independently 30 times, the optimal solutions and costs are listed in Tables 4 and 5, where *Rank* represents the rank of different algorithms based on Friedman test at the level of α = 0.05. In both Tables 4 and 5, SaFBI all obtains the optimal solutions on two different engineering problems. Meanwhile, the time consumption of SaFBI is extremely low compared with other competitors. This experiment certifies the effectiveness and robustness of our proposed algorithm in solving real-world engineering problems.

4.6 Discussion

We utilize two CEC benchmark optimization function datasets, a pressure vessel optimization problem, and a welded beam design problem to testify the effectiveness of SaFBI in comparison with original FBI and other state-of-the-art algorithms. In this section, the process and performance of proposed SaFBI are analyzed in detail when dealing with optimization problems.

Table 5 The experimental results of welded beam design problem

Algorithm	T_w	L_w	W	T_b	Optimal cost	Time	Rank
FBI	0.2444	6.2177	8.2910	0.2444	2.3813	13.08s	3
SaFBI	0.2444	6.2175	8.2915	0.2444	**2.3810**	**0.39s**	1
RSA	0.2250	6.6733	8.5988	0.2433	2.4540	6.05s	8
HGS	0.2423	6.2804	8.3062	0.2443	2.3870	1.79s	7
SMA	0.2444	6.2215	8.2873	0.2446	2.3830	7.17s	6
INFO	0.2444	6.2175	8.2915	0.2444	**2.3810**	6.06s	2
OPA	0.2443	6.2178	8.2939	0.2444	2.3818	8.21s	4
CSA	0.2440	6.1949	8.3309	0.2441	2.3831	1.08s	5
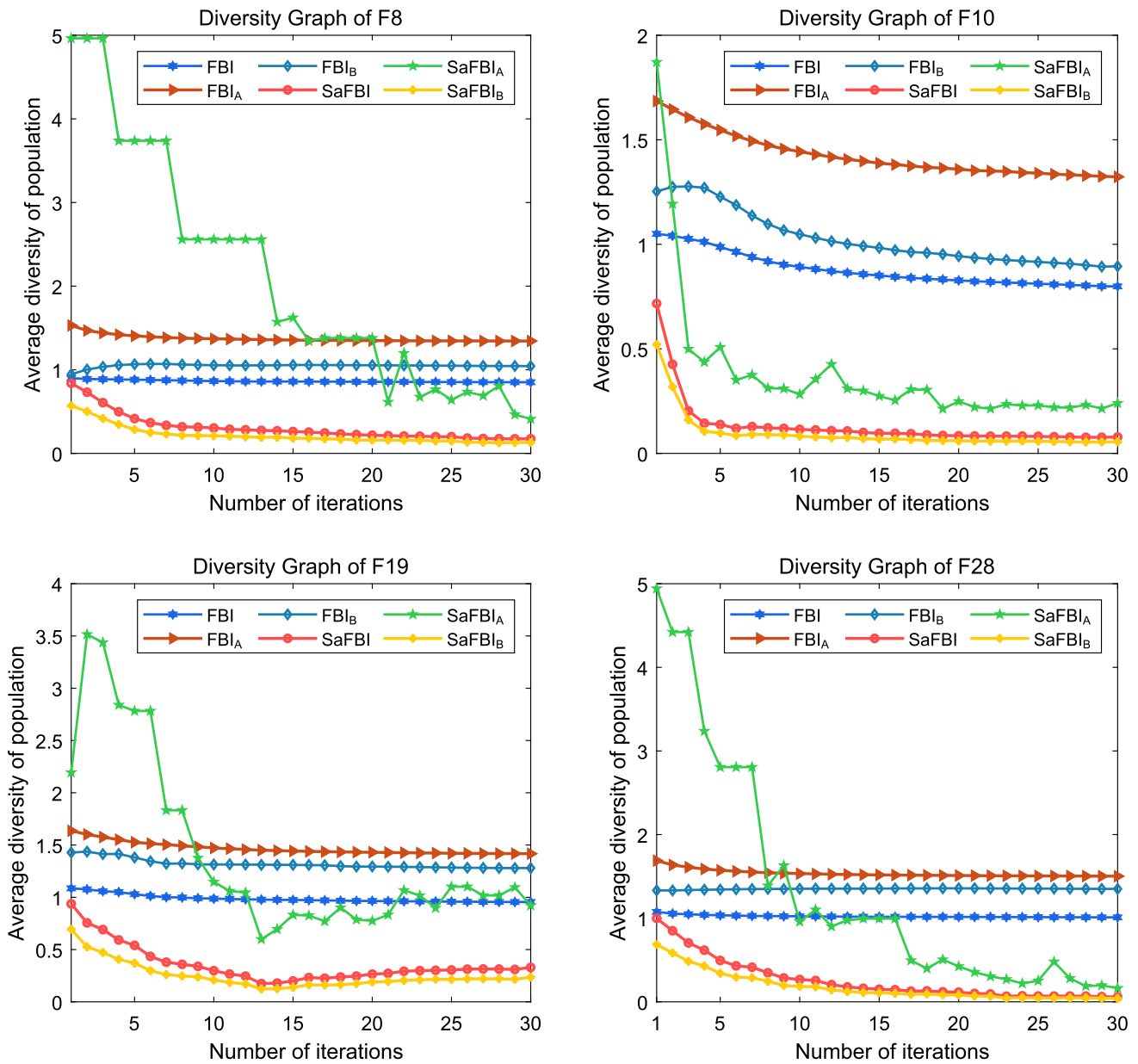
Fig. 7 Diversity of population in SaFBI under the different iterations on three CEC functions

F10

F23

Number of iterations: 300

Number of iterations: 300

Number of iterations: 600

Number of iterations: 1500

Number of iterations: 2400

Number of iterations: 3000
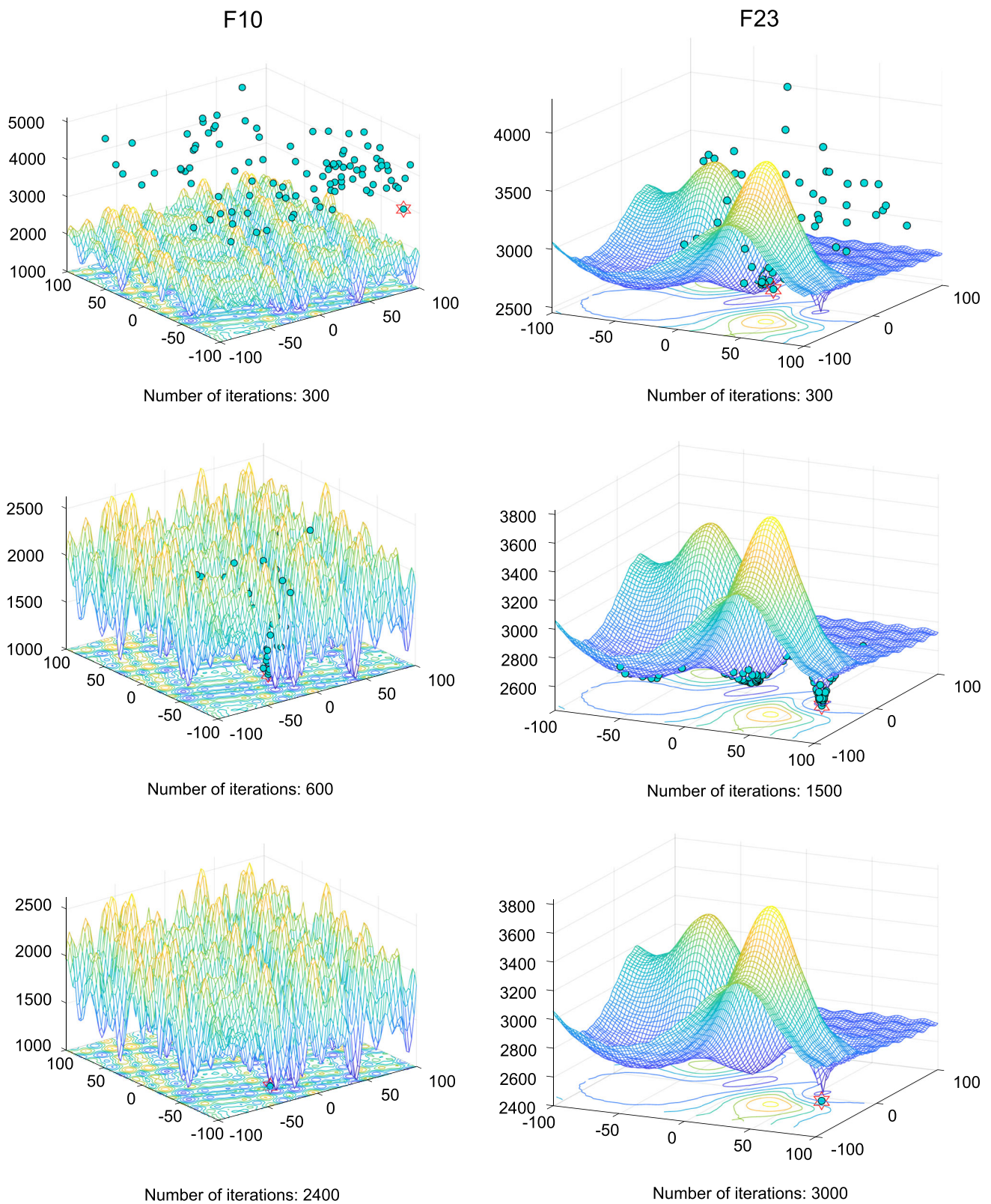
Fig. 8 3-dimensional conformation with contour map of SaFBI under the different iterations on three CEC functions

4.6.1 Population Diversity

In the process of biological evolution, population evolves continuously when it maintains a high-level diversity, which increases with the expansion of spatial distribution in population. When dealing with function optimization by population-based algorithms, the level of population diversity directly reflects the exploration ability of an algorithm. We utilize squared deviation to describe the spatial distribution of population, which can be formulated as:

$$D(X^g) = \frac{\sqrt{\sum_{i=1}^{N} \parallel X_i^g - \bar{X}^g \parallel}}{N} \tag{12}$$

where $\parallel X_i^g - \bar{X}^g \parallel$ is euclidean distance between ith individual in gth iteration and arithmetic mean of the whole population in gth iteration. We plot four convergence graphs in Fig. 7 and each subgraph includes six diversity curves of different population types with the increase of iterations. In these figures, it can be observed that the differences between population diversity of investigation team and pursuit team in original FBI are modest. FBI maintains similar population diversity when it conducts functions with different complexity, which causes the poor overall performance. On the contrary, the self-adaptive population structure of our proposed SaFBI can improve significantly the exploration of investigation team and the exploitation of pursuit team to achieve a better balance than the original algorithm.

4.6.2 3-Dimensional Conformation and Contour Map

Contour map refers to a closed curve connected by adjacent points of equal height on a topographic map. The first two dimensions of the optimization function are selected as two axes, and each individual is distributed in different region within the coordinate axis. However, two-dimensional plane can't reflect whether the location of population is global optimal. Therefore, the evaluation fitness of the current population is taken as the third dimension, and the 3-dimensional conformation with contour map is described as Fig.8. In this figure, we draw the population transformation optimized by SaFBI as the number of iterations increase on two different functions. From the F10 on the left, it is observed that SaFBI can fastly converge with excellent exploitation when tackles the simple function. On the contrary, a complex function with several extremums in different regions needs remarkable exploration. From the right of Fig. 8, we can find that most of individuals are trapped into the local optima in the beginning. However, several individuals later jump out of local optima and seek for the global optima, which reflects that proposed self-adaptive mechanism can effectively keep

balance between exploration and exploitation to improve the performance.

5 Conclusions

This paper proposes a self-adaptive forensic-based investigation algorithm with dynamic structure. In the original FBI, the whole population is divided into two sub-populations (investigation team and pursuit team) to generate the offspring by distinct operators. According to the shortcomings of original population structure in disposing complex and specific optimization, a self-adaptive state vector has been utilized to represent the variation tendency of each individual. The whole population structure will be constantly adjusted under different situations during the iteration, which better matches the peculiarity of problem to be optimized. In order to further maintain the balance between exploration and exploitation of proposed SaFBI, two operators are modified to reflect the individual personalities of different teams. To verify the performance of SaFBI, 51 CEC benchmark functions and two engineering optimization problems are selected as experiment targets, nine state-of-the-art original algorithms are tested as competitors. The experimental results and statistical graphs unambiguously indicate the absolute dominance of our proposed SaFBI in comparison to other competitors. SaFBI demonstrates fast convergence speeds when solving simple unimodal functions. When solving complex hybrid functions, SaFBI displays excellent exploration capabilities. The extensibility of SaFBI is showcased by its ability to optimize real-world problems with varying dimensions and constraints. In addition to superior performance, SaFBI has an acceptable time complexity compared to certain inflated variants of classical algorithms. In future research, we will consider more informative interactions and expand the application field, including wind farm layout optimization [46, 47], reservoir operation optimization [48, 49], and economic load dispatch problem [50, 51], and so on. Additionally, we plan to incorporate the Pareto frontier into SaFBI and apply it to a multi-objective optimization algorithm [52, 53].

Author Contributions Pengxing Cai: Conceptualization, Writing- original draft, Methodology and Software. Yu Zhang: Software, Writing-review & editing, Methodology. Ting Jin: Writing- review & editing, Methodology. Yuki Todo: Writing- Reviewing and Editing. Shangce Gao: Conceptualization, Writing- review & editing, Methodology, Software, Supervision.

Declarations

Conflict of Interest Title: Self-adaptive forensic-based investigation algorithm with dynamic population for solving engineering problems The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper. Author names: Pengxing Cai, Yu Zhang, Ting Jin, Yuki Todo, and Shangce Gao

Author Agreement Statement We confirm that the manuscript has been read and approved by all named authors and that there are no other persons who satisfied the criteria for authorship but are not listed. We further confirm that the order of authors listed in the manuscript has been approved by all of us. We understand that the Corresponding Author is the sole contact for the Editorial process. He/she is responsible for communicating with the other authors about progress, submissions of revisions and final approval of proofs. Author names: Pengxing Cai, Yu Zhang, Ting Jin, Yuki Todo, and Shangce Gao

Ethical Approval No human or animal subjects were involved in this experiment.

Consent to Participate No human subjects were involved in this experiment.

Consent to publish Not applicable.

References

1. Pozna, C., Precup, R.E., Horváth, E., Petriu, E.M.: Hybrid particle filter-particle swarm optimization algorithm and application to fuzzy controlled servo systems. IEEE Trans. Fuzzy Syst. **30**(10), 4286–4297 (2022)
2. Li, T., Shi, J., Deng, W., Hu, Z.: Pyramid particle swarm optimization with novel strategies of competition and cooperation. Appl. Soft Comput. **121**, 108731 (2022)
3. Zhou, X., Ma, H., Gu, J., Chen, H., Deng, W.: Parameter adaptation-based ant colony optimization with dynamic hybrid mechanism. Eng. Appl. Artif. Intell. **114**, 105139 (2022)
4. Yavuz, G., Durmuş, B., Aydın, D.: Artificial bee colony algorithm with distant savants for constrained optimization. Appl. Soft Comput. **116**, 108343 (2022)
5. Wang, C., Song, W., Shen, P.: A new bat algorithm based on a novel topology and its convergence. J. Comput. Sci. **66**, 101931 (2023)
6. Ahmadi, B., Giraldo, J.S., Hoogsteen, G.: Dynamic hunting leadership optimization: algorithm and applications. J. Comput. Sci. **69**, 102010 (2023)
7. Chauhan, S., Vashishtha, G.: A synergy of an evolutionary algorithm with slime mould algorithm through series and parallel construction for improving global optimization and conventional design problem. Eng. Appl. Artif. Intell. **118**, 105650 (2023)
8. Wang, K., Li, X., Gao, L., Li, P., Gupta, S.M.: A genetic simulated annealing algorithm for parallel partial disassembly line balancing problem. Appl. Soft Comput. **107**, 107404 (2021)
9. Htay, K.M., Othman, R.R., Amir, A., Alkanaani, J.M.H.: Gravitational search algorithm based strategy for combinatorial t-way test suite generation. J. King Saud Univ. Comput. Inf. Sci. **34**(8), 4860–4873 (2022)
10. Kaveh, A., Akbari, H., Hosseini, S.M.: Plasma generation optimization: a new physically-based metaheuristic algorithm for solving constrained optimization problems. Eng. Comput. (2020)
11. Gupta, S., Su, R.: Diversity-enhanced modified sine cosine algorithm and its application in solving engineering design problems. J. Comput. Sci. **72**, 102105 (2023)
12. Deng, W., Ni, H., Liu, Y., Chen, H., Zhao, H.: An adaptive differential evolution algorithm based on belief space and generalized opposition-based learning for resource allocation. Appl. Soft Comput. **127**, 109419 (2022)
13. Squires, M., Tao, X., Elangovan, S., Gururajan, R., Zhou, X., Acharya, U.R.: A novel genetic algorithm based system for the scheduling of medical treatments. Expert Syst. Appl. **195**, 116464 (2022)
14. Li, D., Liu, S., Gao, F., Sun, X.: Continual learning classification method with constant-sized memory cells based on the artificial immune system. Knowl.-Based Syst. **213**, 106673 (2021)
15. Sapin, E., De Jong, K.A., Shehu, A.: From optimization to mapping: An evolutionary algorithm for protein energy landscapes. IEEE/ACM Trans. Comput. Biol. Bioinf. **15**(3), 719–731 (2016)
16. Jiang, C., Yang, R.Q., Yuan, B.: An evolutionary algorithm with indirect representation for droplet routing in digital microfluidic biochips. Eng. Appl. Artif. Intell. **115**, 105305 (2022)
17. Camero, A., Arellano-Verdejo, J., Alba, E.: Road map partitioning for routing by using a micro steady state evolutionary algorithm. Eng. Appl. Artif. Intell. **71**, 155–165 (2018)
18. Chauhan, S., Singh, M., Aggarwal, A.K.: Designing of optimal digital IIR filter in the multi-objective framework using an evolutionary algorithm. Eng. Appl. Artif. Intell. **119**, 105803 (2023)
19. Črepinšek, M., Liu, S.H., Mernik, M.: Exploration and exploitation in evolutionary algorithms: a survey. ACM Comput. Surv. (CSUR) **45**(3), 1–33 (2013)
20. Gao, S., Yu, Y., Wang, Y., Wang, J., Cheng, J., Zhou, M.: Chaotic local search-based differential evolution algorithms for optimization. IEEE Trans. Syst. Man Cybern. : Syst. (2019)
21. Belagoune, S., Bali, N., Atif, K., Labdelaoui, H.: A discrete chaotic Jaya algorithm for optimal preventive maintenance scheduling of power systems generators. Appl. Soft Comput. **119**, 108608 (2022)
22. Alyasseri, Z.A.A., Al-Betar, M.A., Awadallah, M.A., Makhadmeh, S.N., Abasi, A.K., Doush, I.A., et al.: A hybrid flower pollination with β-hill climbing algorithm for global optimization. J. King Saud Univ. -Comput. Inf. Sci. **34**(8), 4821–4835 (2022)
23. Boumedine, N., Bouroubi, S.: Protein folding in 3D lattice HP model using a combining cuckoo search with the Hill-Climbing algorithms. Appl. Soft Comput. **119**, 108564 (2022)
24. Zhang, K., Huang, Q., Zhang, Y.: Enhancing comprehensive learning particle swarm optimization with local optima topology. Inf. Sci. **471**, 1–18 (2019)

25. Lei, Z., Gao, S., Gupta, S., Cheng, J., Yang, G.: An aggregative learning gravitational search algorithm with self-adaptive gravitational constants. Expert Syst. Appl. **152**, 113396 (2020)

26. Bi, J., Yuan, H., Zhai, J., Zhou, M., Poor, H.V.: Self-adaptive bat algorithm with genetic operations. IEEE/CAA J. Autom. Sinica. **9**(7), 1284–1294 (2022)

27. Tao, R., Meng, Z., Zhou, H.: A self-adaptive strategy based firefly algorithm for constrained engineering design problems. Appl. Soft Comput. **107**, 107417 (2021)

28. Chen, Z., Liu, Y., Yang, Z., Fu, X., Tan, J., Yang, X.: An enhanced teaching-learning-based optimization algorithm with self-adaptive and learning operators and its search bias towards origin. Swarm Evol. Comput. **60**, 100766 (2021)

29. Chou, J.S., Nguyen, N.M.: FBI inspired meta-optimization. Appl. Soft Comput. **93**, 106339 (2020)

30. Kaveh, A., Hamedani, K.B., Kamalinejad, M.: An enhanced forensic-based investigation algorithm and its application to optimal design of frequency-constrained dome structures. Comput. Struct. **256**, 106643 (2021)

31. Chou, J.S., Truong, D.N.: Multiobjective forensic-based investigation algorithm for solving structural design problems. Autom. Constr. **134**, 104084 (2022)

32. Wu, G., Mallipeddi, R., Suganthan, P.N.: Problem definitions and evaluation criteria for the CEC 2017 competition on constrained real-parameter optimization. National University of Defense Technology, Changsha, Hunan, PR China and Kyungpook National University, Daegu, South Korea and Nanyang Technological University, Singapore, Technical Report (2017)

33. Das, S., Suganthan, P.N.: Problem definitions and evaluation criteria for CEC 2011 competition on testing evolutionary algorithms on real world optimization problems. Jadavpur University, Nanyang Technological University, Kolkata. p. 341-359 (2010)

34. Coello, C.A.C.: Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art. Comput. Methods Appl. Mech. Eng. **191**(11–12), 1245–1287 (2002)

35. Abualigah, L., Abd Elaziz, M., Sumari, P., Geem, Z.W., Gandomi, A.H.: Reptile search algorithm (RSA): a nature-inspired meta-heuristic optimizer. Expert Syst. Appl. **191**, 116158 (2022)

36. Ahmadianfar, I., Heidari, A.A., Noshadian, S., Chen, H., Gandomi, A.H.: INFO: an efficient optimization algorithm based on weighted mean of vectors. Expert Syst. Appl. **195**, 116516 (2022)

37. Yu, Y., Gao, S., Wang, Y., Todo, Y.: Global optimum-based search differential evolution. IEEE/CAA J. Autom. Sinica. **6**(2), 379–394 (2019)

38. Ma, L., Cheng, S., Shi, Y.: Enhancing learning efficiency of brain storm optimization via orthogonal learning design. IEEE Trans. Syst. Man Cybern.: Syst. **51**(11), 6723–6742 (2020)

39. Rengasamy, S., Murugesan, P.: PSO based data clustering with a different perception. Swarm Evol. Comput. **64**, 100895 (2021)

40. Yang, Y., Chen, H., Heidari, A.A., Gandomi, A.H.: Hunger games search: visions, conception, implementation, deep analysis, perspectives, and towards performance shifts. Expert Syst. Appl. **177**, 114864 (2021)

41. Li, S., Chen, H., Wang, M., Heidari, A.A., Mirjalili, S.: Slime mould algorithm: a new method for stochastic optimization. Futur. Gener. Comput. Syst. **111**, 300–323 (2020)

42. Dhiman, G.: SSC: a hybrid nature-inspired meta-heuristic optimization algorithm for engineering applications. Knowl.-Based Syst. **222**, 106926 (2021)

43. Jiang, Y., Wu, Q., Zhu, S., Zhang, L.: Orca predation algorithm: a novel bio-inspired algorithm for global optimization problems. Expert Syst. Appl. **188**, 116026 (2022)

44. Braik, M.S.: Chameleon swarm algorithm: a bio-inspired optimizer for solving engineering design problems. Expert Syst. Appl. **174**, 114685 (2021)

45. Karami, H., Anaraki, M.V., Farzin, S., Mirjalili, S.: Flow direction algorithm (FDA): a novel optimization approach for solving optimization problems. Comput. Ind. Eng. **156**, 107224 (2021)

46. Lei, Z., Gao, S., Wang, Y., Yu, Y., Guo, L.: An adaptive replacement strategy-incorporated particle swarm optimizer for wind farm layout optimization. Energy Convers. Manag. **269**, 116174 (2022)

47. Bai, F., Ju, X., Wang, S., Zhou, W., Liu, F.: Wind farm layout optimization using adaptive evolutionary algorithm with Monte Carlo tree search reinforcement learning. Energy Convers. Manag. **252**, 115047 (2022)

48. Chong, K.L., Lai, S.H., Ahmed, A.N., Jaafar, W.Z.W., El-Shafie, A.: Optimization of hydropower reservoir operation based on hedging policy using Jaya algorithm. Appl. Soft Comput. **106**, 107325 (2021)

49. Sharifi, M.R., Akbarifard, S., Qaderi, K., Madadi, M.R.: Comparative analysis of some evolutionary-based models in optimization of dam reservoirs operation. Sci. Rep. **11**(1), 1–17 (2021)

50. Fu, L., Ouyang, H., Zhang, C., Li, S., Mohamed, A.W.: A constrained cooperative adaptive multi-population differential evolutionary algorithm for economic load dispatch problems. Appl. Soft Comput. **121**, 108719 (2022)

51. Acharya, S., Ganesan, S., Kumar, D.V., Subramanian, S.: A multi-objective multi-verse optimization algorithm for dynamic load dispatch problems. Knowl.-Based Syst. **231**, 107411 (2021)

52. Momenitabar, M., Ebrahimi, Z.D., Ghasemi, P.: Designing a sustainable bioethanol supply chain network: A combination of machine learning and meta-heuristic algorithms. Ind. Crops Prod. **189**, 115848 (2022)

53. Momenitabar, M., Ebrahimi, Z.D., Abdollahi, A., Helmi, W., Bengtson, K., Ghasemi, P.: An integrated machine learning and quantitative optimization method for designing sustainable bioethanol supply chain networks. Decis. Anal. J. **7**, 100236 (2023)