



Article

A Multi-Objective Sine Cosine Algorithm Based on a Competitive Mechanism and Its Application in Engineering Design Problems

Nengxian Liu ¹, Jeng-Shyang Pan ², Genggeng Liu ¹, Mingjian Fu ¹, Yanyan Kong ^{3,*} and Pei Hu ⁴

¹ College of Computer and Data Science, Fuzhou University, Fuzhou 350108, China; lylnx@fzu.edu.cn (N.L.); liugenggeng@fzu.edu.cn (G.L.); sinceway@fzu.edu.cn (M.F.)

² School of Artificial Intelligence, Nanjing University of Information Science & Technology, Nanjing 210044, China; jsyan@cc.kuas.edu.tw

³ School of Information Science and Engineering, Zhejiang Sci-Tech University, Hangzhou 310018, China

⁴ School of Computer and Software, Nanyang Institute of Technology, Nanyang 473004, China

* Correspondence: yanyankong@zstu.edu.cn

Abstract: There are a lot of multi-objective optimization problems (MOPs) in the real world, and many multi-objective evolutionary algorithms (MOEAs) have been presented to solve MOPs. However, obtaining non-dominated solutions that trade off convergence and diversity remains a major challenge for a MOEA. To solve this problem, this paper designs an efficient multi-objective sine cosine algorithm based on a competitive mechanism (CMOSCA). In the CMOSCA, the ranking relies on non-dominated sorting, and the crowding distance rank is utilized to choose the outstanding agents, which are employed to guide the evolution of the SCA. Furthermore, a competitive mechanism stemming from the shift-based density estimation approach is adopted to devise a new position updating operator for creating offspring agents. In each competition, two agents are randomly selected from the outstanding agents, and the winner of the competition is integrated into the position update scheme of the SCA. The performance of our proposed CMOSCA was first verified on three benchmark suites (i.e., DTLZ, WFG, and ZDT) with diversity characteristics and compared with several MOEAs. The experimental results indicated that the CMOSCA can obtain a Pareto-optimal front with better convergence and diversity. Finally, the CMOSCA was applied to deal with several engineering design problems taken from the literature, and the statistical results demonstrated that the CMOSCA is an efficient and effective approach for engineering design problems.

Keywords: multi-objective algorithm; sine cosine algorithm (SCA); competitive mechanism; engineering design problem



Citation: Liu, N.; Pan, J.-S.; Liu, G.; Fu, M.; Kong, Y.; Hu, P. A Multi-Objective Sine Cosine Algorithm Based on a Competitive Mechanism and Its Application in Engineering Design Problems. *Biomimetics* **2024**, *9*, 115. <https://doi.org/10.3390/biomimetics9020115>

Academic Editors: Heming Jia, Laith Abualigah and Xuewen Xia

Received: 22 January 2024

Revised: 7 February 2024

Accepted: 9 February 2024

Published: 15 February 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Multi-objective optimization problems (MOPs) arise from engineering applications [1], community detection problems [2], charging station placement problems [3], the detection of ice accretion on aircraft [4], etc. MOPs often involve multiple conflicting objectives [5,6], such as environmental/economic dispatch (EED) problems with the two objectives of minimizing pollution emissions and generation costs, and there does not exist a solution that is able to optimize all objectives simultaneously. The mathematical model of the MOP for minimization is formulated as follows:

$$\begin{aligned} \min \quad & F(x) = (f_1(x), f_2(x), \dots, f_m(x)) \\ \text{s.t.} \quad & g_i(x) \leq 0 \quad (p = 1, 2, \dots, p) \\ & h_j(x) = 0 \quad (q = 1, 2, \dots, q) \end{aligned} \quad (1)$$

where $x = (x_1, x_2, \dots, x_D)$ denotes the D -dimensional candidate solution in decision space; $f_1(x), f_2(x), \dots, f_m(x)$ are objective functions; and m is the number of objectives.

$g_i(x)$ and $h_j(x)$ are constraint functions. Given two candidate solutions x_a and x_b in the feasible region, x_a dominates x_b , if and only if $\forall i, f_i(x_a) \leq f_i(x_b)$ and $\exists j, f_j(x_a) < f_j(x_b)$, $i, j \in \{1, 2, \dots, m\}$. If no other solution can dominate x^* , then x^* is called the Pareto-optimal solution. The set of all Pareto-optimal solutions is known as the Pareto-optimal set (PS), and the set of their corresponding objective values is called the Pareto front (PF).

In the past twenty years, multi-objective evolutionary algorithms (MOEAs) have achieved great popularity due to their excellent capabilities in dealing with MOPs [7,8]. In 1985, Schaffer introduced the first MOEA, i.e., the vector-evaluated genetic algorithm (VEGA) [9]. After the VEGA, a large number of MOEAs based on various metaheuristic algorithms were presented, such as the whale optimization algorithm [10], particle swarm optimization (PSO) [11], the carnivorous plant algorithm [12], the equilibrium optimizer slime mould algorithm [1], differential evolution (DE) [13,14], the butterfly optimization algorithm [15], the remora optimization algorithm [16,17], the crayfish optimization algorithm [18], and the gray wolf optimizer [19]. Based on their evolution schemes, MOEAs can be roughly partitioned into three types.

The first type is dominance-based MOEAs. These rank solutions according to the dominance relationship and choose solutions for the next generation based on Pareto and diversity selection criteria. The most representative and famous example for MOPs is the fast elite multi-objective GA (NSGA-II), which was introduced by Deb et al. [20]. To further address Many-objective optimization problems, scholars designed several new dominance rules, such as θ -dominance [21], grid dominance [22], fuzzy dominance [23], and ϵ -dominance [24]. Recently, inspired by the idea of a competitive swarm optimizer (CSO) [25], Zhang et al. developed a multi-objective PSO (CMOPSO) that utilizes a pairwise competitive mechanism to update the velocity of particles [26]. The comparison results demonstrated that the CMOPSO performed better than the competing algorithms. To enhance the performance of the CMOPSO, Han et al. adopted a tripartite competition mechanism to propose an improved multi-objective PSO, called the TC-MOPSO [27]. Using the level swarm optimizer and a competition mechanism, Zhang et al. introduced an MOEA named EMOSO [28].

The second type is decomposition-based MOEAs. This type can be further classified into two categories, one wherein the MOP is transformed into a series of single-objective sub-problems [29,30], and the other wherein a complex MOP is transformed into a series of simple MOPs [31]. A representative example of a decomposition-based MOEA is the MOEA/D [29]. The same authors introduced a new variant of the MOEA/D with DE operators, called MOEA/D-DE [32]. What is more, some advanced versions of the MOEA/D have been proposed to handle more challenging MOPs, such as MOEA/D-PaS [33] and MOEA/D-AM2M [34]. Using a decomposition strategy, Cheng et al. presented an MOEA with a set of reference vectors, named the RVEA [35]. Furthermore, Zhao et al. introduced a surrogate-ensemble assisted MOEA on the basis of the RVEA to deal with expensive problems [36]. Recently, Yang et al. introduced an MOEA with a dual decomposition strategy to solve large-scale multi-objective optimization problems [37].

The third type is indicator-based MOEAs, which utilize performance indicators rather than fitness for evolutionary selection. Beume et al. [38] introduced an s-metric selection-based MOEA, called the SMS-EMOA, via embedding the hypervolume (HV) with a non-dominated sorting design. Zitzler et al. developed an indicator-based evolutionary algorithm (IBEA), in which the HV and ϵ -indicator are employed in the evolutionary selection [39]. However, the computational time for the HV rapidly increases as the number of objectives increases. Hence, Bader et al. developed a hypervolume-based MOEA (Hype), which adopts Monte Carlo simulations to calculate the HV values [40]. To enhance the performance of MOEAs for irregular Pareto frontier problems, Tian et al. presented an indicator-based MOEA (AR-MOEA) whose reference points are adaptive [41].

Additionally, many researchers have combined the above three strategies and proposed some efficient MOEAs. Li et al. developed an MOEA/DD method using both decomposition and dominance mechanisms [42]. Based on the famous NSGA-II, Deb et al.

proposed an improved MOEA (NSGA-III) on the basis of a decomposition mechanism and non-dominated sorting strategy [43]. Wang et al. presented a MOEA (Two-arch2) by utilizing both the dominance and the performance indicator [44]. Although researchers have proposed many MOEAs to improve the convergence performance and maintain diversity and attained better performance, the balance of convergence and diversity is still a major challenge in multi-objective optimization.

In this paper, to promote the effectiveness of the MOEA in addressing MOPs, an efficient multi-objective sine cosine algorithm using an SDE-based competitive mechanism is proposed. The competitive idea utilized in the CSO is an effective mechanism that has also been employed to enhance the performance of other methods [10,26,45,46]. The sine cosine algorithm (SCA) is an effective meta-heuristic introduced in 2016 [47]. The SCA has demonstrated its robustness and effectiveness in terms of accuracy, convergence, and computational efforts [48]. However, the SCA has not yet been integrated with competitive mechanisms for handling MOPs. To take advantages of the SCA and the effectiveness of the competitive mechanism, we develop a multi-objective SCA based on the competitive mechanism, named the CMOSCA. The contributions of this work are summarized as follows.

- A new position updating operation based on a competitive mechanism with the shift-based density estimation (SDE) strategy is proposed. In this operation, an agent with better SDE fitness value is employed to guide the search of evolution. This operation can make use of the SDE-based competitive mechanism to attain a well balance between the diversity and convergence.
- We also present two variants of the CMOSCA, which utilize the Euclidean distance-based competitive mechanism and angle-based competitive mechanism, respectively. The performance of these two variants with CMOSCA was experimentally compared, and the experimental results indicate the virtue of the SDE-based competition mechanism.
- The performance of the CMOSCA is extensively analyzed via comparing CMOSCA with several representative MOEAs on twenty test functions having various landscapes of Pareto fronts. Furthermore, the proposed CMOSCA is also applied to address several engineering design problems. The comparison results evidence the competitive performance of our proposed CMOSCA.

The remaining article is organized as follows. Section 2 describes the idea of the SCA and related works on multi-objective SCA. Section 3 provides our proposed CMOSCA in detail. Experiments on twenty test functions and several engineering design problems with some discussions are given in Section 4. At last, the conclusions are summarized in Section 5.

2. Related Work

2.1. Sine Cosine Algorithm (SCA)

The SCA is a population-based meta-heuristic optimization approach, proposed by Seyedali Mirjalili in 2016, which is inspired by the mathematical characteristics of sine and cosine functions [47]. It employs the rules of trigonometric sine and cosine functions to update the positions of agents for the optimal solution. The positions of agents in the SCA are updated utilizing the Equation (2).

$$\begin{cases} x_i^{t+1} = x_i^t + r_1 \times \sin(r_2) \times |r_3 p_i^t - x_i^t| & r_4 < 0.5 \\ x_i^{t+1} = x_i^t + r_1 \times \cos(r_2) \times |r_3 p_i^t - x_i^t| & r_4 \geq 0.5 \end{cases} \quad (2)$$

where x_i^{t+1} , x_i^t are the i th ($i = 1, 2, \dots, N$) agents at $(t + 1)$ th and t th iteration respectively and p_i^t represents the best agent (destination point) at the t th iteration, and r_1, r_2, r_3 , and r_4 are random control parameters. These parameters are integrated to prevent the algorithm from getting stuck in local optima and to balance the exploration and exploitation of the search process.

The parameter r_1 makes a contribution to the exploration in the first half of the iterative search process and to the exploitation in the second half of the iterative search process. To balance exploration and exploitation, r_1 decreases adaptively from a preset constant a to 0 with the following linear equation.

$$r_1 = a - t \frac{a}{T} \quad (3)$$

where a is a constant, t and T represent the current and maximum iteration, respectively.

The r_2 is a direction parameter, which indicates the movement of the current agent either toward or outside of the destination. r_2 is in the range of $[0, 2\pi]$. The r_3 is a weight parameter, which is used to randomly emphasize ($r_3 > 1$) or weaken ($r_3 < 1$) the impact of the destination on other solution agents. The range of r_3 is $[0, 2]$. Finally, r_4 is a uniformly random parameter in the range of $[0, 1]$, which performs as a switch to equally select sine or cosine function in Equation (2). The influence of the sine and cosine functions in Equation (2) on the next position in the $[-2, 2]$ interval is illustrated in Figure 1.

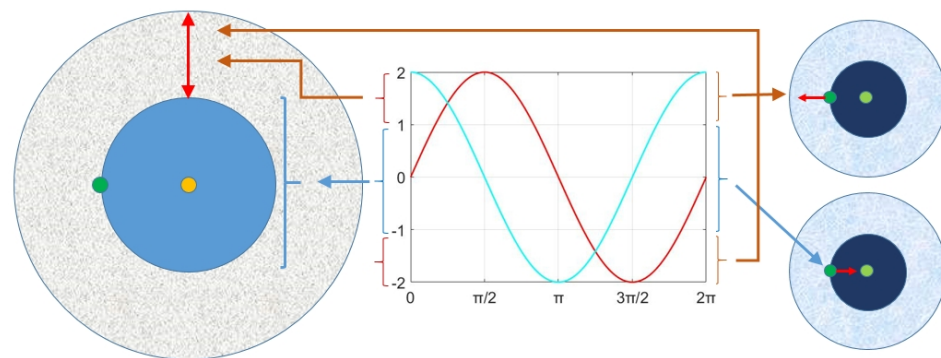


Figure 1. The influence of sine and cosine functions in Equation (2) on the next position in the $[-2, 2]$ interval.

Like other optimization approaches, The SCA begins with an initialization step, where a population of agents (initial solutions) is generated in a stochastic manner. These agents are updated iteratively via Equation (2) and the iterative process stops when the termination condition is met. The flowchart of the SCA is illustrated in Figure 2.

2.2. Existing Multi-Objective SCA Algorithms

In recent years, many ideas have been combined with the original SCA to propose multi-objective SCA (MOSCA) for solving MOPs. In this part, some recently proposed multi-objective SCA methods are introduced in brief.

Rizk-Allah et al. [49] introduced a multi-objective SCA, called the MSCO, to deal with the nonsmooth EELD problem. In the MSCO, random initialization and opposition strategy are used to maintain the diversity of agents. The Pareto front concepts are employed to find a group of non-dominated solutions. The MSCO method was verified on 6-unit and 10-unit test systems, and the results evidenced the effectiveness and robustness of the MSCO.

Tawhid and Savsani presented a multi-objective SCA (MO-SCA) to deal with various benchmark MOPs and some engineering design problems [50]. In the MO-SCA, the elitist non-dominated sorting strategy and the crowding distance method are employed to define the non-dominated ranks and enhanced coverage of the obtained Pareto optimal solutions. The attained results reveal that the MO-SCA can effectively create the Pareto fronts.

Wan et al. introduced a multi-objective SCA (MOSCA) for band selecting of hyperspectral image (HSI) [51]. The effectiveness of the MOSCA was assessed on two real-world HSI scenes and the obtained experimental results reveal the superior performance of the MOSCA compared to other competing methods. In 2022, Wan et al. further presented a multi-objective SCA for spatial-spectral clustering of remote sensing image data (MOSCA-

SSC), which treats the clustering task as a multi-objective optimization problem [52]. In the MOSCA-SCC, the destination agent is automatically chosen and renewed from the current Pareto front via utilizing the knee-point-based selection method.

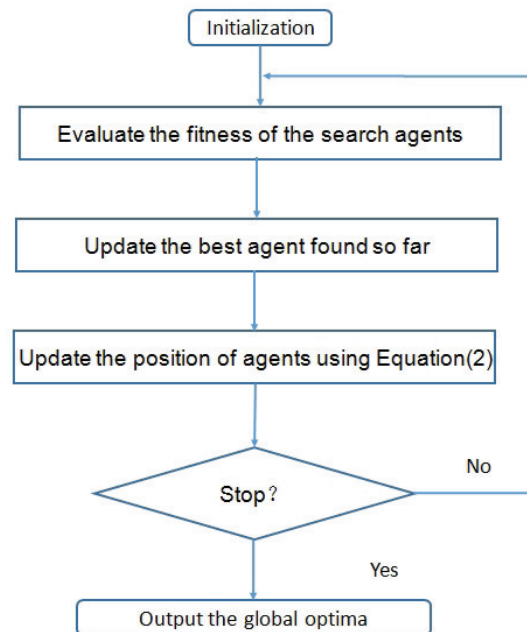


Figure 2. Flowchart of the Sine Cosine Algorithm.

Abdel-Basset et al. presented a multi-objective technique to handle the task scheduling in multiprocessor systems (MPS) with the modified SCA (MSCA) to optimize both the makespan and energy objectives [53]. This algorithm uses the Pareto dominance strategy and is called energy-aware multi-objective MSCA (EA-M2SCA). Furthermore, the EA-M2SCA was hybridized with the polynomial mutation mechanism to enhance its performance and promote the convergence behavior. This hybrid improved version is called the EA-MHSCA. Finally, the proposed EA-MHSCA is compared with many well-established MOEAs, and the EA-MHSCA shows superiority in most test cases.

Wang et al. [54] developed a multi-objective SCA (MOSCA) for wind speed forecasting, in which a hybrid wavelet neural network based on the MOSCA is proposed and experimental results show that the proposed algorithm achieved better accuracy and stability.

Selim et al. [55] introduced an efficient multi-objective SCA for the optimal allocation of distribution static compensators in distribution networks, which employs a grey relational analysis and a fuzzy loss sensitivity factor. The experimental results reveal the effectiveness and superiority of the proposed algorithm.

Altay and Alatas [56] proposed DE and SCA based novel hybrid multi-objective methods for numerical association rule mining. This study proposes three new hybrid approaches by integrating sine cosine operators into the DE method and using global exploration capability of the DE and local exploitation capability of the SCA.

Raut et al. [57] introduced Pareto multi-objective SCA for performance improvement of radial distribution network, where both the self-adapting levy mutation and exponential variation of the conversion parameter are used to improve its performance.

Narayanan et al. [58] proposed a new Many-objective SCA (MaOSCA), which uses reference points and information feedback mechanisms, and experimental results show that the MaOSCA can obtain effective and robust performance.

More related works on the MOSCAs can be found in [48,59–61]. Literature reviews indicate that no previous work has utilized the competitive mechanism to develop the SCA for handling multi-objective optimization problems.

In this paper, we proposed a multi-objective SCA algorithm based on a competition mechanism, which is different from previous multi-objective algorithms. For example, compared with paper [56], paper [56] proposed a hybrid multi-objective algorithm based on DE and SCA for numerical association rule mining. The proposed algorithm uses the SCA operator instead of the DE operator. Compared with paper [55], paper [55] proposed an effective multi-objective SCA by using a gray correlation analysis and fuzzy loss sensitivity factors. Among them, the gray correlation analysis is used to select leader solutions, while our proposed CMOSCA uses a competition mechanism to select leader individuals. Compared with the paper [13], which employed a competition mechanism to develop a multi-objective DE algorithm, our proposed CMOSCA utilizes a competition mechanism to design a new SCA algorithm. Compared with the paper [53], the paper [53] developed a multi-objective SCA algorithm to deal with task scheduling problems. The main innovations of this algorithm include individual initialization, individual encoding and mutation operations.

3. The Proposed CMOSCA

In this section, the details of our proposed CMOSCA are given. We first describe the complete framework of the CMOSCA, and then develop the position updated strategy for the SCA based on the competition mechanism with the SDE strategy. Third, two variants of our proposed CMOSCA are provided. Finally, we analyze the time complexity of the CMOSCA algorithm.

3.1. The Framework of the CMOSCA

Similar to the CMOPSO, our proposed CMOSCA has a simple and clear framework as given in Algorithm 1, where the main loop involves two modules, namely, the competition mechanism based position updating scheme for search agents and environment selection. To illustrate more clearly, the main framework of the CMOSCA is presented in Figure 3. It is worth mentioning that this framework follows the general framework of other evolutionary algorithms, such as the GA, which uses crossover and mutation operators to produce offspring, and selects individuals from both parent and descendant populations to form a new population.

Algorithm 1 The framework of CMOSCA

Input: Number of search agents N , Dimension of the problem D , Maximum number of function evaluations $MaxFES$, Current number of function evaluations FES .

Output: All non-dominant search agents in P .

- 1: Initialize the search agents of P
 - 2: **while** ($FES < MaxFES$) **do**
 - 3: $P' \leftarrow$ CompetitionBasedSCAPositionUpdate(P);
 - 4: $P \leftarrow$ EnvironmentalSelection(P, P');
 - 5: **end while**
 - 6: **Return** P
-

The process of implementing the CMOSCA is provided as follows. Firstly, the population of search agents is initialized in a random manner. Then the search agents in the population P are renewed by the proposed SCA position update scheme to generate descendants P' . At last, we adopt the environment selection to choose N agents from the parent population P and descendant population P' . Note that the CMOSCA utilizes the same environmental selection approach as the paper [62] in the iterative process. The above process (namely, SCA position updating and environmental selection) terminates when the stopping condition is met.

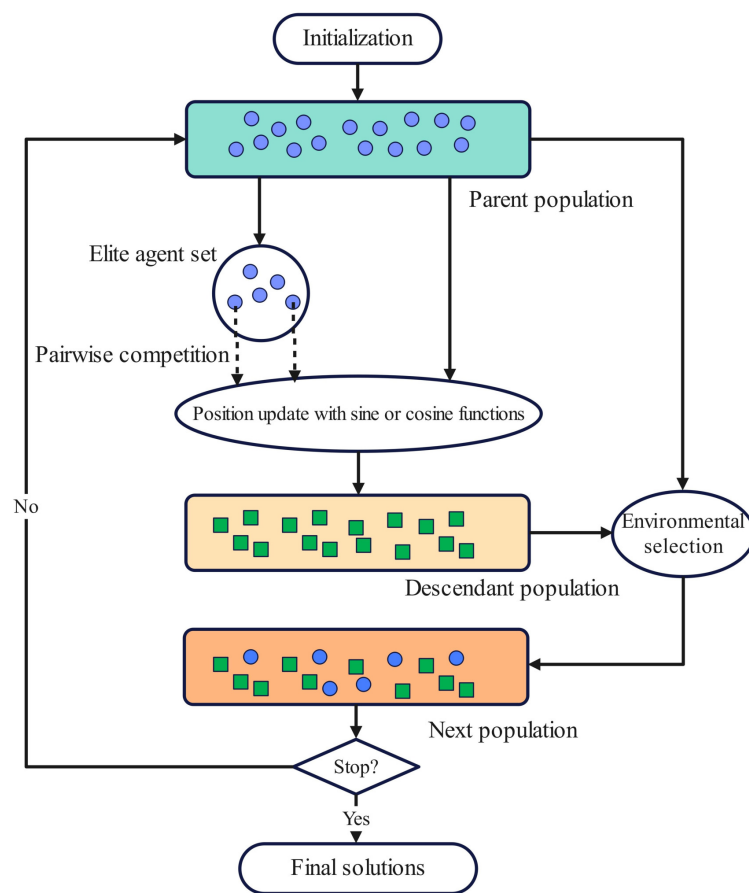


Figure 3. The framework of CMOSCA. 1. Elite agent set is used to guide the search of the population. 2. Pairwise competition is employed to select a winner to participate in the position update of the agent. 3. Environmental selection is adopted to choose N agents from the parent population and descendant population.

3.2. The SCA Position Update Scheme Based on a Competition Mechanism

There are four components in the SCA position update scheme with an SDE-based competition mechanism, i.e., creating the elite agent set E , fitness value calculation, pairwise competition, and the SCA position update. The pseudo-code of the SCA position update scheme with an SDE-based competition mechanism is presented in Algorithm 2.

First, the elite agent set is generated (line 2 of Algorithm 2). The elite agent set plays a major role in the proposed SCA position update scheme because it provides candidate agents employed in pairwise competition to lead the search of the population. Similar to the CMOPSO, the elite agent set utilizes the approach proposed in the NSGA-II [20] to select elite agents, i.e., a non-dominated sorting and crowding distance based ranking approach which can maintain better diversity and convergence. Specifically, the non-dominated sorting is first conducted on the population P to get the Pareto fronts F_1, F_2, \dots, F_p , where p represents the maximum index of fronts. Then, the minimum number q is found such that $|F_1 \cup F_2 \cup \dots \cup F_q| \geq \beta$, where β denotes the number of elite agents to be chosen. Finally, all agents belonging to first $q - 1$ fronts are chosen as the elite agents and the remaining agents are chosen from F_q based on the crowding distance of each agent in F_q . It is worth mentioning that the search agents in the elite agent set are chosen directly from the current population, so the CMOSCA does not require an external archive to record non-dominated solutions. Additionally, the elite agent size β has a crucial impact on the effectiveness of the CMOSCA, and we will provide a detailed discussion on β in the experimental section.

Algorithm 2 CompetitionBasedSCAPositionUpdate(P)**Input:** current population P , the size of elite agent set β .**Output:** descendant population P' .

```

1:  $P' \leftarrow \Phi$ ;
   /*Choosing the outstanding agents to form the elite agent set*/;
2:  $E \leftarrow$  Choose  $\beta$  agents from  $P$  relied on the front index and crowding distance.
3:  $Fitness \leftarrow$  Evaluate the fitness value of each agent in  $E$  by Equation (4);
4: for each agent  $x_i$  in  $P$  do
5:    $\{a, b\} \leftarrow$  Randomly choose two agents from  $E$ 
6:   if  $Fitness(a) < Fitness(b)$  then
7:      $x_l = a$ ; //loser in the pairwise competition
8:      $x_w = b$ ; //winner in the pairwise competition
9:   else
10:     $x_l = b$ ;
11:     $x_w = a$ ;
12:   end if
13:   Update  $r_1, r_2$ , and  $r_4$ , randomly;
14:   Generate new  $x'_i$  by Equation (5);
15:    $P' \leftarrow P' \cup \{x'_i\}$ ;
16: end for
17: Return  $P'$ 

```

Then, we utilize the shift-based density estimation (SDE) method [63] to evaluate the fitness value of each agent in the elite agent set E (line 3 of Algorithm 2). To be Specificial, the minimum SDE-based distance Equation (4) [46] is adopted to evaluate the fitness value of an agent x .

$$Fitness(x) = \min_{y \in P \setminus \{x\}} \sqrt{\sum_{i=1}^m (\max\{0, f_i(y) - f_i(x)\})^2} \quad (4)$$

where m represents the number of objectives and $f_i(x)$ indicates the i th objective value of x . The SDE method has been utilized in several MOEAs [27,28,46] as it can assess the quality of a candidate solution (search agent) taking into account both diversity and convergence. Thus, CMOSCA employs the SDE-based distance to measure both diversity and convergence of each search agent in the population.

Afterward, two agents a, b are randomly picked up from the elite agent set E and compared in pairs (lines 5–12 of Algorithm 2). The one having a higher fitness value is represented as the winner x_w , and the other is represented as the loser x_l . Then the winner is adopted to participate in the position update scheme of the agent (line 14 of Algorithm 2). The position update scheme is defined as follows.

$$\begin{cases} x'_i = x_i + r_1 \times \sin(r_2) \times |r_3 x_{w,i} - x_i| & r_4 < 0.5 \\ x'_i = x_i + r_1 \times \cos(r_2) \times |r_3 x_{w,i} - x_i| & r_4 \geq 0.5 \end{cases} \quad (5)$$

In the proposed position update scheme, the winner x_w is used to replace the best agent (destination point) in the original position update scheme Equation (2) as the best agent does not exist in the MOPs. On the other hand, the winner is close to the Pareto front and has low density since the winner in the elite agent set is chosen by the approach introduced in the NSGA-II. Therefore, the proposed position update scheme with the winner to guide the search of evolution has better convergence and diversity. Note that the r_3 in Equation (5) is set to 1 to maintain the relationship with the Pareto front.

Finally, the generated new position x'_i is put into the descendant population P' . After creating N new positions, the process will go back to Algorithm 1 to execute the MOEA environmental selection.

3.3. Two Variants of the Proposed CMOSCA

There are two other approaches that can be adopted to measure the quality of candidate agents (solutions) [26,35], i.e., the angle between two agents (solutions) and the Euclidean distance between the origin and the agent. According to these two approaches, we introduce two variants of the proposed CMOSCA, which are denoted as the CMOSCAA and the CMOSCAD, respectively. Algorithms 3 and 4 provide the position update pseudo codes of these two variants. Generally, the angle approach is utilized to evaluate the diversity of the agents, and the distance approach is employed to evaluate the convergence of the agents. However, the SDE approach can measure both diversity and convergence of the agents. Hence, the proposed CMOSCA outperforms the other two variants and the experimental results will confirm this conclusion.

Algorithm 3 CompetitionBasedSCAPositionUpdate_Angle(P)

Input: current population P , the size of elite agent set β .

Output: descendant population P' .

```

1:  $P' \leftarrow \Phi$ ;
   /*Choosing the outstanding agents to form the elite agent set*/;
2:  $E \leftarrow$  Select  $\beta$  agents from  $P$  relied on the front index and crowding distance.
3: for each agent  $x_i$  in  $P$  do
4:    $\{a, b\} \leftarrow$  Randomly choose two agents from  $E$ 
5:   compute the angle  $\phi_1$  between  $a$  and  $x_i$ ,  $\phi_2$  between  $b$  and  $x_i$ 
6:   if  $\phi_2 < \phi_1$  then
7:      $x_l = a$ ; //loser in the pairwise competition
8:      $x_w = b$ ; //winner in the pairwise competition
9:   else
10:     $x_l = b$ ;
11:     $x_w = a$ ;
12:   end if
13:   Update  $r_1, r_2$ , and  $r_4$ , randomly;
14:   Generate new  $x'_i$  by Equation (5);
15:    $P' \leftarrow P' \cup \{x'_i\}$ ;
16: end for
17: Return  $P'$ 

```

Algorithm 4 CompetitionBasedSCAPositionUpdate_Distance(P)

Input: current population P , the size of leader set β .

Output: descendant population P' .

```

1:  $P' \leftarrow \Phi$ ;
   /*Choosing the outstanding agents to form the elite agent set*/;
2:  $E \leftarrow$  Select  $\beta$  agents from  $P$  relied on the front index and crowding distance.
3:  $Distance \leftarrow$  Calculate the Euclidean distance for each agent in  $E$ ;
4: for each agent  $x_i$  in  $P$  do
5:    $\{a, b\} \leftarrow$  Randomly select two agents from  $E$ 
6:   if  $Distance(b) < Distance(a)$  then
7:      $x_l = a$ ; //loser in the pairwise competition
8:      $x_w = b$ ; //winner in the pairwise competition
9:   else
10:     $x_l = b$ ;
11:     $x_w = a$ ;
12:   end if
13:   Update  $r_1, r_2$ , and  $r_4$ , randomly;
14:   Generate new  $x'_i$  by Equation (5);
15:    $P' \leftarrow P' \cup \{x'_i\}$ ;
16: end for
17: Return  $P'$ 

```

3.4. Computational Complexity of the CMOSCA

The time complexity of the CMOSCA is mainly determined by the operations of agent updating strategy and environmental selection. The agent updating strategy mainly composed of creating elite agent set, fitness calculation, and the SCA position update. For a population size N and a m -objective problem, the elite agents are chosen by non-dominated sorting and crowding distance sorting. The computational complexity of nondominated sorting is $O(m(2N)^2)$ in the worst case, crowding distance assignment is $O(m(2N)\log(2N))$, and the time complexity of sort on crowded-comparison operator is $O(2N\log(2N))$, hence, the overall computational complexity of creating elite agent set is $O(m(2N)^2 + m(2N)\log(2N) + 2N\log(2N)) \sim O(mN^2)$ in the worst case. The time complexity of calculating the fitness of each agent in elite agent set is $O(\beta N^2)$, where β is the size of elite agent set. The time complexity of the SCA position update is $O(N)$. So the total computational complexity of the agent updating strategy is $O(mN^2 + \beta N^2 + N) \sim O(mN^2)$. The time complexity of environmental selection is $O(2mN\log(2N))$ in the worst case [62]. In summary, the overall computational complexity of one generation in the CMOSCA is $O(mN^2)$ in the worst case. In the experimental section, we will compare the average runtimes of CMOSCA with that of other six competing MOEAs for MOPs.

4. Experimental Studies

In this part, to investigate the effectiveness of our proposed algorithm CMOSCA, we compare it with six typical MOEAs, including EMOSO [28], CMOPSO [26], MOEA/D [29], NSGA-II [20], MOEA/D-DE [32], MMOPSO [64]. These competing methods are chosen because they have shown good performance on MOPs with various types of PF. Where the EMOSO and the CMOPSO are recently proposed methods that also use the competitive mechanism, the MOEA/D and the NSGA-II are two famous algorithms with the GA operators, the MOEA/D-DE is a variant of the MOEA/D with DE operators, and the MMOPSO is an improved version of multi-objective PSO using multiple search strategies. These six MOEAs are programmed with Matlab and embedded into the PlatEMO [65].

We choose 20 test problems, including DTLZ1-DTLZ7 [66], WFG1-WFG9 [67], and ZDT1-ZDT4 [68]. These test problems were designed taking into account diversity characteristics, covering a good representation of various real-world scenarios, such as being disconnected, convex, concave, degenerate, multimodal, and with an irregular Pareto front shape, which can evaluate the efficiency and reliability of the MOEAs. On the other hand, these test problems were widely adopted to validate the performance of the other MOEAs [26–28]. Detailed information about these test problems is listed in Table 1. Problems DTLZ and WFG are variable objective functions, and problems ZDT are bi-objective functions. The number of decision variables for problems DTLZ are set to $n = k + m - 1$, where m and n are the number of objectives and decision variables, respectively. According to the paper [69], k is set to 5 for DTLZ1, to 10 for DTLZ2-DTLZ6 and to 20 for DTLZ7. For problems WFG, the number of decision variables is set to $n = k + l$ as recommended in the paper [70], where k and l are set to $m - 1$ and 10. For problems ZDT, the number of decision variables n is set to 30 for ZDT1-ZDT3 and 10 for ZDT4 as suggested in the paper [26].

Inverted generational distance (IGD) [71] and hypervolume (HV) [72] are employed as performance metrics to compare the CMOSCA with other competing MOEAs. To calculate the IGD values, approximately 10,000 reference points are chosen on the real Pareto front of each test problem. Smaller IGD values are better. To calculate the values of the HV, reference points are set to $(1, 1, \dots, 1)$ and all objective values are normalized as suggested in the paper [65]. Higher HV values are better.

For the sake of fair comparison, all parameters of the competing methods are set to the values suggested by the original papers. The related parameters employed in the experiments for each method are listed in Table 2. The parameter β of the proposed CMOSCA is set to 5 and it will be analysed in the following section. The population size is set to $N = 100$ for all competing MOEAs. All competing MOEAs utilize maximum number of function evaluations (FES) as the termination criteria. The maximum FES is set to 30,000

for all two- and three-objective problems. All competing MOEAs are independently run 30 times on each test instance, and the mean and the standard deviation of the performance metrics values are reported. In addition, for a comprehensive evaluation, Wilcoxon Signed Rank Test at a significance level of 0.05 was further utilized to test the performance of all competing MOEAs, where the symbols “+”, “=” and “-” indicate that the results of competing MOEAs are statistically superior, similar, and inferior to results obtained by CMOSCA, respectively. All the experiments are performed on a computer with Intel Core i5 @ 3.3 GHz dual-core CPU and Windows 7 operating system with MATLAB 2020b.

Table 1. Characteristics of test problems.

Test Problems	Objective Numbers	Properties of PF
DTLZ1	2,3	Linear
DTLZ2	2,3	Concave, Uni-modal
DTLZ3	2,3	Multimodal
DTLZ4	2,3	Concave, Biased, Uni-modal
DTLZ5	2,3	Concave, Degenerated
DTLZ6	2,3	Concave, Degenerated, Biased
DTLZ7	2,3	Mixed, Disconnected, Scaled
WFG1	2,3	Sharp tails
WFG2	2,3	Disconnected
WFG3	2,3	Mostly degenerated
WFG4-9	2,3	Concave
ZDT1	2	Convex
ZDT2	2	Concave
ZDT3	2	Disconnected, Multimodal
ZDT4	2	Convex, Multimodal

Table 2. Parameters Settings of all competing MOEAs.

Method	Parameters Settings
EMOSO	$NL = 4, c_1, c_2 \in [0, 1], r_1, r_2, r_3, r_4, r_5, r_6 \in [0, 1], p_m = 1/D, \eta_m = 20, \gamma = 5$
CMOPSO	$\omega, r_1 \in [0, 1], p_m = 1/D, \eta_m = 20, \gamma = 10$
MOEA/D	$p_c = 1, p_m = 1/D, \eta_c = \eta_m = 20, T = 0.1 * N$
NSGA-II	$p_c = 1, p_m = 1/D, \eta_c = \eta_m = 20$
MOEA/D-DE	$CR = 1, F = 0.5, \delta = 0.9, nr = 2, \eta_m = 20, T = 0.1 * N$
MMOPSO	$\omega \in [0.1, 0.5], c_1, c_2 \in [1.5, 2.5], r_1, r_2 \in [0, 1],$ $p_c = 0.9, p_m = 1/D, \eta_c = \eta_m = 20$
CMOSCA	$r_1 \in [0, 2], r_2 \in [0, 2\pi], r_3 = 1, r_4 \in [0, 1], \beta = 5$

4.1. Comparisons CMOSCA with Other Competing MOEAs

Table 3 gives the IGD results of the seven competing MOEAs on DTLZ1-DTLZ7, WFG1-WFG9, and ZDT1-ZDT4 test problems. It is obvious that the proposed CMOSCA outperforms the other six competing MOEAs. Specifically, the proposed CMOSCA obtains the best mean IGD results on 15 out of the 36 test problems, the EMOSO on 2 test problems, the CMOPSO on 5 test problems, the MOEA/D on 4 test problems, the NSGA-II on 6 test problems, the MOEA/D-DE on 2 test problems, and the NSGA-II on 2 test problems. The proposed CMOSCA attains better (similar) performances in comparison with the EMOSO on 25 (5) out of the 36 problems. Both CMOSCA and EMOSO adopt the outstanding solutions to guide the search, but they utilize different evolutionary operators. The proposed CMOSCA has better (similar) performances in comparison with the CMOPSO on 19 (6) out of the 36 problems. These two MOEAs have equivalent frameworks, but they employ different competition mechanisms and evolutionary operators, so they perform differently. The proposed CMOSCA has better performances in comparison with the MOEA/D, NSGA-II, MOEA/D-DE, and MMOPSO on 19, 20, 20, and 20 out of the 36 problems, respectively. The CMOSCA obtains better performance mainly owing to the fact that it employs the SDE-based competition mechanism and the SCA evolution operators.

HV results of the seven competing MOEAs are recorded in Table 4. From Table 4, we can clearly observe that our proposed CMOSCA obtains better HV results than the other six competing MOEAs on most of test problems. CMOSCA obtains the best mean HV results

on 13 out of 36 problems. The EMOSO, CMOPSO, MOEA/D, NSGA-II, MOEA/D-DE, and MMOPSO obtain the best mean HV results on 5, 1, 5, 7, 2, and 3 out of 36 problems, respectively. It can also be observed from Table 4 that several HV results are zero, which represents that the corresponding MOEA cannot obtain any candidate solution to dominate the reference point on corresponding test problems. For example, our proposed CMOSCA obtains zero HV results on two- and three-objective DTLZ3. This means that the CMOSCA is unable to effectively solve the highly multi-model DTLZ3. EMOSO gets zero HV results on two- and three-objective DTLZ1, two- and three-objective DTLZ3. CMOPSO obtains zero HV results on three-objective DTLZ1, two- and three-objective DTLZ3. The HV result of MMOPSO is zero on two-objective DTLZ3. These experimental results also reflect the limitations of the EMOSO, CMOPSO and MMOPSO.

Table 3. IGD values achieved by CMOSCA and six competing MOEAs.

Problem	M	D	EMOSO	CMOPSO	MOEAD	CMOSCA
DTLZ1	2	6	1.7772 × 10 ¹ (2.97 × 10 ⁰) =	8.0014 × 10 ⁻¹ (1.15 × 10 ⁰) +	2.6325 × 10 ⁻³ (7.36 × 10 ⁻⁴) +	1.8233 × 10 ¹ (6.02 × 10 ⁰)
	3	7	1.6639 × 10 ¹ (2.89 × 10 ⁰) -	8.2647 × 10 ⁰ (4.52 × 10 ⁰) +	2.1353 × 10 ⁻² (1.59 × 10 ⁻³) +	1.1081 × 10 ¹ (4.81 × 10 ⁰)
DTLZ2	2	11	4.3749 × 10 ⁻³ (6.68 × 10 ⁻⁵) -	4.3891 × 10 ⁻³ (6.81 × 10 ⁻⁵) -	3.9697 × 10 ⁻³ (1.32 × 10 ⁻⁵) +	4.1149 × 10 ⁻³ (3.78 × 10 ⁻⁵)
	3	12	5.8551 × 10 ⁻² (9.79 × 10 ⁻⁴) -	5.7573 × 10 ⁻² (9.46 × 10 ⁻⁴) -	5.4467 × 10 ⁻² (1.71 × 10 ⁻⁶) -	5.3243 × 10 ⁻² (4.56 × 10 ⁻⁴)
DTLZ3	2	11	1.8287 × 10 ² (1.26 × 10 ¹) -	3.2331 × 10 ¹ (1.91 × 10 ¹) +	9.9329 × 10 ⁻² (2.00 × 10 ⁻¹) +	1.5902 × 10 ² (1.94 × 10 ¹)
	3	12	1.8301 × 10 ² (1.44 × 10 ¹) -	8.9128 × 10 ¹ (4.24 × 10 ¹) +	7.5061 × 10 ⁻¹ (8.10 × 10 ⁻¹) +	1.6203 × 10 ² (1.60 × 10 ¹)
DTLZ4	2	11	1.7657 × 10 ⁻¹ (3.17 × 10 ⁻¹) -	1.7657 × 10 ⁻¹ (3.17 × 10 ⁻¹) -	2.5001 × 10 ⁻¹ (3.54 × 10 ⁻¹) -	4.1119 × 10 ⁻³ (3.26 × 10 ⁻⁵)
	3	12	9.8710 × 10 ⁻² (9.90 × 10 ⁻³) -	1.2015 × 10 ⁻¹ (2.24 × 10 ⁻¹) -	3.9264 × 10 ⁻¹ (3.35 × 10 ⁻¹) =	5.9577 × 10 ⁻² (1.47 × 10 ⁻³)
DTLZ5	2	11	4.4092 × 10 ⁻³ (6.13 × 10 ⁻⁵) -	4.3778 × 10 ⁻³ (5.31 × 10 ⁻⁵) -	3.9714 × 10 ⁻³ (1.59 × 10 ⁻⁵) +	4.1107 × 10 ⁻³ (2.65 × 10 ⁻⁵)
	3	12	7.6799 × 10 ⁻³ (9.41 × 10 ⁻⁴) -	6.5749 × 10 ⁻³ (6.23 × 10 ⁻⁴) -	3.3783 × 10 ⁻² (5.81 × 10 ⁻⁵) -	4.3665 × 10 ⁻³ (1.69 × 10 ⁻⁴)
DTLZ6	2	11	4.1288 × 10 ⁻³ (3.15 × 10 ⁻⁵) =	4.1243 × 10 ⁻³ (3.07 × 10 ⁻⁵) =	3.9659 × 10 ⁻³ (7.10 × 10 ⁻⁷) +	4.1140 × 10 ⁻³ (3.03 × 10 ⁻⁵)
	3	12	4.1934 × 10 ⁻³ (4.70 × 10 ⁻⁵) =	4.2017 × 10 ⁻³ (5.73 × 10 ⁻⁵) =	3.3854 × 10 ⁻² (9.78 × 10 ⁻⁵) -	4.1788 × 10 ⁻³ (5.83 × 10 ⁻⁵)
DTLZ7	2	21	3.3686 × 10 ⁻² (1.11 × 10 ⁻¹) -	3.3711 × 10 ⁻² (1.11 × 10 ⁻¹) =	1.2727 × 10 ⁻¹ (1.96 × 10 ⁻¹) -	4.5057 × 10 ⁻³ (5.13 × 10 ⁻⁵)
	3	22	6.4654 × 10 ⁻² (1.67 × 10 ⁻³) -	1.3352 × 10 ⁻¹ (1.95 × 10 ⁻¹) -	1.9754 × 10 ⁻¹ (1.64 × 10 ⁻¹) -	5.8941 × 10 ⁻² (6.68 × 10 ⁻⁴)
WFG1	2	11	1.4596 × 10 ⁰ (3.48 × 10 ⁻²) -	6.6972 × 10 ⁻¹ (1.00 × 10 ⁻¹) -	1.6275 × 10 ⁻¹ (1.19 × 10 ⁻¹) -	3.8033 × 10 ⁻¹ (1.19 × 10 ⁻¹)
	3	12	1.8478 × 10 ⁰ (4.27 × 10 ⁻²) -	1.4486 × 10 ⁰ (5.24 × 10 ⁻²) -	3.2387 × 10 ⁻¹ (3.54 × 10 ⁻²) +	8.5394 × 10 ⁻¹ (1.43 × 10 ⁻¹)
WFG2	2	11	1.2332 × 10 ⁻² (4.29 × 10 ⁻⁴) =	1.1911 × 10 ⁻² (3.63 × 10 ⁻⁴) =	6.9447 × 10 ⁻² (4.10 × 10 ⁻²) -	3.5635 × 10 ⁻² (5.38 × 10 ⁻²)
	3	12	1.8572 × 10 ⁻¹ (6.77 × 10 ⁻³) +	1.7966 × 10 ⁻¹ (4.66 × 10 ⁻³) =	2.5232 × 10 ⁻¹ (1.07 × 10 ⁻²) -	1.9064 × 10 ⁻¹ (3.08 × 10 ⁻²)
WFG3	2	11	1.3828 × 10 ⁻² (3.72 × 10 ⁻⁴) +	1.3974 × 10 ⁻² (3.74 × 10 ⁻⁴) +	2.2398 × 10 ⁻² (3.39 × 10 ⁻³) +	2.9280 × 10 ⁻² (5.16 × 10 ⁻²)
	3	12	2.0509 × 10 ⁻¹ (1.10 × 10 ⁻²) -	1.5467 × 10 ⁻¹ (1.40 × 10 ⁻²) -	1.7054 × 10 ⁻¹ (1.83 × 10 ⁻²) -	1.3883 × 10 ⁻¹ (6.04 × 10 ⁻²)
WFG4	2	11	6.9249 × 10 ⁻² (4.63 × 10 ⁻³) -	4.5178 × 10 ⁻² (1.29 × 10 ⁻²) -	3.1383 × 10 ⁻² (4.21 × 10 ⁻³) -	2.0532 × 10 ⁻² (4.51 × 10 ⁻³)
	3	12	2.6565 × 10 ⁻¹ (5.11 × 10 ⁻³) -	2.6055 × 10 ⁻¹ (5.32 × 10 ⁻³) -	2.6299 × 10 ⁻¹ (6.02 × 10 ⁻³) -	2.4095 × 10 ⁻¹ (8.39 × 10 ⁻³)
WFG5	2	11	6.6847 × 10 ⁻² (1.72 × 10 ⁻³) -	6.7869 × 10 ⁻² (2.04 × 10 ⁻³) -	7.1356 × 10 ⁻² (1.40 × 10 ⁻³) -	6.3608 × 10 ⁻² (1.83 × 10 ⁻⁴)
	3	12	2.4004 × 10 ⁻¹ (4.84 × 10 ⁻³) -	2.4796 × 10 ⁻¹ (6.45 × 10 ⁻³) -	2.5268 × 10 ⁻¹ (3.45 × 10 ⁻³) -	2.2262 × 10 ⁻¹ (3.68 × 10 ⁻³)
WFG6	2	11	1.9095 × 10 ⁻² (8.98 × 10 ⁻³) +	1.8875 × 10 ⁻² (5.49 × 10 ⁻³) +	9.9941 × 10 ⁻² (2.20 × 10 ⁻²) +	2.2547 × 10 ⁻¹ (1.68 × 10 ⁻⁵)
	3	12	2.3183 × 10 ⁻¹ (5.23 × 10 ⁻³) +	2.4133 × 10 ⁻¹ (7.05 × 10 ⁻³) +	2.9133 × 10 ⁻¹ (1.67 × 10 ⁻²) +	3.3555 × 10 ⁻¹ (1.77 × 10 ⁻³)
WFG7	2	11	1.4701 × 10 ⁻² (3.70 × 10 ⁻⁴) -	1.4495 × 10 ⁻² (3.87 × 10 ⁻⁴) -	2.9914 × 10 ⁻² (3.85 × 10 ⁻³) -	1.2639 × 10 ⁻² (1.05 × 10 ⁻⁴)
	3	12	2.2766 × 10 ⁻¹ (3.31 × 10 ⁻³) -	2.3370 × 10 ⁻¹ (4.57 × 10 ⁻³) -	3.4056 × 10 ⁻¹ (4.60 × 10 ⁻²) -	2.1066 × 10 ⁻¹ (1.88 × 10 ⁻³)
WFG8	2	11	1.2258 × 10 ⁻¹ (3.57 × 10 ⁻³) -	1.1731 × 10 ⁻¹ (3.10 × 10 ⁻³) =	1.2890 × 10 ⁻¹ (1.08 × 10 ⁻²) -	1.1660 × 10 ⁻¹ (3.17 × 10 ⁻³)
	3	12	3.4539 × 10 ⁻¹ (7.05 × 10 ⁻³) -	3.3864 × 10 ⁻¹ (7.18 × 10 ⁻³) -	3.2357 × 10 ⁻¹ (1.07 × 10 ⁻²) -	3.1606 × 10 ⁻¹ (9.77 × 10 ⁻³)
WFG9	2	11	2.8963 × 10 ⁻² (2.07 × 10 ⁻³) +	2.6831 × 10 ⁻² (1.65 × 10 ⁻³) +	7.5073 × 10 ⁻² (6.03 × 10 ⁻²) +	2.2676 × 10 ⁻¹ (2.47 × 10 ⁻³)
	3	12	2.3940 × 10 ⁻¹ (4.84 × 10 ⁻²) +	2.2124 × 10 ⁻¹ (4.13 × 10 ⁻³) +	2.9889 × 10 ⁻¹ (3.10 × 10 ⁻²) +	3.4464 × 10 ⁻¹ (3.88 × 10 ⁻³)
ZDT1	2	30	4.0907 × 10 ⁻³ (6.75 × 10 ⁻⁵) -	4.1992 × 10 ⁻³ (1.03 × 10 ⁻⁴) -	1.3219 × 10 ⁻² (1.17 × 10 ⁻²) -	3.8994 × 10 ⁻³ (4.98 × 10 ⁻⁵)
ZDT2	2	30	4.0207 × 10 ⁻³ (6.83 × 10 ⁻⁵) -	4.1095 × 10 ⁻³ (7.72 × 10 ⁻⁵) -	2.9084 × 10 ⁻² (3.89 × 10 ⁻²) -	3.8933 × 10 ⁻³ (3.48 × 10 ⁻⁵)
ZDT3	2	30	4.6658 × 10 ⁻³ (8.52 × 10 ⁻⁵) =	4.6397 × 10 ⁻³ (6.51 × 10 ⁻⁵) +	3.1477 × 10 ⁻² (2.11 × 10 ⁻²) -	4.6982 × 10 ⁻³ (7.19 × 10 ⁻⁵)
ZDT4	2	10	1.7673 × 10 ¹ (4.62 × 10 ⁰) -	3.0164 × 10 ⁻¹ (3.89 × 10 ⁻¹) +	2.3644 × 10 ⁻² (1.66 × 10 ⁻²) +	1.7501 × 10 ¹ (1.63 × 10 ¹)
+/-/=			6/25/5	11/19/6	15/20/1	
Problem	M	D	NSGAII	MOEADDE	MMOPSO	CMOSCA
DTLZ1	2	6	2.4598 × 10 ⁻³ (3.99 × 10 ⁻⁴) +	2.3927 × 10 ⁻³ (1.30 × 10 ⁻³) +	9.1228 × 10 ⁻³ (3.15 × 10 ⁻²) +	1.8233 × 10 ¹ (6.02 × 10 ⁰)
	3	7	3.7968 × 10 ⁻² (4.95 × 10 ⁻²) +	7.2322 × 10 ⁻² (9.75 × 10 ⁻²) +	3.5135 × 10 ⁻¹ (2.76 × 10 ⁻¹) +	1.1081 × 10 ¹ (4.81 × 10 ⁰)
DTLZ2	2	11	5.0930 × 10 ⁻³ (1.71 × 10 ⁻⁴) -	3.9778 × 10 ⁻³ (5.04 × 10 ⁻⁶) +	5.3158 × 10 ⁻³ (5.06 × 10 ⁻⁴) -	4.1149 × 10 ⁻³ (3.78 × 10 ⁻⁵)
	3	12	7.2640 × 10 ⁻² (2.81 × 10 ⁻³) -	7.6176 × 10 ⁻² (1.11 × 10 ⁻³) -	7.1806 × 10 ⁻² (3.05 × 10 ⁻³) -	5.3243 × 10 ⁻² (4.56 × 10 ⁻⁴)
DTLZ3	2	11	6.4469 × 10 ⁻² (1.99 × 10 ⁻¹) +	1.0659 × 10 ¹ (1.22 × 10 ¹) +	9.3097 × 10 ⁰ (7.97 × 10 ⁰) +	1.5902 × 10 ² (1.94 × 10 ¹)
	3	12	2.3057 × 10 ⁻¹ (3.63 × 10 ⁻¹) +	4.5983 × 10 ⁰ (7.79 × 10 ⁰) +	1.3998 × 10 ¹ (9.70 × 10 ⁰) +	1.6203 × 10 ² (1.60 × 10 ¹)
DTLZ4	2	11	1.0339 × 10 ⁻¹ (2.55 × 10 ⁻¹) -	4.1241 × 10 ⁻³ (9.48 × 10 ⁻⁵) =	1.2797 × 10 ⁻¹ (2.79 × 10 ⁻¹) -	4.1119 × 10 ⁻³ (3.26 × 10 ⁻⁵)
	3	12	9.9949 × 10 ⁻² (1.60 × 10 ⁻¹) -	1.2732 × 10 ⁻¹ (6.97 × 10 ⁻²) -	7.0944 × 10 ⁻² (2.98 × 10 ⁻³) -	5.9577 × 10 ⁻² (1.47 × 10 ⁻³)
DTLZ5	2	11	5.0880 × 10 ⁻³ (2.22 × 10 ⁻⁴) -	3.9799 × 10 ⁻³ (9.64 × 10 ⁻⁶) +	5.2739 × 10 ⁻³ (2.28 × 10 ⁻⁴) -	4.1107 × 10 ⁻³ (2.65 × 10 ⁻⁵)
	3	12	6.3135 × 10 ⁻³ (3.30 × 10 ⁻⁴) -	1.4368 × 10 ⁻² (9.98 × 10 ⁻⁵) -	6.2533 × 10 ⁻³ (4.41 × 10 ⁻⁴) -	4.3665 × 10 ⁻³ (1.69 × 10 ⁻⁴)
DTLZ6	2	11	5.6922 × 10 ⁻³ (3.31 × 10 ⁻⁴) -	3.9664 × 10 ⁻³ (7.16 × 10 ⁻⁸) +	5.6898 × 10 ⁻³ (4.63 × 10 ⁻⁴) -	4.1140 × 10 ⁻³ (3.03 × 10 ⁻⁵)
	3	12	6.4813 × 10 ⁻³ (3.20 × 10 ⁻⁴) -	1.4503 × 10 ⁻² (5.27 × 10 ⁻⁵) -	6.8147 × 10 ⁻³ (8.01 × 10 ⁻⁴) -	4.1788 × 10 ⁻³ (5.83 × 10 ⁻⁵)
DTLZ7	2	21	5.2692 × 10 ⁻³ (1.42 × 10 ⁻⁴) -	9.4450 × 10 ⁻² (1.78 × 10 ⁻¹) -	1.8044 × 10 ⁻¹ (1.18 × 10 ⁻¹) -	4.5057 × 10 ⁻³ (5.13 × 10 ⁻⁵)
	3	22	7.8673 × 10 ⁻² (3.90 × 10 ⁻³) -	2.5102 × 10 ⁻¹ (1.37 × 10 ⁻¹) -	1.5383 × 10 ⁻¹ (1.57 × 10 ⁻¹) -	5.8941 × 10 ⁻² (6.68 × 10 ⁻⁴)
WFG1	2	11	8.3736 × 10 ⁻² (3.38 × 10 ⁻²) +	3.9448 × 10 ⁻¹ (9.68 × 10 ⁻²) =	1.1203 × 10 ⁻¹ (5.09 × 10 ⁻²) +	3.8033 × 10 ⁻¹ (1.19 × 10 ⁻¹)
	3	12	2.3801 × 10 ⁻¹ (1.60 × 10 ⁻²) +	1.1987 × 10 ⁰ (1.32 × 10 ⁻¹) -	3.8707 × 10 ⁻¹ (4.84 × 10 ⁻²) +	8.5394 × 10 ⁻¹ (1.43 × 10 ⁻¹)
WFG2	2	11	1.2893 × 10 ⁻² (5.50 × 10 ⁻⁴) +	2.2462 × 10 ⁻² (7.29 × 10 ⁻⁴) +	1.2785 × 10 ⁻² (5.11 × 10 ⁻⁴) +	3.5635 × 10 ⁻² (5.38 × 10 ⁻²)
	3	12	2.3473 × 10 ⁻¹ (1.62 × 10 ⁻²) -	3.4041 × 10 ⁻¹ (1.91 × 10 ⁻²) -	2.2874 × 10 ⁻¹ (1.07 × 10 ⁻²) -	1.9064 × 10 ⁻¹ (3.08 × 10 ⁻²)

Table 3. Cont.

Problem	M	D	EMOSO	CMOPSO	MOEAD	CMOSCA
WFG3	2	11	1.5434 × 10 ⁻² (7.48 × 10 ⁻⁴) +	1.5965 × 10 ⁻² (7.23 × 10 ⁻⁴) +	1.4776 × 10 ⁻² (6.86 × 10 ⁻⁴) +	2.9280 × 10 ⁻² (5.16 × 10 ⁻²)
	3	12	1.0219 × 10 ⁻¹ (1.42 × 10 ⁻²) +	1.7134 × 10 ⁻¹ (2.62 × 10 ⁻²) -	9.9848 × 10⁻² (2.37 × 10⁻²) +	1.3883 × 10 ⁻¹ (6.04 × 10 ⁻²)
WFG4	2	11	1.5646 × 10⁻² (6.48 × 10⁻⁴) +	5.1517 × 10 ⁻² (8.41 × 10 ⁻³) -	1.7160 × 10 ⁻² (1.10 × 10 ⁻³) +	2.0532 × 10 ⁻² (4.51 × 10 ⁻³)
	3	12	2.8260 × 10 ⁻¹ (8.38 × 10 ⁻³) -	3.8857 × 10 ⁻¹ (8.12 × 10 ⁻³) -	3.0067 × 10 ⁻¹ (1.15 × 10 ⁻²) -	2.4095 × 10⁻¹ (8.39 × 10⁻³)
WFG5	2	11	6.5711 × 10 ⁻² (1.52 × 10 ⁻³) -	6.8569 × 10 ⁻² (1.90 × 10 ⁻³) -	6.6765 × 10 ⁻² (2.47 × 10 ⁻³) -	6.3608 × 10⁻² (1.83 × 10⁻⁴)
	3	12	2.8644 × 10 ⁻¹ (9.22 × 10 ⁻³) -	3.3799 × 10 ⁻¹ (5.88 × 10 ⁻³) -	2.8841 × 10 ⁻¹ (1.13 × 10 ⁻²) -	2.2262 × 10⁻¹ (3.68 × 10⁻³)
WFG6	2	11	7.7548 × 10 ⁻² (1.95 × 10 ⁻²) +	6.4813 × 10 ⁻² (7.48 × 10 ⁻²) +	6.0904 × 10 ⁻² (6.81 × 10 ⁻²) +	2.2547 × 10 ⁻¹ (1.68 × 10 ⁻⁵)
	3	12	3.1640 × 10 ⁻¹ (1.97 × 10 ⁻²) +	3.9947 × 10 ⁻¹ (3.51 × 10 ⁻²) -	3.3096 × 10 ⁻¹ (5.51 × 10 ⁻²) +	3.3555 × 10 ⁻¹ (1.77 × 10 ⁻³)
WFG7	2	11	1.7271 × 10 ⁻² (7.33 × 10 ⁻⁴) -	1.4203 × 10 ⁻² (3.44 × 10 ⁻⁴) -	1.6735 × 10 ⁻² (1.51 × 10 ⁻³) -	1.2639 × 10⁻² (1.05 × 10⁻⁴)
	3	12	2.9255 × 10 ⁻¹ (1.31 × 10 ⁻²) -	3.5950 × 10 ⁻¹ (5.15 × 10 ⁻³) -	2.8503 × 10 ⁻¹ (1.25 × 10 ⁻²) -	2.1066 × 10⁻¹ (3.87 × 10⁻³)
WFG8	2	11	1.1129 × 10 ⁻¹ (1.44 × 10 ⁻³) +	1.0641 × 10⁻¹ (4.83 × 10⁻³) +	1.1043 × 10 ⁻¹ (2.34 × 10 ⁻³) +	1.1660 × 10 ⁻¹ (1.17 × 10 ⁻³)
	3	12	3.7766 × 10 ⁻¹ (1.28 × 10 ⁻²) -	4.2707 × 10 ⁻¹ (1.17 × 10 ⁻²) -	3.6982 × 10 ⁻¹ (1.18 × 10 ⁻²) -	3.1606 × 10⁻¹ (9.77 × 10⁻³)
WFG9	2	11	2.7899 × 10 ⁻² (3.79 × 10 ⁻²) +	2.8731 × 10 ⁻² (2.89 × 10 ⁻³) +	2.3062 × 10⁻² (2.93 × 10⁻³) +	2.2676 × 10 ⁻¹ (2.47 × 10 ⁻³)
	3	12	3.0118 × 10 ⁻¹ (3.33 × 10 ⁻²) +	3.3531 × 10 ⁻¹ (4.42 × 10 ⁻³) +	2.8868 × 10 ⁻¹ (2.06 × 10 ⁻²) +	3.4464 × 10 ⁻¹ (3.88 × 10 ⁻³)
ZDT1	2	30	4.7757 × 10 ⁻³ (1.65 × 10 ⁻⁴) -	1.2678 × 10 ⁻² (3.54 × 10 ⁻³) -	4.8826 × 10 ⁻³ (2.46 × 10 ⁻⁴) -	3.8994 × 10⁻³ (4.98 × 10⁻⁵)
ZDT2	2	30	4.8999 × 10 ⁻³ (1.74 × 10 ⁻⁴) -	8.5343 × 10 ⁻³ (1.84 × 10 ⁻³) -	5.1728 × 10 ⁻³ (2.42 × 10 ⁻⁴) -	3.8933 × 10⁻³ (3.48 × 10⁻⁵)
ZDT3	2	30	6.4385 × 10 ⁻³ (5.32 × 10 ⁻³) -	1.7481 × 10 ⁻² (6.53 × 10 ⁻³) -	5.5287 × 10 ⁻³ (2.89 × 10 ⁻⁴) -	4.6982 × 10 ⁻³ (7.19 × 10 ⁻⁵)
ZDT4	2	10	5.3574 × 10⁻³ (7.20 × 10⁻⁴) +	1.9761 × 10 ⁻¹ (1.30 × 10 ⁻¹) +	2.1125 × 10 ⁻² (4.84 × 10 ⁻²) +	1.7501 × 10 ¹ (1.63 × 10 ¹)
+/-/=			16/20/0	14/20/2	16/20/0	

“+”, “=” and “-” indicate that the results of competing MOEAs are statistically superior, similar, and inferior to results obtained by the CMOSCA, respectively. The best result of each test problem is displayed in bold.

Table 4. HV values achieved by CMOSCA and six competing MOEAs.

Problem	M	D	EMOSO	CMOPSO	MOEAD	CMOSCA
DTLZ1	2	6	0.0000 × 10 ⁰ (0.00 × 10 ⁰) =	1.5837 × 10 ⁻¹ (2.39 × 10 ⁻¹) +	5.7877 × 10 ⁻¹ (2.40 × 10 ⁻³) +	0.0000 × 10 ⁰ (0.00 × 10 ⁰)
	3	7	0.0000 × 10 ⁰ (0.00 × 10 ⁰) =	0.0000 × 10 ⁰ (0.00 × 10 ⁰) =	8.3698 × 10⁻¹ (5.86 × 10⁻³) +	0.0000 × 10 ⁰ (0.00 × 10 ⁰)
DTLZ2	2	11	3.4650 × 10 ⁻¹ (1.66 × 10 ⁻⁴) -	3.4644 × 10 ⁻¹ (1.89 × 10 ⁻⁴) -	3.4720 × 10 ⁻¹ (5.03 × 10 ⁻⁶) -	3.4743 × 10⁻¹ (5.68 × 10⁻⁵)
	3	12	5.4144 × 10 ⁻¹ (3.16 × 10 ⁻³) -	5.4220 × 10 ⁻¹ (2.65 × 10 ⁻³) -	5.5947 × 10⁻¹ (3.80 × 10⁻³) -	5.5278 × 10 ⁻¹ (1.94 × 10 ⁻³)
DTLZ3	2	11	0.0000 × 10 ⁰ (0.00 × 10 ⁰) =	0.0000 × 10 ⁰ (0.00 × 10 ⁰) =	2.5915 × 10 ⁻¹ (6.32 × 10 ⁻²) +	0.0000 × 10 ⁰ (0.00 × 10 ⁰)
	3	12	0.0000 × 10 ⁰ (0.00 × 10 ⁰) =	0.0000 × 10 ⁰ (0.00 × 10 ⁰) =	2.0017 × 10 ⁻¹ (2.09 × 10 ⁻¹) +	0.0000 × 10 ⁰ (0.00 × 10 ⁰)
DTLZ4	2	11	2.8682 × 10 ⁻¹ (1.10 × 10 ⁻¹) -	2.8669 × 10 ⁻¹ (1.10 × 10 ⁻¹) -	2.6177 × 10 ⁻¹ (1.23 × 10 ⁻¹) -	3.4733 × 10⁻¹ (7.32 × 10⁻⁵)
	3	12	5.3362 × 10 ⁻¹ (3.36 × 10 ⁻³) -	5.0357 × 10 ⁻¹ (1.12 × 10 ⁻¹) -	3.9433 × 10 ⁻¹ (1.71 × 10 ⁻¹) =	5.3966 × 10⁻¹ (3.31 × 10⁻³)
DTLZ5	2	11	3.4643 × 10 ⁻¹ (1.26 × 10 ⁻⁴) -	3.4643 × 10 ⁻¹ (1.22 × 10 ⁻⁴) -	3.4720 × 10 ⁻¹ (4.70 × 10 ⁻⁶) -	3.4744 × 10⁻¹ (6.74 × 10⁻⁵)
	3	12	1.9630 × 10 ⁻¹ (9.65 × 10 ⁻⁴) -	1.9766 × 10 ⁻¹ (4.40 × 10 ⁻⁴) -	1.8191 × 10 ⁻¹ (2.68 × 10 ⁻⁵) -	1.9972 × 10⁻¹ (1.87 × 10⁻⁴)
DTLZ6	2	11	3.4755 × 10⁻¹ (3.67 × 10⁻⁵) =	3.4755 × 10 ⁻¹ (5.26 × 10 ⁻⁵) =	3.4721 × 10 ⁻¹ (1.01 × 10 ⁻⁵) -	3.4753 × 10 ⁻¹ (4.45 × 10 ⁻⁵)
	3	12	2.0018 × 10⁻¹ (4.09 × 10⁻⁵) =	2.0017 × 10 ⁻¹ (4.09 × 10 ⁻⁵) =	1.8187 × 10 ⁻¹ (1.12 × 10 ⁻⁴) -	2.0017 × 10 ⁻¹ (3.98 × 10 ⁻⁵)
DTLZ7	2	21	2.3843 × 10 ⁻¹ (1.70 × 10 ⁻²) -	2.3839 × 10 ⁻¹ (1.69 × 10 ⁻²) -	2.2259 × 10 ⁻¹ (2.88 × 10 ⁻²) -	2.4294 × 10⁻¹ (1.03 × 10⁻⁵)
	3	22	2.7138 × 10 ⁻¹ (1.60 × 10 ⁻³) -	2.6245 × 10 ⁻¹ (1.85 × 10 ⁻²) -	2.5032 × 10 ⁻¹ (1.32 × 10 ⁻²) -	2.7871 × 10⁻¹ (5.17 × 10⁻⁴)
WFG1	2	11	5.5400 × 10 ⁻³ (8.56 × 10 ⁻³) -	3.3397 × 10 ⁻¹ (4.65 × 10 ⁻²) -	6.2054 × 10 ⁻¹ (1.24 × 10 ⁻²) +	5.0330 × 10 ⁻¹ (5.46 × 10 ⁻²)
	3	12	1.0248 × 10 ⁻¹ (2.21 × 10 ⁻²) -	3.1667 × 10 ⁻¹ (1.98 × 10 ⁻²) -	8.4128 × 10 ⁻¹ (3.92 × 10 ⁻²) +	5.6834 × 10 ⁻¹ (5.89 × 10 ⁻²)
WFG2	2	11	6.3175 × 10 ⁻¹ (3.41 × 10 ⁻⁴) =	6.3196 × 10 ⁻¹ (3.95 × 10 ⁻⁴) =	6.1218 × 10 ⁻¹ (8.57 × 10 ⁻³) -	6.1720 × 10 ⁻¹ (3.33 × 10 ⁻²)
	3	12	9.2764 × 10⁻¹ (1.28 × 10⁻³) +	9.2689 × 10 ⁻¹ (2.04 × 10 ⁻³) +	8.9243 × 10 ⁻¹ (1.45 × 10 ⁻²) -	9.1668 × 10 ⁻¹ (3.63 × 10 ⁻²)
WFG3	2	11	5.7980 × 10 ⁻¹ (3.01 × 10 ⁻⁴) +	5.7966 × 10 ⁻¹ (3.02 × 10 ⁻⁴) +	5.7376 × 10 ⁻¹ (2.14 × 10 ⁻³) +	5.7214 × 10 ⁻¹ (2.70 × 10 ⁻²)
	3	12	3.2577 × 10 ⁻¹ (6.90 × 10 ⁻³) -	3.5440 × 10 ⁻¹ (7.68 × 10 ⁻³) -	3.5180 × 10 ⁻¹ (1.12 × 10 ⁻²) -	3.6348 × 10 ⁻¹ (3.13 × 10 ⁻²)
WFG4	2	11	3.1782 × 10 ⁻¹ (1.16 × 10 ⁻³) -	3.2833 × 10 ⁻¹ (6.54 × 10 ⁻³) -	3.3684 × 10 ⁻¹ (1.71 × 10 ⁻³) -	3.3972 × 10 ⁻¹ (3.23 × 10 ⁻³)
	3	12	4.8848 × 10 ⁻¹ (3.95 × 10 ⁻³) -	4.9065 × 10 ⁻¹ (3.94 × 10 ⁻³) -	5.2865 × 10⁻¹ (4.09 × 10⁻³) +	5.0911 × 10 ⁻¹ (8.68 × 10 ⁻³)
WFG5	2	11	3.1084 × 10 ⁻¹ (1.58 × 10 ⁻³) -	3.1021 × 10 ⁻¹ (2.03 × 10 ⁻³) -	3.0645 × 10 ⁻¹ (6.45 × 10 ⁻⁴) -	3.1363 × 10⁻¹ (1.39 × 10⁻⁴)
	3	12	4.8677 × 10 ⁻¹ (5.07 × 10 ⁻³) -	4.8050 × 10 ⁻¹ (5.54 × 10 ⁻³) -	4.9739 × 10 ⁻¹ (3.17 × 10 ⁻³) -	5.0496 × 10⁻¹ (3.75 × 10⁻³)
WFG6	2	11	3.4199 × 10⁻¹ (6.34 × 10⁻³) +	3.4151 × 10 ⁻¹ (4.22 × 10 ⁻³) +	2.9466 × 10 ⁻¹ (1.19 × 10 ⁻²) +	2.2697 × 10 ⁻¹ (5.36 × 10 ⁻⁵)
	3	12	5.2590 × 10⁻¹ (5.49 × 10⁻³) +	5.1572 × 10 ⁻¹ (8.77 × 10 ⁻³) +	4.7715 × 10 ⁻¹ (1.76 × 10 ⁻²) +	4.1628 × 10 ⁻¹ (8.87 × 10 ⁻⁴)
WFG7	2	11	3.4528 × 10 ⁻¹ (2.57 × 10 ⁻⁴) -	3.4525 × 10 ⁻¹ (2.83 × 10 ⁻⁴) -	3.3629 × 10 ⁻¹ (1.67 × 10 ⁻³) -	3.4719 × 10⁻¹ (1.15 × 10⁻⁴)
	3	12	5.3049 × 10 ⁻¹ (3.25 × 10 ⁻³) -	5.2333 × 10 ⁻¹ (3.58 × 10 ⁻³) -	5.0286 × 10 ⁻¹ (2.07 × 10 ⁻²) -	5.5302 × 10⁻¹ (2.17 × 10⁻³)
WFG8	2	11	2.8272 × 10 ⁻¹ (1.61 × 10 ⁻³) -	2.8459 × 10 ⁻¹ (1.61 × 10 ⁻³) -	2.7964 × 10 ⁻¹ (4.85 × 10 ⁻³) -	2.8525 × 10 ⁻¹ (1.69 × 10 ⁻³)
	3	12	4.2466 × 10 ⁻¹ (5.19 × 10 ⁻³) -	4.2870 × 10 ⁻¹ (5.08 × 10 ⁻³) -	4.5009 × 10⁻¹ (6.01 × 10⁻³) +	4.4087 × 10 ⁻¹ (6.67 × 10 ⁻³)
WFG9	2	11	3.3360 × 10 ⁻¹ (1.95 × 10 ⁻³) +	3.3452 × 10 ⁻¹ (1.99 × 10 ⁻³) +	3.0765 × 10 ⁻¹ (3.14 × 10 ⁻²) +	2.2643 × 10 ⁻¹ (1.04 × 10 ⁻³)
	3	12	4.9830 × 10 ⁻¹ (4.16 × 10 ⁻²) +	5.1324 × 10⁻¹ (4.38 × 10⁻³) +	4.6505 × 10 ⁻¹ (2.97 × 10 ⁻²) +	4.0623 × 10 ⁻¹ (2.68 × 10 ⁻³)
ZDT1	2	30	7.1962 × 10 ⁻¹ (1.51 × 10 ⁻⁴) -	7.1928 × 10 ⁻¹ (2.26 × 10 ⁻⁴) -	7.0904 × 10 ⁻¹ (9.57 × 10 ⁻³) -	7.2049 × 10⁻¹ (5.25 × 10⁻⁵)
ZDT2	2	30	4.4442 × 10 ⁻¹ (1.64 × 10 ⁻⁴) -	4.4401 × 10 ⁻¹ (2.07 × 10 ⁻⁴) -	4.0696 × 10 ⁻¹ (4.05 × 10 ⁻²) -	4.4510 × 10⁻¹ (4.02 × 10⁻⁵)
ZDT3	2	30	5.9952 × 10 ⁻¹ (9.36 × 10 ⁻⁵) -	5.9965 × 10 ⁻¹ (1.10 × 10 ⁻⁴) -	6.2106 × 10⁻¹ (5.00 × 10⁻²) =	5.9979 × 10 ⁻¹ (2.39 × 10 ⁻⁵)
ZDT4	2	10	0.0000 × 10 ⁰ (0.00 × 10 ⁰) =	4.4020 × 10 ⁻¹ (2.14 × 10 ⁻¹) +	6.9311 × 10 ⁻¹ (1.65 × 10 ⁻²) +	0.0000 × 10 ⁰ (0.00 × 10 ⁰)
+/-/=			6/22/8	8/22/6	15/19/2	
Problem	M	D	NSGAI	MOEADDE	MMOPSO	CMOSCA
DTLZ1	2	6	5.7995 × 10 ⁻¹ (1.60 × 10 ⁻³) +	5.8013 × 10⁻¹ (4.16 × 10⁻³) +	5.6631 × 10 ⁻¹ (6.57 × 10 ⁻²) +	0.0000 × 10 ⁰ (0.00 × 10 ⁰)
	3	7	7.9637 × 10 ⁻¹ (1.21 × 10 ⁻¹) +	6.9729 × 10 ⁻¹ (2.31 × 10 ⁻¹) +	2.7383 × 10 ⁻¹ (3.22 × 10 ⁻¹) +	0.0000 × 10 ⁰ (0.00 × 10 ⁰)
DTLZ2	2	11	3.4656 × 10 ⁻¹ (2.11 × 10 ⁻⁴) -	3.4702 × 10 ⁻¹ (3.78 × 10 ⁻⁵) -	3.4668 × 10 ⁻¹ (2.19 × 10 ⁻⁴) -	3.4743 × 10⁻¹ (5.68 × 10⁻⁵)
	3	12	5.2873 × 10 ⁻¹ (4.91 × 10 ⁻³) -	5.2691 × 10 ⁻¹ (1.65 × 10 ⁻³) -	5.3071 × 10 ⁻¹ (4.40 × 10 ⁻³) -	5.5278 × 10 ⁻¹ (1.94 × 10 ⁻³)
DTLZ3	2	11	3.0575 × 10⁻¹ (8.42 × 10⁻²) +	5.6429 × 10 ⁻² (1.19 × 10 ⁻¹) +	0.0000 × 10 ⁰ (0.00 × 10 ⁰) =	0.0000 × 10 ⁰ (0.00 × 10 ⁰)
	3	12	3.8921 × 10⁻¹ (1.62 × 10⁻¹) +	1.8937 × 10 ⁻¹ (2.22 × 10 ⁻¹) +	1.5222 × 10 ⁻² (8.34 × 10 ⁻²) =	0.0000 × 10 ⁰ (0.00 × 10 ⁰)
DTLZ4	2	11	3.1245 × 10 ⁻¹ (8.84 × 10 ⁻²) -	3.4683 × 10 ⁻¹ (6.56 × 10 ⁻⁵) -	3.0403 × 10 ⁻¹ (9.70 × 10 ⁻²) -	3.4733 × 10⁻¹ (7.32 × 10⁻⁵)
	3	12	5.1573 × 10 ⁻¹ (8.03 × 10 ⁻²) -	5.1937 × 10 ⁻¹ (2.29 × 10 ⁻²) -		

Table 4. Cont.

Problem	M	D	EMOSO	CMOPSO	MOEAD	CMOSCA
DTLZ5	2	11	3.4650×10^{-1} (2.07×10^{-4}) -	3.4701×10^{-1} (3.21×10^{-5}) -	3.4664×10^{-1} (1.78×10^{-4}) -	3.4744×10^{-1} (6.74×10^{-5})
	3	12	1.9874×10^{-1} (2.30×10^{-4}) -	1.9438×10^{-1} (7.71×10^{-5}) -	1.9916×10^{-1} (1.77×10^{-4}) -	1.9972×10^{-1} (1.87×10^{-4})
DTLZ6	2	11	3.4638×10^{-1} (2.33×10^{-4}) -	3.4721×10^{-1} (7.87×10^{-8}) -	3.4663×10^{-1} (2.18×10^{-4}) -	3.4753×10^{-1} (4.45×10^{-5})
	3	12	1.9912×10^{-1} (1.80×10^{-4}) -	1.9477×10^{-1} (2.22×10^{-5}) -	1.9922×10^{-1} (1.59×10^{-4}) -	2.0017×10^{-1} (3.98×10^{-5})
DTLZ7	2	21	2.4272×10^{-1} (4.13×10^{-5}) -	2.2855×10^{-1} (2.69×10^{-2}) -	2.1603×10^{-1} (3.32×10^{-2}) -	2.4294×10^{-1} (1.03×10^{-5})
	3	22	2.6689×10^{-1} (2.12×10^{-3}) -	2.0996×10^{-1} (1.76×10^{-2}) -	2.5880×10^{-1} (1.52×10^{-2}) -	2.7871×10^{-1} (5.17×10^{-4})
WFG1	2	11	6.6884×10^{-1} (1.32×10^{-2}) +	4.9843×10^{-1} (4.87×10^{-2}) =	6.5123×10^{-1} (2.76×10^{-2}) +	5.0330×10^{-1} (5.46×10^{-2})
	3	12	9.2184×10^{-1} (5.97×10^{-3}) +	4.0981×10^{-1} (5.21×10^{-2}) -	8.1347×10^{-1} (2.81×10^{-2}) +	5.6834×10^{-1} (5.89×10^{-2})
WFG2	2	11	6.3234×10^{-1} (4.78×10^{-4}) +	6.2802×10^{-1} (9.83×10^{-4}) +	6.3307×10^{-1} (1.90×10^{-4}) +	6.1720×10^{-1} (3.33×10^{-2})
	3	12	9.1559×10^{-1} (3.51×10^{-3}) -	8.7644×10^{-1} (6.11×10^{-3}) -	9.1401×10^{-1} (3.42×10^{-3}) -	9.1668×10^{-1} (3.63×10^{-2})
WFG3	2	11	5.7929×10^{-1} (6.66×10^{-4}) +	5.7806×10^{-1} (6.61×10^{-4}) +	5.8024×10^{-1} (4.29×10^{-4}) +	5.7214×10^{-1} (2.70×10^{-2})
	3	12	3.9521×10^{-1} (3.69×10^{-3}) +	3.4249×10^{-1} (1.55×10^{-2}) -	3.9426×10^{-1} (5.41×10^{-3}) +	3.6348×10^{-1} (3.13×10^{-2})
WFG4	2	11	3.4575×10^{-1} (2.63×10^{-4}) +	3.2351×10^{-1} (3.49×10^{-3}) -	3.4422×10^{-1} (6.90×10^{-4}) +	3.3972×10^{-1} (3.23×10^{-3})
	3	12	5.1694×10^{-1} (6.01×10^{-3}) +	4.6432×10^{-1} (7.04×10^{-3}) -	4.9689×10^{-1} (6.50×10^{-3}) -	5.0911×10^{-1} (8.68×10^{-3})
WFG5	2	11	3.1238×10^{-1} (1.39×10^{-3}) -	3.0803×10^{-1} (1.69×10^{-3}) -	3.1289×10^{-1} (1.73×10^{-4}) -	3.1363×10^{-1} (1.39×10^{-4})
	3	12	4.8840×10^{-1} (5.07×10^{-3}) -	4.5556×10^{-1} (3.66×10^{-3}) -	4.8551×10^{-1} (7.37×10^{-3}) -	5.0496×10^{-1} (3.75×10^{-3})
WFG6	2	11	3.0608×10^{-1} (1.10×10^{-2}) +	3.1454×10^{-1} (4.13×10^{-2}) +	3.1757×10^{-1} (3.76×10^{-2}) +	2.2697×10^{-1} (5.36×10^{-5})
	3	12	4.7001×10^{-1} (1.56×10^{-2}) +	4.2788×10^{-1} (4.57×10^{-2}) +	4.6580×10^{-1} (4.83×10^{-2}) +	4.1628×10^{-1} (8.87×10^{-4})
WFG7	2	11	3.4503×10^{-1} (3.59×10^{-4}) -	3.4474×10^{-1} (2.90×10^{-4}) -	3.4583×10^{-1} (2.75×10^{-4}) -	3.4719×10^{-1} (1.15×10^{-4})
	3	12	5.1679×10^{-1} (4.32×10^{-3}) -	4.9114×10^{-1} (5.76×10^{-3}) -	5.2276×10^{-1} (5.08×10^{-3}) -	5.5302×10^{-1} (2.17×10^{-3})
WFG8	2	11	2.8775×10^{-1} (6.89×10^{-4}) +	2.8973×10^{-1} (2.51×10^{-3}) +	2.8842×10^{-1} (1.33×10^{-3}) +	2.8525×10^{-1} (1.69×10^{-3})
	3	12	4.3610×10^{-1} (4.36×10^{-3}) -	3.9117×10^{-1} (9.68×10^{-3}) -	4.3457×10^{-1} (5.02×10^{-3}) -	4.4087×10^{-1} (6.67×10^{-3})
WFG9	2	11	3.3651×10^{-1} (2.11×10^{-2}) +	3.3126×10^{-1} (2.28×10^{-3}) +	3.3914×10^{-1} (2.07×10^{-3}) +	2.2643×10^{-1} (1.04×10^{-3})
	3	12	4.8142×10^{-1} (3.13×10^{-2}) +	4.7633×10^{-1} (5.60×10^{-3}) +	4.9105×10^{-1} (2.08×10^{-2}) +	4.0623×10^{-1} (2.68×10^{-3})
ZDT1	2	30	7.1910×10^{-1} (2.07×10^{-4}) -	7.0598×10^{-1} (4.70×10^{-3}) -	7.1932×10^{-1} (2.26×10^{-4}) -	7.2049×10^{-1} (5.25×10^{-5})
ZDT2	2	30	4.4383×10^{-1} (2.18×10^{-4}) -	4.3332×10^{-1} (3.65×10^{-3}) -	4.4400×10^{-1} (2.21×10^{-4}) -	4.4510×10^{-1} (4.02×10^{-5})
ZDT3	2	30	6.0228×10^{-1} (1.62×10^{-2}) +	5.9542×10^{-1} (6.91×10^{-3}) -	5.9942×10^{-1} (9.92×10^{-5}) -	5.9979×10^{-1} (2.39×10^{-5})
ZDT4	2	10	7.1724×10^{-1} (1.44×10^{-3}) +	4.8618×10^{-1} (1.38×10^{-1}) +	6.9739×10^{-1} (6.24×10^{-2}) +	0.0000×10^0 (0.00×10^0)
+/-/=			18/18/0	12/23/1	14/20/2	

“+”, “=” and “-” indicate that the results of competing MOEAs are statistically superior, similar, and inferior to results obtained by the CMOSCA, respectively. The best result of each test problem is displayed in bold.

Figures 4–7 plot the non-dominated solutions obtained by the CMOSCA and other six competing MOEAs on two-objective DTLZ6 and ZDT3, and three-objective DTLZ7 and WFG6. It can be observed from Figures 4 and 5 that the non-dominated solutions obtained by our proposed CMOSCA on two-objective DTLZ6 and ZDT3 problems have a good distribution and approximate the PF well. This exhibits that the CMOSCA can attain a good trade-off between the diversity and convergence of non-dominated solutions. We can also observe from Figures 6 and 7 that the CMOSCA is able to obtain non-dominated solutions with good diversity and convergence on three-objective problems DTLZ7 and WFG6.

To investigate the computation cost of the CMOSCA, we record the actual running time of those competing MOEAs. Figure 8 shows the average runtimes of the seven competing MOEAs tested on ZDT, DTLZ and WFG series test problems. From Figure 8, we can find that the NSGA-II obtains the best performance in terms of computation cost because the simple dominance relationship selection strategy has significant advantages in real-time computation. We can also find that CMOSCA performs better than MOEA/D and MOEA/D-DE, but weaker than EMOSO, CMOPSO, NSGA-II, and MMOPSO. Hence, we can conclude that the computation time of our proposed CMOSCA is not significantly reduced compared to other competing MOEAs, which is caused by evaluating the SDE fitness values in CMOSCA. However, for MOPs, better performance of the algorithm is more important than computational cost. Therefore, we believe that the computational cost of CMOSCA is acceptable since it can provide decision-makers with a better basis for decision-making.

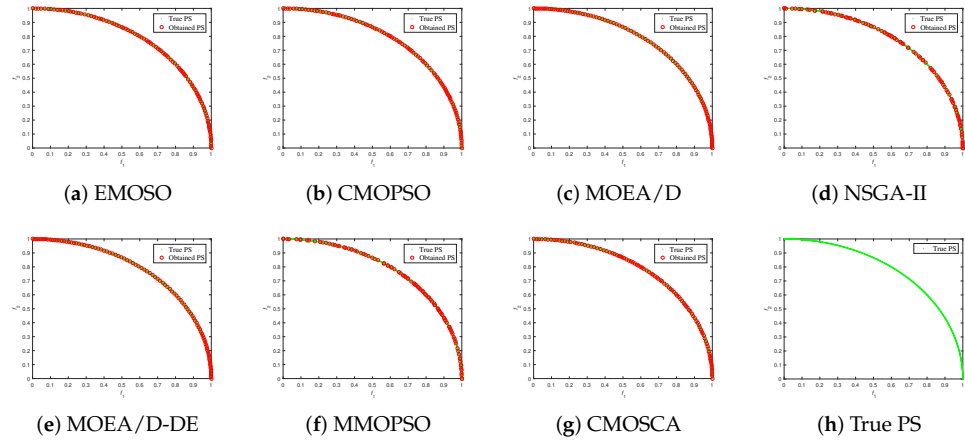


Figure 4. Nondominated solutions obtained by the six competing MOEAs and CMOSCA on two-objective DTLZ6 problem.

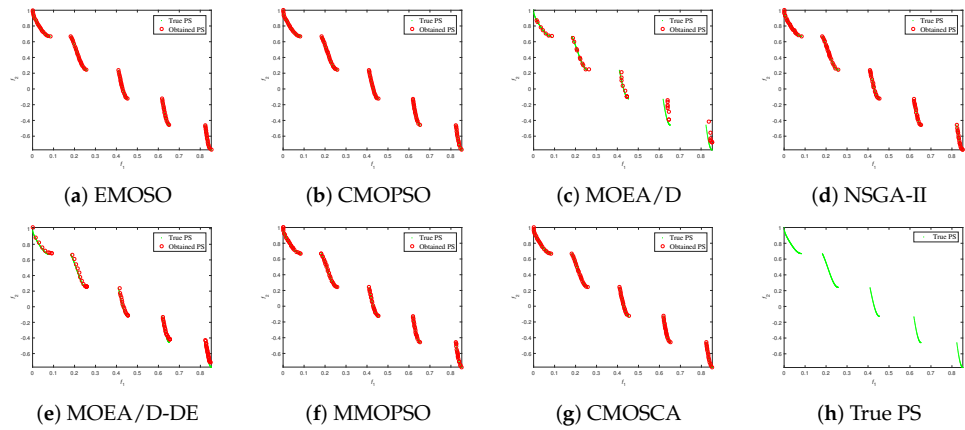


Figure 5. Nondominated solutions obtained by the six competing MOEAs and CMOSCA on two-objective ZDT3 problem.

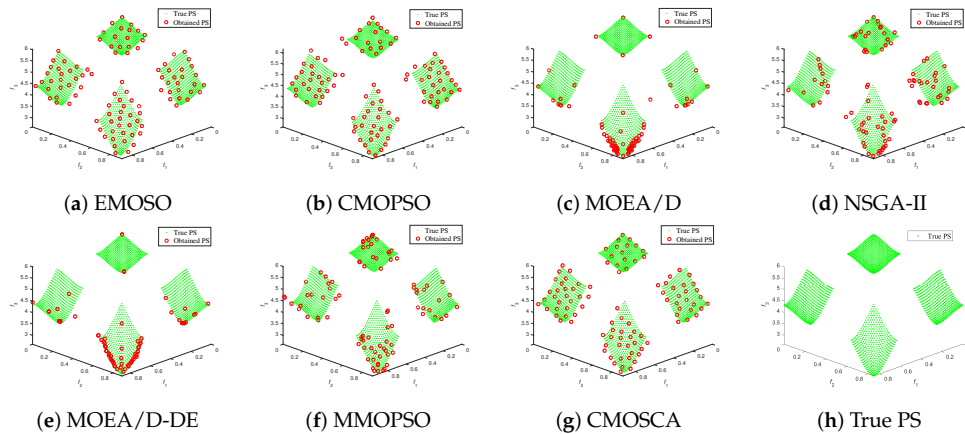


Figure 6. Nondominated solutions obtained by the six competing MOEAs and CMOSCA on three-objective DTLZ7 problem.

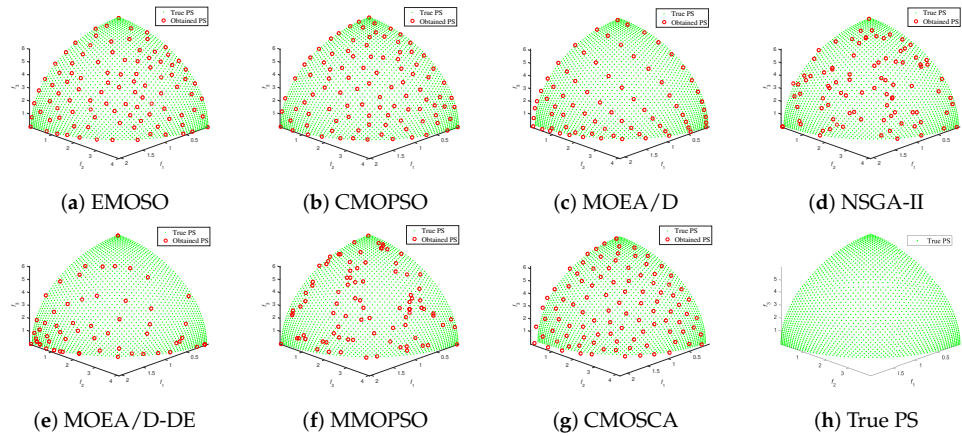


Figure 7. Nondominated solutions obtained by the six competing MOEAs and CMOSCA on three-objective WFG6 problem.

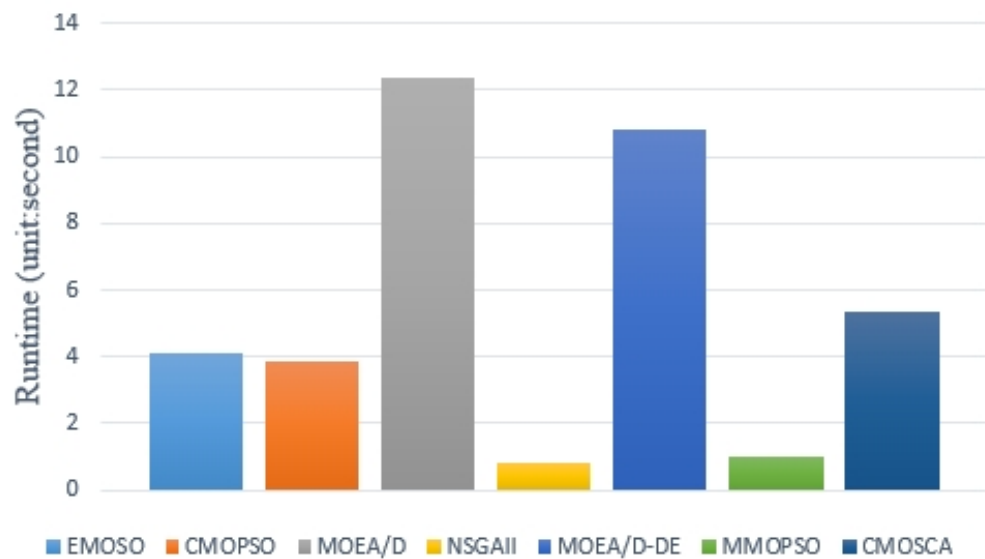


Figure 8. Average runtimes of the all competing methods tested on the all the benchmark problems.

4.2. Parameter Analysis

In the CMOSCA, the parameter β defines the size of elite agent set E , which has a great influence on the performance of our proposed CMOSCA, and the outstanding agents in the elite set are adopted to lead the search of the population. Experiments are performed to analyze CMOSCA with different β values (varying from 2 to 30). The other parameter value settings of the CMOSCA are the same as in the previous section. Table 5 reports the experiment IGD values of 30 independent running on DTLZ1-DTLZ7, WFG1-WFG9, and ZDT1-ZDT4 test problems. We can observe from the Table 5 that the CMOSCA using the parameter $\beta = 5$ has relatively superior results. Table 6 lists the experiment HV values of 30 independent running on DTLZ1-DTLZ7, WFG1-WFG9, and ZDT1-ZDT4 test problems. It can be seen from the Table 6 that CMOSCA using the parameter $\beta = 2$ has relatively superior results. From these two tables, we can find that the CMOSCA with small β values is better. In this study, we set β to 5 for our proposed CMOSCA to deal with MOPs.

Table 5. IGD values achieved by CMOSCA using different β values.

Problem	M	D	CMOSCA $\beta = 2$	CMOSCA $\beta = 10$	CMOSCA $\beta = 15$	CMOSCA $\beta = 5$
DTLZ1	2	6	$1.9342 \times 10^1 (5.45 \times 10^0) =$	$1.5759 \times 10^1 (5.60 \times 10^0) =$	$1.7927 \times 10^1 (4.60 \times 10^0) =$	$1.8233 \times 10^1 (6.02 \times 10^0)$
	3	7	$1.1176 \times 10^1 (3.34 \times 10^0) =$	$1.0587 \times 10^1 (4.83 \times 10^0) =$	$1.2098 \times 10^1 (4.92 \times 10^0) =$	$1.1081 \times 10^1 (4.81 \times 10^0)$
DTLZ2	2	11	$4.1321 \times 10^{-3} (2.54 \times 10^{-5}) =$	$4.1045 \times 10^{-3} (2.59 \times 10^{-5}) =$	$4.1115 \times 10^{-3} (3.16 \times 10^{-5}) =$	$4.1149 \times 10^{-3} (3.78 \times 10^{-5})$
	3	12	$5.3615 \times 10^{-2} (6.60 \times 10^{-4}) -$	$5.3771 \times 10^{-2} (6.50 \times 10^{-4}) -$	$5.4166 \times 10^{-2} (7.16 \times 10^{-4}) -$	$5.3243 \times 10^{-2} (4.56 \times 10^{-4})$
DTLZ3	2	11	$1.7068 \times 10^2 (1.35 \times 10^1) -$	$1.6755 \times 10^2 (2.07 \times 10^1) -$	$1.6684 \times 10^2 (1.54 \times 10^1) -$	$1.5902 \times 10^2 (1.94 \times 10^1)$
	3	12	$1.5333 \times 10^2 (1.49 \times 10^1) +$	$1.6358 \times 10^2 (1.43 \times 10^1) =$	$1.6016 \times 10^2 (1.60 \times 10^1) =$	$1.6203 \times 10^2 (1.60 \times 10^1)$
DTLZ4	2	11	$4.1405 \times 10^{-3} (4.41 \times 10^{-5}) -$	$4.1098 \times 10^{-3} (2.08 \times 10^{-5}) =$	$4.1039 \times 10^{-3} (3.59 \times 10^{-5}) =$	$4.1119 \times 10^{-3} (3.26 \times 10^{-5})$
	3	12	$5.7988 \times 10^{-2} (1.69 \times 10^{-3}) +$	$5.8669 \times 10^{-2} (2.07 \times 10^{-3}) =$	$5.8298 \times 10^{-2} (1.97 \times 10^{-3}) +$	$5.9577 \times 10^{-2} (1.47 \times 10^{-3})$
DTLZ5	2	11	$4.1298 \times 10^{-3} (2.95 \times 10^{-5}) -$	$4.1037 \times 10^{-3} (3.67 \times 10^{-5}) =$	$4.1130 \times 10^{-3} (3.83 \times 10^{-5}) =$	$4.1107 \times 10^{-3} (2.65 \times 10^{-5})$
	3	12	$4.5261 \times 10^{-3} (2.47 \times 10^{-4}) -$	$4.3021 \times 10^{-3} (1.38 \times 10^{-4}) =$	$4.3323 \times 10^{-3} (1.75 \times 10^{-4}) =$	$4.3665 \times 10^{-3} (1.69 \times 10^{-4})$
DTLZ6	2	11	$4.1265 \times 10^{-3} (2.99 \times 10^{-5}) =$	$4.1003 \times 10^{-3} (3.25 \times 10^{-5}) =$	$4.0799 \times 10^{-3} (3.55 \times 10^{-5}) +$	$4.1140 \times 10^{-3} (3.03 \times 10^{-5})$
	3	12	$4.2171 \times 10^{-3} (8.36 \times 10^{-5}) =$	$4.1822 \times 10^{-3} (4.75 \times 10^{-5}) =$	$4.1835 \times 10^{-3} (4.99 \times 10^{-5}) =$	$4.1788 \times 10^{-3} (5.83 \times 10^{-5})$
DTLZ7	2	21	$4.5212 \times 10^{-3} (4.63 \times 10^{-5}) =$	$4.5097 \times 10^{-3} (6.11 \times 10^{-5}) =$	$4.5075 \times 10^{-3} (5.73 \times 10^{-5}) =$	$4.5057 \times 10^{-3} (5.13 \times 10^{-5})$
	3	22	$5.8902 \times 10^{-2} (1.13 \times 10^{-3}) =$	$5.8975 \times 10^{-2} (1.10 \times 10^{-3}) =$	$5.8410 \times 10^{-2} (8.37 \times 10^{-4}) +$	$5.8941 \times 10^{-2} (6.68 \times 10^{-4})$
WFG1	2	11	$4.4875 \times 10^{-1} (1.15 \times 10^{-1}) -$	$3.9312 \times 10^{-1} (1.50 \times 10^{-1}) =$	$3.4205 \times 10^{-1} (6.73 \times 10^{-2}) =$	$3.8033 \times 10^{-1} (1.19 \times 10^{-1})$
	3	12	$7.0762 \times 10^{-1} (1.32 \times 10^{-1}) +$	$1.1179 \times 10^0 (1.55 \times 10^{-1}) -$	$1.2398 \times 10^0 (1.14 \times 10^{-1}) -$	$8.5394 \times 10^{-1} (1.43 \times 10^{-1})$
WFG2	2	11	$3.5297 \times 10^{-2} (4.81 \times 10^{-2}) +$	$3.1870 \times 10^{-2} (4.89 \times 10^{-2}) =$	$2.1731 \times 10^{-2} (3.61 \times 10^{-2}) =$	$3.5635 \times 10^{-2} (5.38 \times 10^{-2})$
	3	12	$1.9292 \times 10^{-1} (3.11 \times 10^{-2}) =$	$1.9270 \times 10^{-1} (3.23 \times 10^{-2}) =$	$1.8012 \times 10^{-1} (4.91 \times 10^{-3}) =$	$1.9064 \times 10^{-1} (3.08 \times 10^{-2})$
WFG3	2	11	$1.0300 \times 10^{-1} (9.72 \times 10^{-2}) -$	$2.3611 \times 10^{-2} (4.27 \times 10^{-2}) =$	$1.8118 \times 10^{-2} (3.09 \times 10^{-2}) =$	$2.9280 \times 10^{-2} (5.16 \times 10^{-2})$
	3	12	$1.7046 \times 10^{-1} (6.91 \times 10^{-2}) -$	$1.1792 \times 10^{-1} (1.20 \times 10^{-2}) =$	$1.2634 \times 10^{-1} (3.23 \times 10^{-2}) =$	$1.3883 \times 10^{-1} (6.04 \times 10^{-2})$
WFG4	2	11	$2.5007 \times 10^{-2} (6.96 \times 10^{-3}) -$	$2.1590 \times 10^{-2} (4.26 \times 10^{-3}) =$	$2.1172 \times 10^{-2} (5.39 \times 10^{-3}) =$	$2.0532 \times 10^{-2} (4.51 \times 10^{-3})$
	3	12	$2.4046 \times 10^{-1} (1.03 \times 10^{-2}) =$	$2.5109 \times 10^{-1} (6.54 \times 10^{-3}) -$	$2.5137 \times 10^{-1} (4.65 \times 10^{-3}) -$	$2.4095 \times 10^{-1} (8.39 \times 10^{-3})$
WFG5	2	11	$6.3656 \times 10^{-2} (3.17 \times 10^{-4}) =$	$6.4203 \times 10^{-2} (1.56 \times 10^{-3}) -$	$6.5213 \times 10^{-2} (2.10 \times 10^{-3}) -$	$6.3608 \times 10^{-2} (1.83 \times 10^{-4})$
	3	12	$2.2368 \times 10^{-1} (3.67 \times 10^{-3}) =$	$2.2769 \times 10^{-1} (6.21 \times 10^{-3}) -$	$2.2713 \times 10^{-1} (4.49 \times 10^{-3}) -$	$2.2262 \times 10^{-1} (3.68 \times 10^{-3})$
WFG6	2	11	$2.2547 \times 10^{-1} (2.73 \times 10^{-5}) =$	$2.2547 \times 10^{-1} (2.81 \times 10^{-5}) =$	$2.2546 \times 10^{-1} (2.27 \times 10^{-5}) =$	$2.2547 \times 10^{-1} (1.68 \times 10^{-5})$
	3	12	$3.3557 \times 10^{-1} (1.70 \times 10^{-3}) =$	$3.3865 \times 10^{-1} (5.34 \times 10^{-3}) -$	$3.4089 \times 10^{-1} (6.47 \times 10^{-3}) -$	$3.3555 \times 10^{-1} (1.77 \times 10^{-3})$
WFG7	2	11	$1.2707 \times 10^{-2} (2.23 \times 10^{-4}) =$	$1.2798 \times 10^{-2} (1.62 \times 10^{-4}) -$	$1.2839 \times 10^{-2} (1.68 \times 10^{-4}) -$	$1.2639 \times 10^{-2} (1.05 \times 10^{-4})$
	3	12	$2.1060 \times 10^{-1} (2.51 \times 10^{-3}) =$	$2.1305 \times 10^{-1} (2.58 \times 10^{-3}) -$	$2.1501 \times 10^{-1} (3.23 \times 10^{-3}) -$	$2.1066 \times 10^{-1} (1.88 \times 10^{-3})$
WFG8	2	11	$1.1816 \times 10^{-1} (7.35 \times 10^{-3}) =$	$1.1589 \times 10^{-1} (3.96 \times 10^{-3}) =$	$1.1436 \times 10^{-1} (2.76 \times 10^{-3}) +$	$1.1660 \times 10^{-1} (3.17 \times 10^{-3})$
	3	12	$3.0929 \times 10^{-1} (9.22 \times 10^{-3}) +$	$3.1856 \times 10^{-1} (6.98 \times 10^{-3}) =$	$3.2552 \times 10^{-1} (9.20 \times 10^{-3}) -$	$3.1606 \times 10^{-1} (9.77 \times 10^{-3})$
WFG9	2	11	$2.1662 \times 10^{-1} (3.48 \times 10^{-2}) +$	$2.1669 \times 10^{-1} (4.58 \times 10^{-2}) +$	$2.2253 \times 10^{-1} (3.17 \times 10^{-2}) +$	$2.2676 \times 10^{-1} (2.47 \times 10^{-3})$
	3	12	$3.4457 \times 10^{-1} (3.72 \times 10^{-3}) =$	$3.4867 \times 10^{-1} (5.88 \times 10^{-3}) -$	$3.4698 \times 10^{-1} (4.62 \times 10^{-3}) -$	$3.4464 \times 10^{-1} (3.88 \times 10^{-3})$
ZDT1	2	30	$3.9024 \times 10^{-3} (3.96 \times 10^{-5}) =$	$3.8799 \times 10^{-3} (5.26 \times 10^{-5}) =$	$3.8795 \times 10^{-3} (4.55 \times 10^{-5}) =$	$3.8994 \times 10^{-3} (3.98 \times 10^{-5})$
ZDT2	2	30	$3.8961 \times 10^{-3} (4.41 \times 10^{-5}) =$	$3.8674 \times 10^{-3} (2.93 \times 10^{-5}) +$	$3.8695 \times 10^{-3} (4.29 \times 10^{-5}) +$	$3.8933 \times 10^{-3} (3.48 \times 10^{-5})$
ZDT3	2	30	$4.6500 \times 10^{-3} (4.80 \times 10^{-5}) +$	$4.6965 \times 10^{-3} (5.97 \times 10^{-5}) =$	$4.6808 \times 10^{-3} (6.32 \times 10^{-5}) =$	$4.6982 \times 10^{-3} (7.19 \times 10^{-5})$
ZDT4	2	10	$2.2937 \times 10^1 (1.94 \times 10^1) =$	$2.2757 \times 10^1 (2.03 \times 10^1) =$	$1.9949 \times 10^1 (2.04 \times 10^1) =$	$1.7501 \times 10^1 (1.63 \times 10^1)$
+/-/=			7/9/20	2/10/24	6/11/19	
Problem	M	D	CMOSCA $\beta = 2$	CMOSCA $\beta = 10$	CMOSCA $\beta = 15$	CMOSCA $\beta = 5$
DTLZ1	2	6	$1.6761 \times 10^1 (3.94 \times 10^0) =$	$1.5874 \times 10^1 (3.94 \times 10^0) =$	$1.7088 \times 10^1 (3.40 \times 10^0) =$	$1.8233 \times 10^1 (6.02 \times 10^0)$
	3	7	$1.1537 \times 10^1 (4.01 \times 10^0) =$	$1.0875 \times 10^1 (4.65 \times 10^0) =$	$9.7277 \times 10^0 (3.74 \times 10^0) =$	$1.1081 \times 10^1 (4.81 \times 10^0)$
DTLZ2	2	11	$4.0992 \times 10^{-3} (2.59 \times 10^{-5}) =$	$4.1089 \times 10^{-3} (3.28 \times 10^{-5}) =$	$4.1011 \times 10^{-3} (2.87 \times 10^{-5}) =$	$4.1149 \times 10^{-3} (3.78 \times 10^{-5})$
	3	12	$5.4058 \times 10^{-2} (7.27 \times 10^{-4}) -$	$5.4141 \times 10^{-2} (6.94 \times 10^{-4}) -$	$5.4381 \times 10^{-2} (7.65 \times 10^{-4}) -$	$5.3243 \times 10^{-2} (4.56 \times 10^{-4})$
DTLZ3	2	11	$1.6731 \times 10^2 (1.56 \times 10^1) =$	$1.6652 \times 10^2 (1.58 \times 10^1) =$	$1.6064 \times 10^2 (1.71 \times 10^1) =$	$1.5902 \times 10^2 (1.94 \times 10^1)$
	3	12	$1.6290 \times 10^2 (1.42 \times 10^1) =$	$1.5810 \times 10^2 (1.35 \times 10^1) =$	$1.6358 \times 10^2 (1.34 \times 10^1) =$	$1.6203 \times 10^2 (1.60 \times 10^1)$
DTLZ4	2	11	$4.1022 \times 10^{-3} (3.43 \times 10^{-5}) =$	$4.1235 \times 10^{-3} (3.78 \times 10^{-5}) =$	$4.1188 \times 10^{-3} (3.42 \times 10^{-5}) =$	$4.1119 \times 10^{-3} (3.26 \times 10^{-5})$
	3	12	$5.8481 \times 10^{-2} (2.17 \times 10^{-3}) +$	$5.8062 \times 10^{-2} (1.63 \times 10^{-3}) +$	$5.7916 \times 10^{-2} (1.36 \times 10^{-3}) +$	$5.9577 \times 10^{-2} (1.47 \times 10^{-3})$
DTLZ5	2	11	$4.1053 \times 10^{-3} (3.52 \times 10^{-5}) =$	$4.1058 \times 10^{-3} (3.48 \times 10^{-5}) =$	$4.1101 \times 10^{-3} (2.91 \times 10^{-5}) =$	$4.1107 \times 10^{-3} (2.65 \times 10^{-5})$
	3	12	$4.3433 \times 10^{-3} (2.32 \times 10^{-4}) =$	$4.3389 \times 10^{-3} (1.90 \times 10^{-4}) =$	$4.3671 \times 10^{-3} (2.53 \times 10^{-4}) =$	$4.3665 \times 10^{-3} (1.69 \times 10^{-4})$
DTLZ6	2	11	$4.0799 \times 10^{-3} (3.35 \times 10^{-5}) +$	$4.0871 \times 10^{-3} (2.94 \times 10^{-5}) +$	$4.0828 \times 10^{-3} (3.18 \times 10^{-5}) +$	$4.1140 \times 10^{-3} (3.03 \times 10^{-5})$
	3	12	$4.1752 \times 10^{-3} (4.29 \times 10^{-5}) =$	$4.1549 \times 10^{-3} (3.64 \times 10^{-5}) +$	$4.1770 \times 10^{-3} (4.24 \times 10^{-5}) =$	$4.1788 \times 10^{-3} (5.83 \times 10^{-5})$
DTLZ7	2	21	$4.4963 \times 10^{-3} (4.94 \times 10^{-5}) =$	$4.4795 \times 10^{-3} (5.82 \times 10^{-5}) =$	$4.4983 \times 10^{-3} (5.59 \times 10^{-5}) =$	$4.5057 \times 10^{-3} (5.13 \times 10^{-5})$
	3	22	$5.8313 \times 10^{-2} (8.07 \times 10^{-4}) +$	$5.8443 \times 10^{-2} (1.06 \times 10^{-3}) +$	$5.8670 \times 10^{-2} (1.23 \times 10^{-3}) =$	$5.8941 \times 10^{-2} (6.68 \times 10^{-4})$
WFG1	2	11	$3.7226 \times 10^{-1} (8.60 \times 10^{-2}) =$	$4.5902 \times 10^{-1} (8.82 \times 10^{-2}) -$	$4.3078 \times 10^{-1} (5.81 \times 10^{-2}) -$	$3.8033 \times 10^{-1} (1.19 \times 10^{-1})$
	3	12	$1.2694 \times 10^0 (1.05 \times 10^{-1}) -$	$1.2971 \times 10^0 (9.35 \times 10^{-2}) -$	$1.3041 \times 10^0 (1.05 \times 10^{-1}) -$	$8.5394 \times 10^{-1} (1.43 \times 10^{-1})$
WFG2	2	11	$2.1843 \times 10^{-2} (3.60 \times 10^{-2}) =$	$1.7349 \times 10^{-2} (2.59 \times 10^{-2}) +$	$2.1994 \times 10^{-2} (3.63 \times 10^{-2}) =$	$3.5635 \times 10^{-2} (5.38 \times 10^{-2})$
	3	12	$1.8193 \times 10^{-1} (4.51 \times 10^{-3}) =$	$1.8330 \times 10^{-1} (1.89 \times 10^{-2}) =$	$1.8331 \times 10^{-1} (7.77 \times 10^{-3}) =$	$1.9064 \times 10^{-1} (3.08 \times 10^{-2})$
WFG3	2	11	$1.2539 \times 10^{-2} (3.20 \times 10^{-4}) =$	$1.8201 \times 10^{-2} (3.10 \times 10^{-2}) =$	$1.8254 \times 10^{-2} (3.08 \times 10^{-2}) +$	$2.9280 \times 10^{-2} (5.16 \times 10^{-2})$
	3	12	$1.2889 \times 10^{-1} (9.48 \times 10^{-3}) +$	$1.3009 \times 10^{-1} (1.47 \times 10^{-2}) +$	$1.3413 \times 10^{-1} (1.38 \times 10^{-2}) +$	$1.3883 \times 10^{-1} (6.04 \times 10^{-2})$
WFG4	2	11	$2.4712 \times 10^{-2} (8.76 \times 10^{-3}) =$	$2.3005 \times 10^{-2} (6.39 \times 10^{-3}) =$	$2.7480 \times 10^{-2} (8.18 \times 10^{-3}) -$	$2.0532 \times 10^{-2} (4.51 \times 10^{-3})$
	3	12	$2.5300 \times 10^{-1} (4.36 \times 10^{-3}) -$	$2.5269 \times 10^{-1} (5.29 \times 10^{-3}) -$	$2.5231 \times 10^{-1} (4.47 \times 10^{-3}) -$	$2.4095 \times 10^{-1} (8.39 \times 10^{-3})$
WFG5	2	11	$6.5284 \times 10^{-2} (2.41 \times 10^{-3}) -$	$6.5157 \times 10^{-2} (2.16 \times 10^{-3}) -$	$6.6438 \times 10^{-2} (2.55 \times 10^{-3}) -$	$6.3608 \times 10^{-2} (1.83 \times 10^{-4})$
	3	12	$2.2939 \times 10^{-1} (7.01 \times 10^{-3}) -$	$2.3148 \times 10^{-1} (6.98 \times 10^{-3}) -$	$2.3074 \times 10^{-1} (6.44 \times 10^{-3}) -$	$2.2262 \times 10^{-1} (3.68 \times 10^{-3})$
WFG6	2	11	$2.2547 \times 10^{-1} (2.02 \times 10^{-5}) =$	$2.2547 \times 10^{-1} (2.90 \times 10^{-5}) =$	$2.2547 \times 10^{-1} (3.34 \times 10^{-5}) =$	$2.2547 \times 10^{-1} (1.68 \times 10^{-5})$
	3	12	$3.4413 \times 10^{-1} (1.00 \times 10^{-2}) -$	$3.4564 \times 10^{-1} (1.06 \times 10^{-2}) -$	$3.4362 \times 10^{-1} (6.62 \times 10^{-3}) -$	$3.3555 \times 10^{-1} (1.77 \times 10^{-3})$
WFG7	2	11	$1.2856 \times 10^{-2} (1.60 \times 10^{-4}) -$	$1.2914 \times 10^{-2} (1.83 \times 10^{-4}) -$	$1.2925 \times 10^{-2} (1.68 \times 10^{-4}) -$	$1.2639 \times 10^{-2} (1.05 \times 10^{-4})$
	3	12	$2.1524 \times 10^{-1} (2.53 \times 10^{-3}) -$	$2.1690 \times 10^{-1} (2.85 \times 10^{-3}) -$	$2.1856 \times 10^{-1} (3.63 \times 10^{-3}) -$	$2.1066 \times 10^{-1} (1.88 \times 10^{-3})$
WFG8	2	11	$1.1472 \times 10^{-1} (2.69 \times 10^{-3}) +$	$1.1336 \times 10^{-1} (1.18 \times 10^{-3}) +$	$1.1368 \times 10^{-1} (2.55 \times 10^{-3}) +$	$1.1660 \times 10^{-1} (3.17 \times 10^{-3})$
	3	12	$3.2575 \times 10^{-1} (7.52 \times 10^{-3}) -$	$3.2585 \times 10^{-1} (6.85 \times 10^{-3}) -$	$3.2562 \times 10^{-1} (8.46 \times 10^{-3}) -$	$3.1606 \times 10^{-1} (9.77 \times 10^{-3})$
WFG9	2	11	$2.2603 \times 10^{-1} (1.26 \times 10^{-2}) +$	$2.2263 \times 10^{-1} (3.45 \times 10^{-2}) +$	$2.2985 \times 10^{-1} (3.06 \times 10^{-3}) -$ </	

Table 5. Cont.

Problem	M	D	CMOSCA $\beta = 2$	CMOSCA $\beta = 10$	CMOSCA $\beta = 15$	CMOSCA $\beta = 5$
ZDT1	2	30	3.8759×10^{-3} (4.42×10^{-5}) =	3.8755×10^{-3} (3.30×10^{-5}) =	3.8648×10^{-3} (4.20×10^{-5}) +	3.8994×10^{-3} (4.98×10^{-5})
ZDT2	2	30	3.8570×10^{-3} (2.88×10^{-5}) +	3.8829×10^{-3} (3.72×10^{-5}) =	3.8800×10^{-3} (4.25×10^{-5}) =	3.8933×10^{-3} (3.48×10^{-5})
ZDT3	2	30	4.6941×10^{-3} (5.65×10^{-5}) =	4.6906×10^{-3} (7.38×10^{-5}) =	4.6678×10^{-3} (6.23×10^{-5}) =	4.6982×10^{-3} (7.19×10^{-5})
ZDT4	2	10	1.0998×10^1 (1.03×10^1) +	1.7827×10^1 (1.64×10^1) =	1.4417×10^1 (1.35×10^1) =	1.7501×10^1 (1.63×10^1)
+/-/=			8/10/18	8/11/17	6/13/17	

“+”, “=” and “-” indicate that the results of competing CMOSCA using different β values are statistically superior, similar, and inferior to results obtained by the CMOSCA using $\beta = 5$ values, respectively. The best result of each test problem is displayed in bold.

Table 6. HV values achieved by CMOSCA using different β values.

Problem	M	D	CMOSCA $\beta = 2$	CMOSCA $\beta = 10$	CMOSCA $\beta = 15$	CMOSCA $\beta = 5$
DTLZ1	2	6	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0)
	3	7	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0) =	8.1450×10^{-6} (4.46×10^{-5}) =	0.0000×10^0 (0.00×10^0)
DTLZ2	2	11	3.4752×10^{-1} (4.20×10^{-3}) +	3.4736×10^{-1} (7.13×10^{-3}) -	3.4732×10^{-1} (7.84×10^{-3}) -	3.4743×10^{-1} (5.68×10^{-3})
	3	12	5.5109×10^{-1} (2.45×10^{-3}) -	5.5078×10^{-1} (1.82×10^{-3}) -	5.4892×10^{-1} (2.29×10^{-3}) -	5.5278×10^{-1} (1.94×10^{-3})
DTLZ3	2	11	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0)
	3	12	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0)
DTLZ4	2	11	3.4728×10^{-1} (1.31×10^{-4}) =	3.4725×10^{-1} (8.88×10^{-5}) -	3.4720×10^{-1} (7.92×10^{-5}) -	3.4733×10^{-1} (7.32×10^{-5})
	3	12	5.4209×10^{-1} (3.87×10^{-3}) +	5.4082×10^{-1} (4.38×10^{-3}) =	5.4194×10^{-1} (3.67×10^{-3}) +	5.3966×10^{-1} (3.31×10^{-3})
DTLZ5	2	11	3.4753×10^{-1} (4.47×10^{-5}) +	3.4736×10^{-1} (6.34×10^{-5}) -	3.4730×10^{-1} (6.47×10^{-5}) -	3.4744×10^{-1} (6.74×10^{-5})
	3	12	1.9949×10^{-1} (2.54×10^{-4}) -	1.9971×10^{-1} (1.72×10^{-4}) =	1.9971×10^{-1} (1.63×10^{-4}) =	1.9972×10^{-1} (1.87×10^{-4})
DTLZ6	2	11	3.4754×10^{-1} (4.62×10^{-5}) =	3.4752×10^{-1} (5.84×10^{-5}) =	3.4753×10^{-1} (4.14×10^{-5}) =	3.4753×10^{-1} (4.45×10^{-5})
	3	12	2.0018×10^{-1} (5.24×10^{-5}) =	2.0018×10^{-1} (2.72×10^{-5}) =	2.0017×10^{-1} (3.78×10^{-5}) =	2.0017×10^{-1} (3.98×10^{-5})
DTLZ7	2	21	2.4294×10^{-1} (9.08×10^{-6}) =	2.4294×10^{-1} (9.57×10^{-6}) =	2.4294×10^{-1} (1.21×10^{-5}) =	2.4294×10^{-1} (1.03×10^{-5})
	3	22	2.7867×10^{-1} (8.10×10^{-4}) =	2.7869×10^{-1} (7.01×10^{-4}) =	2.7870×10^{-1} (7.05×10^{-4}) =	2.7871×10^{-1} (5.17×10^{-4})
WFG1	2	11	4.7170×10^{-1} (5.12×10^{-2}) -	4.8977×10^{-1} (6.80×10^{-2}) =	5.1255×10^{-1} (3.70×10^{-2}) =	5.0330×10^{-1} (5.46×10^{-2})
	3	12	6.2541×10^{-1} (6.57×10^{-2}) +	4.4784×10^{-1} (6.09×10^{-2}) -	3.9320×10^{-1} (3.77×10^{-2}) -	5.6834×10^{-1} (5.89×10^{-2})
WFG2	2	11	6.1647×10^{-1} (2.97×10^{-2}) -	6.1919×10^{-1} (3.02×10^{-2}) =	6.2561×10^{-1} (2.23×10^{-2}) =	6.1720×10^{-1} (3.33×10^{-2})
	3	12	9.1747×10^{-1} (3.67×10^{-2}) =	9.1371×10^{-1} (3.63×10^{-2}) =	9.2543×10^{-1} (1.76×10^{-2}) +	9.1668×10^{-1} (3.63×10^{-2})
WFG3	2	11	5.3376×10^{-1} (5.01×10^{-2}) =	5.7511×10^{-1} (2.24×10^{-2}) =	5.7790×10^{-1} (1.61×10^{-2}) =	5.7214×10^{-1} (2.70×10^{-2})
	3	12	3.4594×10^{-1} (3.59×10^{-2}) -	3.7349×10^{-1} (7.44×10^{-3}) =	3.6804×10^{-1} (1.69×10^{-2}) =	3.6348×10^{-1} (3.13×10^{-2})
WFG4	2	11	3.3661×10^{-1} (4.56×10^{-3}) -	3.3898×10^{-1} (2.94×10^{-3}) =	3.3972×10^{-1} (3.14×10^{-3}) =	3.3972×10^{-1} (3.23×10^{-3})
	3	12	5.1155×10^{-1} (9.66×10^{-3}) =	4.9935×10^{-1} (5.02×10^{-3}) -	4.9652×10^{-1} (4.61×10^{-3}) -	5.0911×10^{-1} (8.68×10^{-3})
WFG5	2	11	3.1363×10^{-1} (1.93×10^{-4}) =	3.1304×10^{-1} (1.43×10^{-3}) -	3.1202×10^{-1} (1.98×10^{-3}) -	3.1363×10^{-1} (1.39×10^{-4})
	3	12	5.0575×10^{-1} (4.61×10^{-3}) =	5.0120×10^{-1} (5.25×10^{-3}) -	5.0072×10^{-1} (3.72×10^{-3}) -	5.0496×10^{-1} (3.75×10^{-3})
WFG6	2	11	2.2699×10^{-1} (5.14×10^{-5}) =	2.2697×10^{-1} (5.25×10^{-5}) =	2.2699×10^{-1} (4.78×10^{-5}) =	2.2697×10^{-1} (5.36×10^{-5})
	3	12	4.1675×10^{-1} (7.97×10^{-4}) +	4.1517×10^{-1} (2.78×10^{-3}) =	4.1348×10^{-1} (3.54×10^{-3}) =	4.1628×10^{-1} (8.87×10^{-4})
WFG7	2	11	3.4734×10^{-1} (9.62×10^{-5}) +	3.4688×10^{-1} (1.27×10^{-4}) -	3.4670×10^{-1} (1.31×10^{-4}) -	3.4719×10^{-1} (1.15×10^{-4})
	3	12	5.5486×10^{-1} (1.92×10^{-3}) +	5.4486×10^{-1} (2.45×10^{-3}) -	5.4097×10^{-1} (3.47×10^{-3}) -	5.5302×10^{-1} (2.17×10^{-3})
WFG8	2	11	2.8421×10^{-1} (3.93×10^{-3}) =	2.8558×10^{-1} (1.94×10^{-3}) =	2.8630×10^{-1} (1.36×10^{-3}) +	2.8525×10^{-1} (1.69×10^{-3})
	3	12	4.4568×10^{-1} (6.95×10^{-3}) +	4.3986×10^{-1} (5.26×10^{-3}) =	4.3361×10^{-1} (6.77×10^{-3}) -	4.4087×10^{-1} (6.67×10^{-3})
WFG9	2	11	2.3244×10^{-1} (1.93×10^{-2}) +	2.3192×10^{-1} (2.41×10^{-2}) +	2.2874×10^{-1} (1.65×10^{-2}) +	2.2643×10^{-1} (1.04×10^{-3})
	3	12	4.0653×10^{-1} (2.88×10^{-3}) =	4.0391×10^{-1} (3.56×10^{-3}) -	4.0433×10^{-1} (2.39×10^{-3}) -	4.0623×10^{-1} (2.68×10^{-3})
ZDT1	2	30	7.2054×10^{-1} (2.95×10^{-5}) +	7.2050×10^{-1} (6.05×10^{-5}) =	7.2048×10^{-1} (6.05×10^{-5}) =	7.2049×10^{-1} (5.25×10^{-5})
ZDT2	2	30	4.4509×10^{-1} (3.58×10^{-5}) =	4.4510×10^{-1} (2.96×10^{-5}) =	4.4507×10^{-1} (3.64×10^{-5}) -	4.4510×10^{-1} (4.02×10^{-5})
ZDT3	2	30	5.9981×10^{-1} (1.98×10^{-5}) +	5.9979×10^{-1} (2.13×10^{-5}) =	5.9979×10^{-1} (2.38×10^{-5}) =	5.9979×10^{-1} (2.39×10^{-5})
ZDT4	2	10	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0)
+/-/=			11/7/18	1/12/23	4/14/18	

Problem	M	D	CMOSCA $\beta = 2$	CMOSCA $\beta = 10$	CMOSCA $\beta = 15$	CMOSCA $\beta = 5$
DTLZ1	2	6	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0)
	3	7	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0)
DTLZ2	2	11	3.4729×10^{-1} (8.35×10^{-5}) -	3.4728×10^{-1} (7.07×10^{-5}) -	3.4727×10^{-1} (8.73×10^{-5}) -	3.4743×10^{-1} (5.68×10^{-5})
	3	12	5.4915×10^{-1} (1.81×10^{-3}) -	5.4899×10^{-1} (2.12×10^{-3}) -	5.4782×10^{-1} (2.21×10^{-3}) -	5.5278×10^{-1} (1.94×10^{-3})
DTLZ3	2	11	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0)
	3	12	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0)
DTLZ4	2	11	3.4719×10^{-1} (1.08×10^{-4}) -	3.4718×10^{-1} (1.25×10^{-4}) -	3.4719×10^{-1} (9.34×10^{-5}) -	3.4733×10^{-1} (7.32×10^{-5})
	3	12	5.4133×10^{-1} (4.38×10^{-3}) +	5.4152×10^{-1} (3.51×10^{-3}) +	5.4120×10^{-1} (3.08×10^{-3}) =	5.3966×10^{-1} (3.31×10^{-3})
DTLZ5	2	11	3.4731×10^{-1} (7.37×10^{-5}) -	3.4727×10^{-1} (7.93×10^{-5}) -	3.4724×10^{-1} (7.99×10^{-5}) -	3.4744×10^{-1} (6.74×10^{-5})
	3	12	1.9967×10^{-1} (1.92×10^{-4}) =	1.9968×10^{-1} (1.78×10^{-4}) =	1.9963×10^{-1} (2.07×10^{-4}) =	1.9972×10^{-1} (1.87×10^{-4})
DTLZ6	2	11	3.4754×10^{-1} (4.09×10^{-5}) =	3.4754×10^{-1} (4.31×10^{-5}) =	3.4753×10^{-1} (4.83×10^{-5}) =	3.4753×10^{-1} (4.45×10^{-5})
	3	12	2.0018×10^{-1} (4.17×10^{-5}) =	2.0016×10^{-1} (3.80×10^{-5}) =	2.0019×10^{-1} (3.24×10^{-5}) =	2.0017×10^{-1} (3.98×10^{-5})
DTLZ7	2	21	2.4294×10^{-1} (1.09×10^{-5}) =	2.4295×10^{-1} (1.23×10^{-5}) =	2.4295×10^{-1} (1.08×10^{-5}) =	2.4294×10^{-1} (1.03×10^{-5})
	3	22	2.7868×10^{-1} (7.45×10^{-4}) =	2.7842×10^{-1} (8.96×10^{-4}) =	2.7850×10^{-1} (5.86×10^{-4}) =	2.7871×10^{-1} (5.17×10^{-4})
WFG1	2	11	4.9539×10^{-1} (4.76×10^{-2}) =	4.4960×10^{-1} (4.55×10^{-2}) -	4.6903×10^{-1} (2.90×10^{-2}) -	5.0330×10^{-1} (5.46×10^{-2})
	3	12	3.7740×10^{-1} (3.59×10^{-2}) -	3.6193×10^{-1} (3.29×10^{-2}) -	3.5814×10^{-1} (3.51×10^{-2}) -	5.6834×10^{-1} (5.89×10^{-2})
WFG2	2	11	6.2556×10^{-1} (2.23×10^{-2}) =	6.2816×10^{-1} (1.60×10^{-2}) +	6.2541×10^{-1} (2.24×10^{-2}) =	6.1720×10^{-1} (3.33×10^{-2})
	3	12	9.2445×10^{-1} (2.07×10^{-3}) +	9.2069×10^{-1} (1.90×10^{-2}) +	9.2197×10^{-1} (2.12×10^{-3}) +	9.1668×10^{-1} (3.63×10^{-2})

Table 6. Cont.

Problem	M	D	CMOSCA $\beta = 2$	CMOSCA $\beta = 10$	CMOSCA $\beta = 15$	CMOSCA $\beta = 5$
WFG3	2	11	5.8079 $\times 10^{-1}$ (3.54 $\times 10^{-4}$) =	5.7781 $\times 10^{-1}$ (1.62 $\times 10^{-2}$) =	5.7776 $\times 10^{-1}$ (1.61 $\times 10^{-2}$) +	5.7214 $\times 10^{-1}$ (2.70 $\times 10^{-2}$)
	3	12	3.6678 $\times 10^{-1}$ (6.44 $\times 10^{-3}$) +	3.6538 $\times 10^{-1}$ (7.94 $\times 10^{-3}$) +	3.6401 $\times 10^{-1}$ (7.27 $\times 10^{-3}$) +	3.6348 $\times 10^{-1}$ (3.13 $\times 10^{-2}$)
WFG4	2	11	3.3771 $\times 10^{-1}$ (4.69 $\times 10^{-3}$) =	3.3886 $\times 10^{-1}$ (3.29 $\times 10^{-3}$) =	3.3634 $\times 10^{-1}$ (4.43 $\times 10^{-3}$) -	3.3972 $\times 10^{-1}$ (3.23 $\times 10^{-3}$)
	3	12	4.9573 $\times 10^{-1}$ (3.69 $\times 10^{-3}$) -	4.9622 $\times 10^{-1}$ (4.49 $\times 10^{-3}$) -	4.9613 $\times 10^{-1}$ (3.74 $\times 10^{-3}$) -	5.0911 $\times 10^{-1}$ (8.68 $\times 10^{-3}$)
WFG5	2	11	3.1203 $\times 10^{-1}$ (2.09 $\times 10^{-3}$) -	3.1198 $\times 10^{-1}$ (1.94 $\times 10^{-3}$) -	3.1099 $\times 10^{-1}$ (2.19 $\times 10^{-3}$) -	3.1363 $\times 10^{-1}$ (1.39 $\times 10^{-4}$)
	3	12	4.9873 $\times 10^{-1}$ (6.00 $\times 10^{-3}$) -	4.9789 $\times 10^{-1}$ (5.59 $\times 10^{-3}$) -	4.9782 $\times 10^{-1}$ (6.11 $\times 10^{-3}$) -	5.0496 $\times 10^{-1}$ (3.75 $\times 10^{-3}$)
WFG6	2	11	2.2697 $\times 10^{-1}$ (4.73 $\times 10^{-5}$) =	2.2699 $\times 10^{-1}$ (4.88 $\times 10^{-5}$) +	2.2698 $\times 10^{-1}$ (4.77 $\times 10^{-5}$) =	2.2697 $\times 10^{-1}$ (5.36 $\times 10^{-5}$)
	3	12	4.1189 $\times 10^{-1}$ (5.26 $\times 10^{-3}$) -	4.1146 $\times 10^{-1}$ (6.09 $\times 10^{-3}$) -	4.1248 $\times 10^{-1}$ (3.92 $\times 10^{-3}$) -	4.1628 $\times 10^{-1}$ (8.87 $\times 10^{-4}$)
WFG7	2	11	3.4662 $\times 10^{-1}$ (1.46 $\times 10^{-4}$) -	3.4660 $\times 10^{-1}$ (1.60 $\times 10^{-4}$) -	3.4657 $\times 10^{-1}$ (1.79 $\times 10^{-4}$) -	3.4719 $\times 10^{-1}$ (1.15 $\times 10^{-4}$)
	3	12	5.3768 $\times 10^{-1}$ (3.58 $\times 10^{-3}$) -	5.3586 $\times 10^{-1}$ (2.44 $\times 10^{-3}$) -	5.3436 $\times 10^{-1}$ (3.19 $\times 10^{-3}$) -	5.5302 $\times 10^{-1}$ (2.17 $\times 10^{-3}$)
WFG8	2	11	2.8623 $\times 10^{-1}$ (1.22 $\times 10^{-3}$) +	2.8674 $\times 10^{-1}$ (5.45 $\times 10^{-4}$) +	2.8665 $\times 10^{-1}$ (1.19 $\times 10^{-3}$) +	2.8525 $\times 10^{-1}$ (1.69 $\times 10^{-3}$)
	3	12	4.3398 $\times 10^{-1}$ (4.56 $\times 10^{-3}$) -	4.3449 $\times 10^{-1}$ (4.51 $\times 10^{-3}$) -	4.3398 $\times 10^{-1}$ (6.53 $\times 10^{-3}$) -	4.4087 $\times 10^{-1}$ (6.67 $\times 10^{-3}$)
WFG9	2	11	2.2721 $\times 10^{-1}$ (8.25 $\times 10^{-3}$) +	2.2889 $\times 10^{-1}$ (1.86 $\times 10^{-2}$) +	2.2512 $\times 10^{-1}$ (1.20 $\times 10^{-3}$) -	2.2643 $\times 10^{-1}$ (1.04 $\times 10^{-3}$)
	3	12	4.0290 $\times 10^{-1}$ (2.87 $\times 10^{-3}$) -	4.0390 $\times 10^{-1}$ (2.62 $\times 10^{-3}$) -	4.0374 $\times 10^{-1}$ (2.82 $\times 10^{-3}$) -	4.0623 $\times 10^{-1}$ (2.68 $\times 10^{-3}$)
ZDT1	2	30	7.2048 $\times 10^{-1}$ (7.51 $\times 10^{-5}$) =	7.2049 $\times 10^{-1}$ (4.55 $\times 10^{-5}$) =	7.2047 $\times 10^{-1}$ (4.25 $\times 10^{-5}$) =	7.2049 $\times 10^{-1}$ (5.25 $\times 10^{-5}$)
ZDT2	2	30	4.4509 $\times 10^{-1}$ (3.48 $\times 10^{-5}$) =	4.4507 $\times 10^{-1}$ (4.90 $\times 10^{-5}$) -	4.4507 $\times 10^{-1}$ (4.96 $\times 10^{-5}$) =	4.4510 $\times 10^{-1}$ (4.02 $\times 10^{-5}$)
ZDT3	2	30	5.9979 $\times 10^{-1}$ (2.23 $\times 10^{-5}$) =	5.9979 $\times 10^{-1}$ (2.81 $\times 10^{-5}$) =	5.9980 $\times 10^{-1}$ (2.39 $\times 10^{-5}$) =	5.9979 $\times 10^{-1}$ (2.39 $\times 10^{-5}$)
ZDT4	2	10	0.0000 $\times 10^0$ (0.00 $\times 10^0$) =	0.0000 $\times 10^0$ (0.00 $\times 10^0$) =	0.0000 $\times 10^0$ (0.00 $\times 10^0$) =	0.0000 $\times 10^0$ (0.00 $\times 10^0$)
+/-/=			5/13/18	7/15/14	4/17/15	

“+”, “=” and “-” indicate that the results of competing CMOSCA using different β values are statistically superior, similar, and inferior to results obtained by the CMOSCA using $\beta = 5$ values, respectively. The best result of each test problem is displayed in bold.

4.3. Comparisons among Three CMOSCA Variants

To verify the effects of the SDE-based competition mechanism on the performance of the CMOSCA, the SDE-based competition mechanism is compared with the other two approaches (i.e., angle-based competition mechanism and Euclidean distance-based competition mechanism). The CMOSCA with these two approaches are denoted as the CMOSCAA and the CMOSCAD, respectively. Experiments are performed to compare these three CMOSCA variants. The parameter value settings for these three CMOSCA variants are the same as in the previous section. Table 7 reports the experiment IGD values of 30 independent running on DTLZ1-DTLZ7, WFG1-WFG9 and ZDT1-ZDT4 test problems. It can be observed from the Table 7 that the proposed CMOSCA outperforms the other two variants. Table 8 provides the experiment HV values of 30 independent running on DTLZ1-DTLZ7, WFG1-WFG9 and ZDT1-ZDT4 test problems. From the Table 8, we can also observe that the proposed CMOSCA outperforms the other two variants. This is because the SDE-based competition mechanism can measure both diversity and convergence of the solutions.

Table 7. IGD values achieved by three CMOSCA variants.

Problem	M	D	CMOSCAA	CMOSCAD	CMOSCA
DTLZ1	2	6	1.8997 $\times 10^1$ (6.26 $\times 10^0$) =	1.8496 $\times 10^1$ (6.55 $\times 10^0$) =	1.8233 $\times 10^1$ (6.02 $\times 10^0$)
	3	7	1.1280 $\times 10^1$ (4.33 $\times 10^0$) =	1.0678 $\times 10^1$ (4.45 $\times 10^0$) =	1.1081 $\times 10^1$ (4.81 $\times 10^0$)
DTLZ2	2	11	4.1038 $\times 10^{-3}$ (3.45 $\times 10^{-5}$) =	4.1125 $\times 10^{-3}$ (3.84 $\times 10^{-5}$) =	4.1149 $\times 10^{-3}$ (3.78 $\times 10^{-5}$)
	3	12	5.3713 $\times 10^{-2}$ (5.98 $\times 10^{-4}$) -	5.2984 $\times 10^{-2}$ (5.50 $\times 10^{-4}$) =	5.3243 $\times 10^{-2}$ (4.56 $\times 10^{-4}$)
DTLZ3	2	11	1.6913 $\times 10^2$ (1.87 $\times 10^1$) -	1.7087 $\times 10^2$ (1.80 $\times 10^1$) -	1.5902 $\times 10^2$ (1.94 $\times 10^1$)
	3	12	1.5527 $\times 10^2$ (1.72 $\times 10^1$) =	1.6192 $\times 10^2$ (1.65 $\times 10^1$) =	1.6203 $\times 10^2$ (1.60 $\times 10^1$)
DTLZ4	2	11	4.1188 $\times 10^{-3}$ (3.66 $\times 10^{-5}$) =	4.1171 $\times 10^{-3}$ (3.60 $\times 10^{-5}$) =	4.1119 $\times 10^{-3}$ (3.26 $\times 10^{-5}$)
	3	12	5.9014 $\times 10^{-2}$ (1.80 $\times 10^{-3}$) =	5.7189 $\times 10^{-2}$ (1.96 $\times 10^{-3}$) +	5.9577 $\times 10^{-2}$ (1.47 $\times 10^{-3}$)
DTLZ5	2	11	4.1033 $\times 10^{-3}$ (4.02 $\times 10^{-5}$) =	4.1142 $\times 10^{-3}$ (2.93 $\times 10^{-5}$) =	4.1107 $\times 10^{-3}$ (2.65 $\times 10^{-5}$)
	3	12	5.1442 $\times 10^{-3}$ (5.49 $\times 10^{-4}$) -	4.2607 $\times 10^{-3}$ (1.03 $\times 10^{-4}$) +	4.3665 $\times 10^{-3}$ (1.69 $\times 10^{-4}$)
DTLZ6	2	11	4.0768 $\times 10^{-3}$ (2.82 $\times 10^{-5}$) +	4.0938 $\times 10^{-3}$ (2.60 $\times 10^{-5}$) +	4.1140 $\times 10^{-3}$ (3.03 $\times 10^{-5}$)
	3	12	4.1680 $\times 10^{-3}$ (5.58 $\times 10^{-5}$) =	4.1686 $\times 10^{-3}$ (4.40 $\times 10^{-5}$) =	4.1788 $\times 10^{-3}$ (5.83 $\times 10^{-5}$)
DTLZ7	2	21	4.4714 $\times 10^{-3}$ (5.49 $\times 10^{-5}$) +	4.5119 $\times 10^{-3}$ (7.67 $\times 10^{-5}$) =	4.5057 $\times 10^{-3}$ (5.13 $\times 10^{-5}$)
	3	22	5.8436 $\times 10^{-2}$ (1.33 $\times 10^{-3}$) +	5.8680 $\times 10^{-2}$ (1.23 $\times 10^{-3}$) =	5.8941 $\times 10^{-2}$ (6.68 $\times 10^{-4}$)
WFG1	2	11	3.6703 $\times 10^{-1}$ (1.09 $\times 10^{-1}$) =	5.7172 $\times 10^{-1}$ (1.42 $\times 10^{-1}$) -	3.8033 $\times 10^{-1}$ (1.19 $\times 10^{-1}$)
	3	12	1.1237 $\times 10^0$ (1.18 $\times 10^{-1}$) -	1.3241 $\times 10^0$ (1.15 $\times 10^{-1}$) -	8.5394 $\times 10^{-1}$ (1.43 $\times 10^{-1}$)
WFG2	2	11	4.5567 $\times 10^{-2}$ (6.09 $\times 10^{-2}$) =	7.4616 $\times 10^{-2}$ (7.14 $\times 10^{-2}$) =	3.5635 $\times 10^{-2}$ (5.38 $\times 10^{-2}$)
	3	12	1.8933 $\times 10^{-1}$ (2.77 $\times 10^{-2}$) =	2.1603 $\times 10^{-1}$ (4.28 $\times 10^{-2}$) -	1.9064 $\times 10^{-1}$ (3.08 $\times 10^{-2}$)
WFG3	2	11	2.9572 $\times 10^{-2}$ (5.16 $\times 10^{-2}$) -	4.0745 $\times 10^{-2}$ (6.41 $\times 10^{-2}$) -	2.9280 $\times 10^{-2}$ (5.16 $\times 10^{-2}$)
	3	12	1.4767 $\times 10^{-1}$ (2.29 $\times 10^{-2}$) -	1.6052 $\times 10^{-1}$ (6.33 $\times 10^{-2}$) -	1.3883 $\times 10^{-1}$ (6.04 $\times 10^{-2}$)

Table 7. Cont.

Problem	M	D	CMOSCAA	CMOSCAD	CMOSCA
WFG4	2	11	2.4884×10^{-2} (7.53×10^{-3}) -	2.5662×10^{-2} (9.48×10^{-3}) -	2.0532×10^{-2} (4.51×10^{-3})
	3	12	2.5274×10^{-1} (4.80×10^{-3}) -	2.4420×10^{-1} (8.85×10^{-3}) =	2.4095×10^{-1} (8.39×10^{-3})
WFG5	2	11	6.4533×10^{-2} (1.39×10^{-3}) -	6.3745×10^{-2} (1.06×10^{-3}) =	6.3608×10^{-2} (1.83×10^{-4})
	3	12	2.2664×10^{-1} (6.26×10^{-3}) -	2.2360×10^{-1} (4.37×10^{-3}) =	2.2262×10^{-1} (3.68×10^{-3})
WFG6	2	11	2.2547×10^{-1} (5.73×10^{-5}) -	2.2547×10^{-1} (2.02×10^{-5}) =	2.2547×10^{-1} (1.68×10^{-5})
	3	12	3.3692×10^{-1} (5.75×10^{-3}) =	3.3754×10^{-1} (4.19×10^{-3}) -	3.3555×10^{-1} (1.77×10^{-3})
WFG7	2	11	1.2834×10^{-2} (1.42×10^{-4}) -	1.2805×10^{-2} (1.88×10^{-4}) -	1.2639×10^{-2} (1.05×10^{-4})
	3	12	2.1510×10^{-1} (4.05×10^{-3}) -	2.1444×10^{-1} (3.11×10^{-3}) -	2.1066×10^{-1} (1.88×10^{-3})
WFG8	2	11	1.1509×10^{-1} (3.53×10^{-3}) +	1.1843×10^{-1} (4.11×10^{-3}) -	1.1660×10^{-1} (3.17×10^{-3})
	3	12	3.2616×10^{-1} (8.27×10^{-3}) -	3.2911×10^{-1} (1.08×10^{-2}) -	3.1606×10^{-1} (9.77×10^{-3})
WFG9	2	11	2.2322×10^{-1} (2.19×10^{-2}) +	2.2021×10^{-1} (3.29×10^{-2}) =	2.2676×10^{-1} (2.47×10^{-3})
	3	12	3.4681×10^{-1} (4.54×10^{-3}) =	3.4470×10^{-1} (3.49×10^{-3}) =	3.4464×10^{-1} (3.88×10^{-3})
ZDT1	2	30	3.8606×10^{-3} (4.58×10^{-5}) +	3.8777×10^{-3} (3.78×10^{-5}) =	3.8994×10^{-3} (4.98×10^{-5})
ZDT2	2	30	3.8505×10^{-3} (2.87×10^{-5}) +	3.8635×10^{-3} (2.90×10^{-5}) +	3.8933×10^{-3} (3.48×10^{-5})
ZDT3	2	30	4.6639×10^{-3} (5.03×10^{-5}) =	4.6779×10^{-3} (4.28×10^{-5}) =	4.6982×10^{-3} (7.19×10^{-5})
ZDT4	2	10	2.1828×10^1 (1.56×10^1) =	2.2784×10^1 (1.82×10^1) =	1.7501×10^1 (1.63×10^1)
+/-/=			7/14/15	4/13/19	

“+”, “=” and “-” indicate that the results of the CMOSCAA and CMOSCAD are statistically superior, similar, and inferior to results obtained by the CMOSCA, respectively. The best result of each test problem is displayed in bold.

Table 8. HV values achieved by three CMOSCA variants.

Problem	M	D	CMOSCAA	CMOSCAD	CMOSCA
DTLZ1	2	6	4.9801×10^{-4} (2.73×10^{-3}) =	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0)
	3	7	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0)
DTLZ2	2	11	3.4724×10^{-1} (9.67×10^{-5}) -	3.4750×10^{-1} (5.17×10^{-5}) +	3.4743×10^{-1} (5.68×10^{-5})
	3	12	5.5033×10^{-1} (2.21×10^{-3}) -	5.5478×10^{-1} (1.42×10^{-3}) +	5.5278×10^{-1} (1.94×10^{-3})
DTLZ3	2	11	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0)
	3	12	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0)
DTLZ4	2	11	3.4714×10^{-1} (7.74×10^{-5}) -	3.4743×10^{-1} (5.31×10^{-5}) +	3.4733×10^{-1} (7.32×10^{-5})
	3	12	5.4179×10^{-1} (3.26×10^{-3}) +	5.4538×10^{-1} (4.14×10^{-3}) +	5.3966×10^{-1} (3.31×10^{-3})
DTLZ5	2	11	3.4720×10^{-1} (8.66×10^{-5}) -	3.4750×10^{-1} (4.72×10^{-5}) +	3.4744×10^{-1} (6.74×10^{-5})
	3	12	1.9911×10^{-1} (3.17×10^{-4}) -	1.9987×10^{-1} (1.15×10^{-4}) +	1.9972×10^{-1} (1.87×10^{-4})
DTLZ6	2	11	3.4753×10^{-1} (4.55×10^{-5}) =	3.4752×10^{-1} (4.51×10^{-5}) =	3.4753×10^{-1} (4.45×10^{-5})
	3	12	2.0018×10^{-1} (4.58×10^{-5}) =	2.0016×10^{-1} (2.65×10^{-5}) =	2.0017×10^{-1} (3.98×10^{-5})
DTLZ7	2	21	2.4295×10^{-1} (9.49×10^{-6}) +	2.4294×10^{-1} (1.57×10^{-5}) =	2.4294×10^{-1} (1.03×10^{-5})
	3	22	2.7884×10^{-1} (7.48×10^{-4}) =	2.7891×10^{-1} (7.43×10^{-4}) =	2.7871×10^{-1} (5.17×10^{-4})
WFG1	2	11	5.1069×10^{-1} (4.98×10^{-2}) =	3.9481×10^{-1} (6.46×10^{-2}) -	5.0330×10^{-1} (5.46×10^{-2})
	3	12	4.6057×10^{-1} (5.11×10^{-2}) -	3.6556×10^{-1} (5.26×10^{-2}) -	5.6834×10^{-1} (5.89×10^{-2})
WFG2	2	11	6.1075×10^{-1} (3.76×10^{-2}) =	5.9275×10^{-1} (4.40×10^{-2}) -	6.1720×10^{-1} (3.33×10^{-2})
	3	12	9.1862×10^{-1} (3.17×10^{-2}) +	8.8812×10^{-1} (5.24×10^{-2}) -	9.1668×10^{-1} (3.63×10^{-2})
WFG3	2	11	5.7178×10^{-1} (2.69×10^{-2}) -	5.6606×10^{-1} (3.36×10^{-2}) -	5.7214×10^{-1} (2.70×10^{-2})
	3	12	3.5253×10^{-1} (1.39×10^{-2}) -	3.5365×10^{-1} (3.06×10^{-2}) -	3.6348×10^{-1} (3.13×10^{-2})
WFG4	2	11	3.3711×10^{-1} (4.72×10^{-3}) -	3.3624×10^{-1} (5.95×10^{-3}) -	3.3972×10^{-1} (3.23×10^{-3})
	3	12	4.9800×10^{-1} (4.03×10^{-3}) =	5.0639×10^{-1} (8.82×10^{-3}) =	5.0911×10^{-1} (8.68×10^{-3})
WFG5	2	11	3.1259×10^{-1} (1.39×10^{-3}) -	3.1344×10^{-1} (9.88×10^{-4}) =	3.1363×10^{-1} (1.39×10^{-4})
	3	12	5.0278×10^{-1} (4.50×10^{-3}) -	5.0607×10^{-1} (4.39×10^{-3}) =	5.0496×10^{-1} (3.75×10^{-3})
WFG6	2	11	2.2696×10^{-1} (5.03×10^{-5}) =	2.2697×10^{-1} (6.21×10^{-5}) =	2.2697×10^{-1} (5.36×10^{-5})
	3	12	4.1606×10^{-1} (3.21×10^{-3}) =	4.1608×10^{-1} (2.25×10^{-3}) =	4.1628×10^{-1} (8.87×10^{-4})
WFG7	2	11	3.4660×10^{-1} (1.39×10^{-4}) -	3.4695×10^{-1} (1.13×10^{-4}) -	3.4719×10^{-1} (1.15×10^{-4})
	3	12	5.4076×10^{-1} (4.01×10^{-3}) -	5.4761×10^{-1} (3.09×10^{-3}) -	5.5302×10^{-1} (2.17×10^{-3})
WFG8	2	11	2.8598×10^{-1} (1.86×10^{-3}) +	2.8431×10^{-1} (2.18×10^{-3}) =	2.8525×10^{-1} (1.69×10^{-3})
	3	12	4.3333×10^{-1} (6.61×10^{-3}) -	4.3261×10^{-1} (7.31×10^{-3}) -	4.4087×10^{-1} (6.67×10^{-3})
WFG9	2	11	2.2865×10^{-1} (1.23×10^{-2}) +	2.2982×10^{-1} (1.74×10^{-2}) =	2.2643×10^{-1} (1.04×10^{-3})
	3	12	4.0447×10^{-1} (2.41×10^{-3}) -	4.0602×10^{-1} (2.33×10^{-3}) =	4.0623×10^{-1} (2.68×10^{-3})
ZDT1	2	30	7.2055×10^{-1} (3.46×10^{-5}) +	7.2051×10^{-1} (4.45×10^{-5}) =	7.2049×10^{-1} (5.25×10^{-5})
ZDT2	2	30	4.4511×10^{-1} (2.69×10^{-5}) =	4.4508×10^{-1} (3.60×10^{-5}) -	4.4510×10^{-1} (4.02×10^{-5})
ZDT3	2	30	5.9980×10^{-1} (2.09×10^{-5}) =	5.9980×10^{-1} (1.98×10^{-5}) =	5.9979×10^{-1} (2.39×10^{-5})
ZDT4	2	10	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0) =	0.0000×10^0 (0.00×10^0)
+/-/=			6/16/14	6/11/19	

“+”, “=” and “-” indicate that the results of the CMOSCAA and CMOSCAD are statistically superior, similar, and inferior to results obtained by the CMOSCA, respectively. The best result of each test problem is displayed in bold.

4.4. Applying Our Proposed CMOSCA to Engineering Design Problems

In this part, to further evaluate the performance of the CMOSCA in handling the real-world problems, we apply the CMOSCA to deal with the engineering design problems including four bar truss design [73], hatch cover design [74], two bar truss design [75], welded beam design [76] and vehicle crashworthiness design [77]. Four bar truss design attempts to minimize the two objective functions (i.e., structural volume and joint displacement) and is subject to the member stresses on the four-bar truss. Hatch cover design is considered as a two-objective problem to minimize the weight of the cover and the constraint violation values [78]. Two bar truss design aims to minimize the two objective functions (i.e., structural volume and joint displacement) and is subject to the member stresses and design variables. Welded beam design is considered as a three-objective problem to minimize the cost, end deflection and the constraint violation values. Vehicle crashworthiness design aims to minimize the three objective functions (i.e., the mass of the vehicle, the integration of collision acceleration and the toe board intrusion). More information on the engineering design problems can be found in the references mentioned above. The proposed CMOSCA and the other six MOEAs are compared on these engineering design problems.

Table 9 provides the statistical HV results obtained by the seven competing MOEAs on these five engineering design problems. From the Table 9, it can be seen that the proposed CMOSCA, MOEA/D-DE and EMOSO obtain the highest HV results on 2, 2, 1 out of 5 engineering design problems, respectively. The other four MOEAs do not have the highest HV results. The proposed CMOSCA performs best on problems four bar truss design and vehicle crashworthiness design. The EMOSO performs best on the hatch cover design problem. The MOEA/D-DE performs best on problems two bar truss design and welded beam design. From the Wilcoxon rank sum test results in the Table 9, it can also be observed that the proposed CMOSCA outperforms the other six MOEAs. Specifically, we can see that CMOSCA outperforms the EMOSO, CMOPSO, MOEA/D, NSGA-II, MOEA/D-DE, and MMOPSO on 2, 2, 3, 4, 3, and 4 out of 5 test problems, respectively, while it loses on 0, 0, 1, 1, 1, and 1 out of 5 test problems. In conclusion, our proposed CMOSCA is a competitive MOEA for solving these engineering design problems.

Table 9. HV results obtained by the seven competing MOEAs on engineering design problems.

Problem	M	D	EMOSO	CMOPSO	MOEAD	CMOSCA
Four bar truss design	2	4	5.9810 × 10 ⁻¹ (1.24 × 10 ⁻⁹) -	5.9810 × 10 ⁻¹ (1.09 × 10 ⁻⁹) -	5.9810 × 10 ⁻¹ (8.25 × 10 ⁻¹⁰) -	5.9810 × 10⁻¹ (9.02 × 10 ⁻¹⁰)
Hatch cover design	2	2	8.8750 × 10⁻¹ (5.76 × 10 ⁻⁶) =	8.8750 × 10 ⁻¹ (5.49 × 10 ⁻⁶) =	8.7980 × 10 ⁻¹ (5.37 × 10 ⁻⁵) -	8.8750 × 10 ⁻¹ (5.81 × 10 ⁻⁶)
Two bar truss design	3	3	9.8210 × 10 ⁻¹ (6.34 × 10 ⁻³) =	9.8187 × 10 ⁻¹ (8.00 × 10 ⁻³) =	9.8748 × 10 ⁻¹ (2.99 × 10 ⁻⁶) =	9.8546 × 10 ⁻¹ (7.64 × 10 ⁻³)
Welded beam design	3	4	7.8936 × 10 ⁻¹ (4.04 × 10 ⁻¹) =	7.5242 × 10 ⁻¹ (4.25 × 10 ⁻¹) =	9.9990 × 10 ⁻¹ (1.55 × 10 ⁻⁵) +	6.0661 × 10 ⁻¹ (4.60 × 10 ⁻¹)
Vehicle crashworthiness design	3	5	2.4398 × 10 ⁻¹ (3.03 × 10 ⁻⁵) -	2.4397 × 10 ⁻¹ (3.54 × 10 ⁻⁵) -	2.4390 × 10 ⁻¹ (4.64 × 10 ⁻⁸) -	2.4399 × 10⁻¹ (1.53 × 10 ⁻⁶)
+/-/=			0/2/3	0/2/3	1/3/1	
Problem	M	D	NSGAI	MOEADDE	MMOPSO	CMOSCA
Four bar truss design	2	4	5.9810 × 10 ⁻¹ (2.61 × 10 ⁻⁹) -	5.9810 × 10 ⁻¹ (1.06 × 10 ⁻⁷) -	5.9810 × 10 ⁻¹ (2.68 × 10 ⁻⁹) -	5.9810 × 10⁻¹ (9.02 × 10 ⁻¹⁰)
Hatch cover design	2	2	8.8747 × 10 ⁻¹ (1.28 × 10 ⁻⁵) -	8.8665 × 10 ⁻¹ (4.11 × 10 ⁻⁶) -	8.8746 × 10 ⁻¹ (1.98 × 10 ⁻⁵) -	8.8750 × 10 ⁻¹ (5.81 × 10 ⁻⁶)
Two bar truss design	3	3	9.7451 × 10 ⁻¹ (1.50 × 10 ⁻²) -	9.9176 × 10⁻¹ (6.66 × 10 ⁻⁵) =	9.7852 × 10 ⁻¹ (1.02 × 10 ⁻²) -	9.8546 × 10 ⁻¹ (7.64 × 10 ⁻³)
Welded beam design	3	4	8.9927 × 10 ⁻¹ (3.05 × 10 ⁻¹) +	9.9991 × 10⁻¹ (2.13 × 10 ⁻⁵) +	9.9874 × 10 ⁻¹ (9.26 × 10 ⁻⁴) +	6.0661 × 10 ⁻¹ (4.60 × 10 ⁻¹)
Vehicle crashworthiness design	3	5	2.4399 × 10 ⁻¹ (4.79 × 10 ⁻⁷) -	2.4399 × 10 ⁻¹ (1.60 × 10 ⁻⁵) -	2.4395 × 10 ⁻¹ (4.47 × 10 ⁻⁵) -	2.4399 × 10⁻¹ (1.53 × 10 ⁻⁶)
+/-/=			1/4/0	1/3/1	1/4/0	

“+”, “=” and “-” indicate that the results of competing MOEAs are statistically superior, similar, and inferior to results obtained by the CMOSCA, respectively. The best result of each test problem is displayed in bold.

However, the following aspects need to be considered when using the proposed COMSCA to solve other real-world problems. First, we need to clarify the objectives of the problem and determine the decision variables and constraints involved. Then make appropriate adjustments and improvements to the proposed COMSCA according to the specific conditions of the problem, such as the individual coding method of the algorithm, to meet the needs of the real-world problems.

5. Conclusions

A novel multi-objective sine cosine algorithm based on a competitive mechanism (CMOSCA) was presented in this study. In the CMOSCA, a SDE-based competitive mechanism is utilized to devise a new position updating operation for creating offspring agents, which can make use of the SDE to achieve a good balance between the diversity and convergence. Moreover, we also proposed two CMOSCA variants, which adopt the angle-based competitive mechanism and Euclidean distance-based competitive mechanism, respectively. Six representative MOEAs were adopted to verify the performance of the proposed CMOSCA on twenty test problems with different characteristics. We experimentally analyzed the influences of the parameter β and compare the CMOSCA with its two variants. The experimental results exhibited the effectiveness and robustness of our proposed CMOSCA. Finally, several engineering design problems were employed to further evaluate the CMOSCA and results of the engineering problems also demonstrated the competitiveness of our proposed CMOSCA.

However, our proposed CMOSCA algorithm has some shortcomings. First, it does not perform well on some problems with multi-modal PF, such as DTLZ3, because it is still possible to fall into a local optimum. Second, the time complexity of the proposed CMOSCA is not the best among the competing MOEAs. Third, the performance of the proposed CMOSCA may degrade as the number of objectives and decision variables in multi-objective problems increases.

There are many more complex and challenging problems in multi-objective problems, such as variables that are stochastic, uncertain, dynamic, large-scale and constrained, which are interesting and challenging research problems. In our future work, we will further extend our proposed CMOSCA with new strategies [79,80] to solve more complex MOPs, large-scale MOPs, and dynamic MOPs. In addition, we will try to apply our improved algorithms to solve more other real-world problems.

Author Contributions: Conceptualization, N.L., G.L. and J.-S.P.; Investigation, M.F., P.H. and Y.K.; Methodology, N.L., Y.K., G.L., P.H. and J.-S.P.; Software, N.L.; funding acquisition, Y.K.; Writing—original draft, N.L.; Writing—review and editing, N.L., Y.K., G.L., J.-S.P. and M.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Zhejiang Provincial Natural Science Foundation of China grant number LY21G020009 and by the Fujian Provincial Natural Science Foundation of China grant number 2022J01117.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: The data presented in this study are available through email upon request to the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

Abbreviation	Definition
CMOPSO	competitive mechanism based multi-objective particle swarm optimizer
CMOSCA	multi-objective sine cosine algorithm based on the competitive mechanism
CMOSCAA	multi-objective sine cosine algorithm based on the angle competitive mechanism
CMOSCAD	multi-objective sine cosine algorithm based on the Euclidean distance competitive mechanism
CSO	competitive swarm optimizer
DE	differential evolution
DTLZ	scalable test problems for evolutionary multiobjective optimization, DTLZ are abbreviations of the reference authors
EA-M2SCA	energy-aware multi-objective modified sine cosine algorithm
EA-MHSCA	hybrid improved version multi-objective modified sine cosine algorithm
EED	environmental/economic dispatch
EELD	economic emission load dispatch
EMOSO	efficient multi-objective optimization algorithm based on level swarm optimizer
FES	current number of function evaluations
HV	hypervolume
IBEA	indicator-based evolutionary algorithm
MaOSCA	many-objective sine cosine algorithm
MaxFES	maximum number of function evaluations
MMOPSO	multi-objective particle swarm optimization with multiple search strategies
MOEA/D	multiobjective evolutionary algorithm based on decomposition
MOEA/D-AM2M	a new variant of MOEA/D-M2M with adaptive adjustment
MOEA/DD	evolutionary many-objective optimization algorithm based on dominance and decomposition
MOEA/D-DE	MOEA/D with differential evolution operators
MOEA/D-M2M	decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems
MOEA/D-PaS	decomposition-based algorithms using pareto adaptive scalarizing methods
MOEAs	multi-objective evolutionary algorithms
MOPs	multi-objective optimization problems
MOSCA	multi-objective sine cosine algorithm
MOSCA-SSC	multi-objective sine cosine algorithm for spatial-spectral clustering
MPS	multiprocessor systems
MSCA	modified sine cosine algorithm
MSCO	multi-objective sine cosine optimization algorithm
NSGA-II	fast and elitist multiobjective genetic algorithm
NSGA-III	evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach
PF	Pareto front
PlatEMO	platform for evolutionary multi-objective optimization
PS	Pareto-optimal set
PSO	particle swarm optimization
RVEA	reference vector guided evolutionary algorithm
SCA	sine cosine algorithm
SDE	shift-based density estimation
SMS-EMOA	s-metric selection based MOEA
TC-MOPSO	multiobjective particle swarm optimization algorithm based on tripartite competition mechanism
Two-arch2	improved two-archive algorithm for many-objective optimization
VEGA	vector evaluated genetic algorithm
WFG	scalable multi-objective test problem toolkit, WFG is abbreviation of Walking Fish Group
ZDT	multi-objective test problem, ZDT are abbreviations of the reference authors

References

1. Luo, Q.; Yin, S.; Zhou, G.; Meng, W.; Zhao, Y.; Zhou, Y. Multi-objective equilibrium optimizer slime mould algorithm and its application in solving engineering problems. *Struct. Multidiscip. Optim.* **2023**, *66*, 1–41. [[CrossRef](#)]
2. Zhang, L.; Pan, H.; Su, Y.; Zhang, X.; Niu, Y. A Mixed Representation-Based Multiobjective Evolutionary Algorithm for Overlapping Community Detection. *IEEE Trans. Cybern.* **2017**, *47*, 2703–2716. [[CrossRef](#)]
3. Deb, S.; Tammi, K.; Gao, X.Z.; Kalita, K.; Mahanta, P. A Hybrid Multi-Objective Chicken Swarm Optimization and Teaching Learning Based Algorithm for Charging Station Placement Problem. *IEEE Access* **2020**, *8*, 92573–92590. [[CrossRef](#)]
4. Bagherzadeh, S.A.; Asadi, D. Detection of the ice assertion on aircraft using empirical mode decomposition enhanced by multi-objective optimization. *Mech. Syst. Signal Process.* **2017**, *88*, 9–24. [[CrossRef](#)]
5. Ponsich, A.; Jaimes, A.L.; Coello, C.A. A survey on multiobjective evolutionary algorithms for the solution of the portfolio optimization problem and other finance and economics applications. *IEEE Trans. Evol. Comput.* **2013**, *17*, 321–344. [[CrossRef](#)]
6. Tian, Y.; Yang, S.; Zhang, X.; Jin, Y. Using PlatEMO to Solve Multi-Objective Optimization Problems in Applications: A Case Study on Feature Selection. In Proceedings of the 2019 IEEE Congress on Evolutionary Computation, CEC 2019—Proceedings, Wellington, New Zealand, 10–13 June 2019; pp. 1710–1717.
7. Xue, Y.; Tang, Y.; Xu, X.; Liang, J.; Neri, F. Multi-Objective Feature Selection with Missing Data in Classification. *IEEE Trans. Emerg. Top. Comput. Intell.* **2022**, *6*, 355–364. [[CrossRef](#)]
8. Xue, Y.; Cai, X.; Neri, F. A multi-objective evolutionary algorithm with interval based initialization and self-adaptive crossover operator for large-scale feature selection in classification. *Appl. Soft Comput.* **2022**, *127*, 109420. [[CrossRef](#)]
9. Schaffer, J. Multiple objective optimization with vector evaluated genetic algorithms. In Proceedings of the 1st international Conference on Genetic Algorithms, Pittsburgh, PA, USA, 24–26 July 1985; pp. 93–100.
10. Zeng, N.; Song, D.; Li, H.; You, Y.; Liu, Y.; Alsaadi, F.E. A competitive mechanism integrated multi-objective whale optimization algorithm with differential evolution. *Neurocomputing* **2021**, *432*, 170–182. [[CrossRef](#)]
11. Qin, S.; Sun, C.; Zhang, G.; He, X.; Tan, Y. A modified particle swarm optimization based on decomposition with different ideal points for many-objective optimization problems. *Complex Intell. Syst.* **2020**, *6*, 263–274. [[CrossRef](#)]
12. Yang, Y.; Zhang, C. A Multi-Objective Carnivorous Plant Algorithm for Solving Constrained Multi-Objective Optimization Problems. *Biomimetics* **2023**, *8*, 136. [[CrossRef](#)]
13. Pan, J.S.; Liu, N.; Chu, S.C. A competitive mechanism based multi-objective differential evolution algorithm and its application in feature selection. *Knowl.-Based Syst.* **2022**, *245*, 108582. [[CrossRef](#)]
14. Meng, Z.; Chen, Y. Differential Evolution with exponential crossover can be also competitive on numerical optimization. *Appl. Soft Comput.* **2023**, *146*, 110750. [[CrossRef](#)]
15. Gonzalez-Sanchez, B.; Vega-Rodríguez, M.A.; Santander-Jiménez, S. A multi-objective butterfly optimization algorithm for protein encoding. *Appl. Soft Comput.* **2023**, *139*, 110269. [[CrossRef](#)]
16. Long, H.; Chen, Z.; Huang, H.; Yu, L.; Li, Z.; Liu, J.; Long, Y. Research on Multi-Objective Optimization Power Flow of Power System Based on Improved Remora Optimization Algorithm. *Eng. Lett.* **2023**, *31*, 1191–1207.
17. Jia, H.; Peng, X.; Lang, C. Remora optimization algorithm. *Expert Syst. Appl.* **2021**, *185*, 115665. [[CrossRef](#)]
18. Jia, H.; Rao, H.; Wen, C.; Mirjalili, S. Crayfish optimization algorithm. *Artif. Intell. Rev.* **2023**, *56*, 1919–1979. [[CrossRef](#)]
19. Yue, L.; Hu, P.; Chu, S.C.; Pan, J.S. Multi-Objective Gray Wolf Optimizer with Cost-Sensitive Feature Selection for Predicting Students' Academic Performance in College English. *Mathematics* **2023**, *11*, 3396. [[CrossRef](#)]
20. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197. [[CrossRef](#)]
21. Yuan, Y.; Xu, H.; Wang, B.; Yao, X. A New Dominance Relation-Based Evolutionary Algorithm for Many-Objective Optimization. *IEEE Trans. Evol. Comput.* **2016**, *20*, 16–37. [[CrossRef](#)]
22. Yang, S.; Li, M.; Liu, X.; Zheng, J. A grid-based evolutionary algorithm for many-objective optimization. *IEEE Trans. Evol. Comput.* **2013**, *17*, 721–736. [[CrossRef](#)]
23. Wang, G.; Jiang, H. Fuzzy-dominance and its application in evolutionary many objective optimization. In Proceedings of the Proceedings—CIS Workshops 2007, 2007 International Conference on Computational Intelligence and Security Workshops, Harbin, China, 15–19 December 2007; pp. 195–198.
24. Laumanns, M.; Thiele, L.; Deb, K.; Zitzler, E. Combining convergence and diversity in evolutionary multiobjective optimization. *Evol. Comput.* **2002**, *10*, 263–282. [[CrossRef](#)]
25. Cheng, R.; Jin, Y. A competitive swarm optimizer for large scale optimization. *IEEE Trans. Cybern.* **2015**, *45*, 191–204. [[CrossRef](#)]
26. Zhang, X.; Zheng, X.; Cheng, R.; Qiu, J.; Jin, Y. A competitive mechanism based multi-objective particle swarm optimizer with fast convergence. *Inf. Sci.* **2018**, *427*, 63–76. [[CrossRef](#)]
27. Han, F.; Zheng, M.; Ling, Q. An improved multiobjective particle swarm optimization algorithm based on tripartite competition mechanism. *Appl. Intell.* **2022**, *52*, 5784–5816. [[CrossRef](#)]
28. Zhang, X.W.; Liu, H.; Tu, L.P.; Zhao, J. An efficient multi-objective optimization algorithm based on level swarm optimizer. *Math. Comput. Simul.* **2020**, *177*, 588–602. [[CrossRef](#)]
29. Zhang, Q.; Li, H. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Trans. Evol. Comput.* **2007**, *11*, 712–731. [[CrossRef](#)]

30. Han, J.; Watanabe, S. A New Hyper-Heuristic Multi-Objective Optimisation Approach Based on MOEA/D Framework. *Biomimetics* **2023**, *8*, 521. [[CrossRef](#)] [[PubMed](#)]
31. Liu, H.L.; Gu, F.; Zhang, Q. Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems. *IEEE Trans. Evol. Comput.* **2014**, *18*, 450–455. [[CrossRef](#)]
32. Li, H.; Zhang, Q. Multiobjective optimization problems with complicated pareto sets, MOEA/ D and NSGA-II. *IEEE Trans. Evol. Comput.* **2009**, *13*, 284–302. [[CrossRef](#)]
33. Wang, R.; Zhang, Q.; Zhang, T. Decomposition-based algorithms using pareto adaptive scalarizing methods. *IEEE Trans. Evol. Comput.* **2016**, *20*, 821–837. [[CrossRef](#)]
34. Liu, H.L.; Chen, L.; Zhang, Q.; Deb, K. Adaptively Allocating Search Effort in Challenging Many-Objective Optimization Problems. *IEEE Trans. Evol. Comput.* **2018**, *22*, 433–448. [[CrossRef](#)]
35. Cheng, R.; Jin, Y.; Olhofer, M.; Sendhoff, B. A Reference Vector Guided Evolutionary Algorithm for Many-Objective Optimization. *IEEE Trans. Evol. Comput.* **2016**, *20*, 773–791. [[CrossRef](#)]
36. Zhao, Y.; Sun, C.; Zeng, J.; Tan, Y.; Zhang, G. A surrogate-ensemble assisted expensive many-objective optimization. *Knowl.-Based Syst.* **2021**, *211*, 106520. [[CrossRef](#)]
37. Yang, C.; Wang, P.; Ji, J. A dual decomposition strategy for large-scale multiobjective evolutionary optimization. *Neural Comput. Appl.* **2023**, *35*, 3767–3788. [[CrossRef](#)]
38. Beume, N.; Naujoks, B.; Emmerich, M. SMS-EMOA: Multiobjective selection based on dominated hypervolume. *Eur. J. Oper. Res.* **2007**, *181*, 1653–1669. [[CrossRef](#)]
39. Zitzler, E.; Künzli, S. Indicator-based selection in multiobjective search. *Lect. Notes Comput. Sci.* **2004**, *3242*, 832–842.
40. Bader, J.; Zitzler, E. HypE: An algorithm for fast hypervolume-based many-objective optimization. *Evol. Comput.* **2011**, *19*, 45–76. [[CrossRef](#)]
41. Tian, Y.; Cheng, R.; Zhang, X.; Cheng, F.; Jin, Y. An Indicator-Based Multiobjective Evolutionary Algorithm with Reference Point Adaptation for Better Versatility. *IEEE Trans. Evol. Comput.* **2018**, *22*, 609–622. [[CrossRef](#)]
42. Li, K.; Deb, K.; Zhang, Q.; Kwong, S. An evolutionary many-objective optimization algorithm based on dominance and decomposition. *IEEE Trans. Evol. Comput.* **2015**, *19*, 694–716. [[CrossRef](#)]
43. Deb, K.; Jain, H. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, Part I: Solving problems with box constraints. *IEEE Trans. Evol. Comput.* **2014**, *18*, 577–601. [[CrossRef](#)]
44. Wang, H.; Jiao, L.; Yao, X. Two Arch2: An Improved Two-Archive Algorithm for Many-Objective Optimization. *IEEE Trans. Evol. Comput.* **2015**, *19*, 524–541. [[CrossRef](#)]
45. Liu, N.; Pan, J.S.; Chu, S.C. A Competitive Learning QUasi Affine TRansformation Evolutionary for Global Optimization and Its Application in CVRP. *J. Internet Technol.* **2020**, *21*, 1863–1883.
46. Tian, Y.; Zheng, X.; Zhang, X.; Jin, Y. Efficient Large-Scale Multiobjective Optimization Based on a Competitive Swarm Optimizer. *IEEE Trans. Cybern.* **2020**, *50*, 3696–3708. [[CrossRef](#)]
47. Mirjalili, S. SCA: A Sine Cosine Algorithm for solving optimization problems. *Knowl.-Based Syst.* **2016**, *96*, 120–133. [[CrossRef](#)]
48. Rizk-Allah, R.M.; Hassanien, A.E. A comprehensive survey on the sine-cosine optimization algorithm. *Artif. Intell. Rev.* **2023**, *56*, 4801–4858. [[CrossRef](#)]
49. Rizk-Allah, R.M.; Mageed, H.M.; El-Sehiemy, R.A.; Aleem, S.H.; El Shahat, A. A new sine cosine optimization algorithm for solving combined non-convex economic and emission power dispatch problems. *Int. J. Energy Convers.* **2017**, *5*, 180–192. [[CrossRef](#)]
50. Tawhid, M.A.; Savsani, V. Multi-objective sine-cosine algorithm (MO-SCA) for multi-objective engineering design problems. *Neural Comput. Appl.* **2019**, *31*, 915–929. [[CrossRef](#)]
51. Wan, Y.; Zhong, Y.; Ma, A.; Zhang, L. Hyperspectral Remote Sensing Image Band Selection Via Multi-Objective Sine Cosine Algorithm. In Proceedings of the IGARSS 2019—2019 IEEE International Geoscience and Remote Sensing Symposium, Yokohama, Japan, 28 July–2 August 2019; pp. 3796–3799.
52. Wan, Y.; Ma, A.; Zhang, L.; Zhong, Y. Multiobjective Sine Cosine Algorithm for Remote Sensing Image Spatial-Spectral Clustering. *IEEE Trans. Cybern.* **2022**, *52*, 11172–11186. [[CrossRef](#)]
53. Abdel-Basset, M.; Mohamed, R.; Abouhawwash, M.; Chakraborty, R.K.; Ryan, M.J. EA-MSCA: An effective energy-aware multi-objective modified sine-cosine algorithm for real-time task scheduling in multiprocessor systems: Methods and analysis. *Expert Syst. Appl.* **2021**, *173*, 114699. [[CrossRef](#)]
54. Wang, J.; Yang, W.; Du, P.; Niu, T. A novel hybrid forecasting system of wind speed based on a newly developed multi-objective sine cosine algorithm. *Energy Convers. Manag.* **2018**, *163*, 134–150. [[CrossRef](#)]
55. Selim, A.; Kamel, S.; Jurado, F. Optimal allocation of distribution static compensators using a developed multi-objective sine cosine approach. *Comput. Electr. Eng.* **2020**, *85*, 106671. [[CrossRef](#)]
56. Altay, E.V.; Alatas, B. Differential evolution and sine cosine algorithm based novel hybrid multi-objective approaches for numerical association rule mining. *Inf. Sci.* **2021**, *554*, 198–221. [[CrossRef](#)]
57. Raut, U.; Mishra, S. A new Pareto multi-objective sine cosine algorithm for performance enhancement of radial distribution network by optimal allocation of distributed generators. *Evol. Intell.* **2021**, *14*, 1635–1656. [[CrossRef](#)]
58. Narayanan, R.C.; Ganesh, N.; Čep, R.; Jangir, P.; Chohan, J.S.; Kalita, K. A Novel Many-Objective Sine-Cosine Algorithm (MaOSCA) for Engineering Applications. *Mathematics* **2023**, *11*, 2301. [[CrossRef](#)]

59. Karimulla, S.; Ravi, K. Solving multi objective power flow problem using enhanced sine cosine algorithm. *Ain Shams Eng. J.* **2021**, *12*, 3803–3817. [[CrossRef](#)]
60. Abualigah, L.; Diabat, A. Advances in Sine Cosine Algorithm: A comprehensive survey. *Artif. Intell. Rev.* **2021**, *54*, 2567–2608. [[CrossRef](#)]
61. Gabis, A.B.; Meraihi, Y.; Mirjalili, S.; Ramdane-Cherif, A. A comprehensive survey of sine cosine algorithm: Variants and applications. *Artif. Intell. Rev.* **2021**, *54*, 5469–5540. [[CrossRef](#)]
62. Schussler, N.; Axhausen, K.W. SPEA2: Improving the strength pareto evolutionary algorithm. *Transp. Res. Rec.* **2009**, *2105*, 28–36.
63. Li, M.; Yang, S.; Liu, X. Shift-based density estimation for pareto-based algorithms in many-objective optimization. *IEEE Trans. Evol. Comput.* **2014**, *18*, 348–365. [[CrossRef](#)]
64. Lin, Q.; Li, J.; Du, Z.; Chen, J.; Ming, Z. A novel multi-objective particle swarm optimization with multiple search strategies. *Eur. J. Oper. Res.* **2015**, *247*, 732–744. [[CrossRef](#)]
65. Tian, Y.; Cheng, R.; Zhang, X.; Jin, Y. PlatEMO: A MATLAB Platform for Evolutionary Multi-Objective Optimization. *IEEE Comput. Intell. Mag.* **2017**, *12*, 73–87. [[CrossRef](#)]
66. Deb, K.; Thiele, L.; Laumanns, M.; Zitzler, E. Scalable Test Problems for Evolutionary Multiobjective Optimization. In *Evolutionary Multiobjective Optimization*; Springer: London, UK, 2005; pp. 105–145.
67. Huband, S.; Barone, L.; While, L.; Hingston, P. A scalable multi-objective test problem toolkit. *Lect. Notes Comput. Sci.* **2005**, *3410*, 280–295.
68. Zitzler, E.; Deb, K.; Thiele, L. Comparison of multiobjective evolutionary algorithms: Empirical results. *Evol. Comput.* **2000**, *8*, 173–195. [[CrossRef](#)]
69. Sun, Y.; Yen, G.G.; Yi, Z. IGD Indicator-Based Evolutionary Algorithm for Many-Objective Optimization Problems. *IEEE Trans. Evol. Comput.* **2019**, *23*, 173–187. [[CrossRef](#)]
70. Xiang, Y.; Zhou, Y.; Li, M.; Chen, Z. A Vector Angle-Based Evolutionary Algorithm for Unconstrained Many-Objective Optimization. *IEEE Trans. Evol. Comput.* **2017**, *21*, 131–152. [[CrossRef](#)]
71. Coello, C.A.; Cortés, N.C. Solving multiobjective optimization problems using an artificial immune system. *Genet. Program. Evolvable Mach.* **2005**, *6*, 163–190. [[CrossRef](#)]
72. Zitzler, E.; Thiele, L. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Trans. Evol. Comput.* **1999**, *3*, 257–271. [[CrossRef](#)]
73. Cheng, F.Y.; Li, X.S. Generalized center method for multiobjective engineering optimization. *Eng. Optim.* **1999**, *31*, 641–661. [[CrossRef](#)]
74. Amir, H.M.; Hasegawa, T. Nonlinear Mixed-Discrete Structural Optimization. *J. Struct. Eng.* **1989**, *115*, 626–646. [[CrossRef](#)]
75. Coello Coello, C.A.; Pulido, G.T. Multiobjective structural optimization using a microgenetic algorithm. *Struct. Multidiscip. Optim.* **2005**, *30*, 388–403. [[CrossRef](#)]
76. Ray, T.; Liew, K.M. A swarm metaphor for multiobjective design optimization. *Eng. Optim.* **2002**, *34*, 141–153. [[CrossRef](#)]
77. Zhang, W.; Liao, X.; Zhong, Z. Multi-objective optimization for crash safety design of vehicles using stepwise regression model. *Jixie Gongcheng Xuebao/Chin. J. Mech. Eng.* **2007**, *43*, 142–147.
78. Tanabe, R.; Ishibuchi, H. An easy-to-use real-world multi-objective optimization problem suite. *Appl. Soft Comput. J.* **2020**, *89*, 106078. [[CrossRef](#)]
79. Jia, H.; Lu, C.; Xing, Z. Memory backtracking strategy: an evolutionary updating mechanism for meta-heuristic algorithms. *Swarm Evol. Comput.* **2023**, *84*, 101456. [[CrossRef](#)]
80. Jia, H.; Lu, C. Guided learning strategy: A novel update mechanism for metaheuristic algorithms design and improvement. *Knowl.-Based Syst.* **2024**, *286*, 111402. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.