Journal of Artificial Intelligence and Data Mining (JAIDM), Vol. 9, No. 4, 2021, 497-514.

Shahrood University of Technology

Journal of Artificial Intelligence and Data Mining (JAIDM)
Journal homepage: http://jad.shahroodut.ac.ir

Research paper

A Multi-objective Approach based on Competitive Optimization Algorithm and its Engineering Applications

Yousef Sharafi[1], Mohammad Teshnehlab[2*] and Mojtaba Ahmadieh Khanesar[3]

1. Department of Computer Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran.
2. Intelligent Systems Laboratory, Faculty of Electrical and Computer Engineering, K. N. Toosi University of Technology, Tehran, Iran.
3. Faculty of Engineering, University of Nottingham, Nottingham, UK.

Article Info

Article History:
Received 04 May 2021
Revised 09 July 2021
Accepted 10 August 2021

DOI:10.22044/JADM.2021.10698.2204

Keywords:
Multi-objective Optimization, Competitive Optimization Agorithm, Initial Population, Engineering Design Problems, Proposed Crowding Distance.

**Corresponding author: teshnehlab@eetd.kntu.ac.ir (M. Teshnehlab).*

Abstract

Herein a new multi-objective evolutionary optimization algorithm is presented based on the competitive optimization algorithm in order to solve the multi-objective optimization problems. Based on the nature-inspired competition, the competitive optimization algorithm acts between the animals such as birds, cats, bees and ants. The present work entails the main contributions as what follows. Primarily, a novel method is presented for pruning the external archive, while keeping the diversity of the Pareto front. Secondly, a hybrid approach of powerful mechanisms such as opposition-based learning and chaotic maps is was used in order to maintain the diversity in the search space of the initial population. Thirdly, a novel method is provided in order to transform a multi-objective optimization problem into a single-objective optimization problem. A Comparison of the results of the simulation for the proposed algorithm is performed with some well-known optimization algorithms. The comparison indicates that the proposed approach could be a better option for solving the multi-objective optimization problems.

1. Introduction

Since many of the existing scientific problems in different science and engineering fields are complex and involve the selection of various parameters, it is necessary to design novel optimization methods in order to deal with these problems. There are two categories of optimization problems: single-objective and multi-objective optimization problems. In the multi-objective optimization problems (MOPs), some parameters, such as the speed of the convergence, quality of the final solutions, and consumed time are highly crucial. Over the recent decades, many different single-objective algorithms have been modified in order to achieve the ideal conditions for solving a series of MOPs. The convergence and diversity are assumed as the major issues to which a great attention should be paid in order to improve or propose a new multi-objective optimization [10,15,29]. The first issue is how to move the population members toward the Pareto-optimal front (speed of convergence), and the second issue is how to maintain the diversity in the Pareto front (PF). Selecting the appropriate pruning mechanism in the archive of the algorithm and ranking the members has a crucial role in creating an appropriate diversity of solutions in the final PF curve in multi-objective evolutionary optimization algorithms (MOEAs). The NSGA-II algorithm uses the concept of crowding distance (CD) for pruning the external archive of the algorithm. Despite the simple CD and being easily used, its mechanism has substantial drawbacks [8,23]. In [8, 44], some considerable drawbacks on the idea of CD have been pointed out, demonstrating that sometimes the mechanism of the algorithm of CD will be laden with errors and lead to the illogical final results. The substantial drawback of CD will occur when the members of the population in the PF curve have similar cost function values, which cause CD performance to be significantly reduced [8].

In this paper, in order to replace the concept of the CD, a new method, called the new crowding distance (NCD) is proposed by relying on the weaknesses of CD. This concept guarantees the diversity maintenance in PF quite logically. This criterion is used in order to control and prune the size of the external archive of the proposed algorithm. Another important concept, which has attracted a great deal of attention over the recent years, is the production of an initial population of algorithm via two methods of chaotic maps and opposition-based learning (OBL) [19, 21], whose efficiency in some MOEAs has been recently presented [11, 24]. An external archive is a place to store non-dominated solutions in the proposed algorithm. This archive is updated at the end of each step of the evolutionary algorithm; the size of this archive is usually equal to the population size of the evolutionary algorithm. The reason for the existence of this archive is that the population of the evolutionary algorithm undergoes many changes, and there must be a place to store non-dominated solutions.

In this work, a hybrid approach is presented, and the advantages of these techniques are employed in order to create an appropriate initial population. The competitive optimization algorithm (COOA) [16] acts based on the bio-computing competition between the particle swarm optimization (PSO) [18], artificial bee colony (ABC) algorithm [9], cat swarm optimization (CSO) [4], ant colony optimization (ACO) [17], and imperialist competitive algorithm (ICA) [2]. In the initial version of the COOA algorithm, the CSO, PSO, ACO, and ABC algorithms are used but any other evolutionary algorithm could be used within COOA. There is no limit to the number of algorithms used within COOA, for example, the firefly algorithm [45] can also be used as a species within this biological competition. According to the no free lunch (NFL) optimization theorems, each algorithm could solve a series of optimization problems, and acts very well to solve a series of problems. On the other hand, the algorithm may fail to yield satisfactory results for the other problems [20]. Therefore, due to the fact that COOA has the strengths of all of these algorithms in the form of an optimization algorithm, it is of an efficient performance for solving different MOPs.

A fundamental concept in the single-objective version of COOA is the cost function value of the population members, which is used for the interaction and competition of various animals, according to which, the weakest member of a group among all groups is characterized. Herein, a new cost function value (*fitness*) for MOPs is proposed.

The research contributions are summarized as follow:

- A multi-objective approach based on the competitive optimization algorithm (MOCOOA) is proposed.
- A method is presented in order to prune the external archive and maintain the diversity of dispersion in the Pareto front.
- A method is provided in order to convert a multi-objective optimization problem to a single-objective optimization problem.
- The new hybrid approach for production of the initial population is proposed.
- The results of the constrained multi-objective engineering problems prove the applicability of MOCOOA in the real-life applications.
- The results obtained confirm the outperformance of MOCOOA over the other compared algorithms.

The rest of this paper is designed as what follows. Section 2 represents the concept of the multi-objective optimization problem, and a brief overview of MOEAs is discussed. In the third section, the general concept of the COOA algorithm is discussed. In the fourth section, the proposed method, entitled as the multi-objective competitive optimization algorithm (MOCOOA), is presented, and in the final section, the result of the simulation for the introduced algorithm is compared to with some well-known meta-heuristic algorithms in order to solve MOPs [22, 25, 29].

2. Literature Review

This section provides the concepts of MOPs and the current techniques in the multi-objective evolutionary algorithms.

2.1. Multi-objective Optimization Problem

The functions of an MOP with *m* objective in a space with n state variables is defined as follows [1]:

$$\textit{Minimize } F(\vec{x}) = (f_1(\vec{x}), f_2(\vec{x}), ..., f_m(\vec{x})) \qquad (1)$$

in which:

$$\vec{x} = (x_1, x_2, ..., x_n)$$
$$f_i : \mathrm{R}^n \to R,\ \forall \mathrm{i} \in (1,2,...,\mathrm{m}) \qquad (2)$$

The set of equality and inequality constraints, if there are any, is as follows:

$$\begin{aligned} g(\vec{x}) &= (g_1(\vec{x}), g_2(\vec{x}), ..., g_k(\vec{x})) \geq 0 \\ h(\vec{x}) &= (h_1(\vec{x}), h_2(\vec{x}), ..., h_p(\vec{x})) = 0 \end{aligned} \tag{3}$$

2.2. Multi-objective Evolutionary Algorithms (MOEAs)

The evolutionary optimization algorithms are of the most well-known meta-heuristic optimization approaches. Various studies have presented different versions of this algorithm for solving different MOPs. MOEAs are classified as the decomposition-based, indicator-based, and Pareto-based algorithms [15, 29].

Pareto-based: In this category, it was attempted to identify the non-dominated solutions of the population and maintain and update them in the optimization process. In these approaches, usually a small population of non-dominated solutions in the evolutionary algorithm search process along with the main population. Some examples of the recent Pareto-based approaches are NSGA-II [6, 8], multi-objective PSO (MOPSO) [5, 12, 35, 48], multi-objective CSO (MOCSO) [14], multi-objective ICA (MOICA) [7, 49], multi-objective ACO [1], and multi-objective grey wolf optimization (MOGWO) algorithm [13], multi-objective emperor penguin optimizer (MOEPO) [31], multi-objective seagull optimization algorithm (MOSOA) [38], and multi-objective hybrid particle swarm, and salp optimization algorithm [32]. The quality of the individuals in PF could improve using the performance indicators, which are used in the convergence and diversity of PF. Among these methods, the IGD indicator-based algorithm [3], hyper volume indicator-based algorithm [33, 34], ε-indicator-based algorithm [36], MOEA/IGD-NS [25], and LIBEA [26] could be mentioned. The details of the effects of the different indicators on the performance of the multi-objective optimization algorithms have been reported in [37].

Over the recent decade, various MOEAs have successfully employed the scalarizing functions. The decomposition method could be utilized in order to convert an MOP into the multiple single-objective optimization problems and solve them at the same time using an evolutionary algorithm. Among the advantages of the decomposition approach, the simplicity and the easy incorporation of the local search methods could be mentioned [27, 39, 40]. Examples of the swarm decomposition-based approaches are MOEA/D [11, 47], MOPSO/D [28], decomposition-based differential evolution (DDE/R) [42], and MOEA/D-IMA [41].

3. Competitive Optimization Algorithm (COOA)

COOA is based on the struggle for existence between the species, such as birds, cats, bees, and ants [16]. The main philosophy of providing COOA is that each evolutionary algorithm has the ability to solve some optimization problems. In fact, it could be declared that no algorithm has been proposed in order to solve all optimization problems to date. Hence, the researchers have offered several evolutionary algorithms inspired by the nature to solve some particular optimization problems over the recent decades. Since selecting a proper evolutionary algorithm to solve a particular optimization problem is highly time-consuming, a nature-inspired competition between species is proposed in COOA. The competition between species makes it possible to benefit from different algorithms at the same time. A more comprehensive review of COOA could be found in [16].

4. Proposed Algorithm

This section is composed of four sub-sections. In the first one, a new criterion is proposed in order to maintain the diversity in PF, and prune the excess members of the external archive. In the second sub-section, the production methods of the initial population for the appropriate and efficient start of the algorithm are studied, and a hybrid approach to maintain a proper diversity is proposed. In the third sub-section, a new approach is provided in order to calculate the cost function value (*fitness*) of an MOP. Finally, in the fourth sub-section, a global search strategy is provided in order to improve the position of the group members. Given that a number of evolutionary algorithms are used within MOCOOA, some of the main operators of MOCOOA are within these algorithms. In the MOPSO algorithm, for example, there are velocity and position update operators, as well as for the other algorithms. Irrespective of the internal algorithms, in MOCOOA, there are initial population production operator, global search operator, pruning operator for non-nominated solutions, control of diversity operator in the Pareto front, and the converting multi-objective space into single-objective space operator.

Figure 1 represents the COOA algorithm processes.

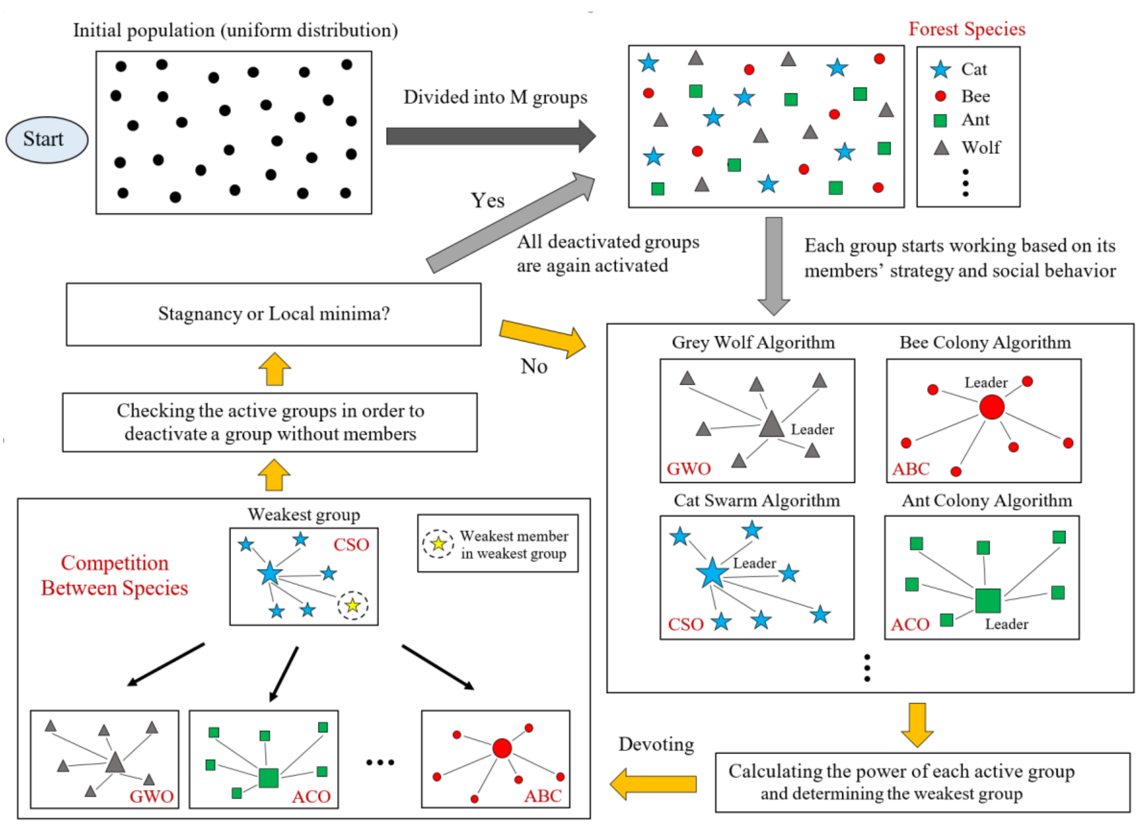


Figure 1. Overall procedure of COOA.

4.1. Concept of New Crowding Distance (NCD)

One of the substantial drawbacks in the concept of the crowding distance appears once the values of the objective functions of some members of the population are similar [8, 23], as shown in Figure 2(a).

In principle, this concept will work well when the members of a PF do not resemble in terms of the values of the objective functions. As depicted in Figure 2(b), on account of the similarity of members 'd' and 'e', none of them are selected; therefore, the diversity throughout the curve of PF is not guaranteed.

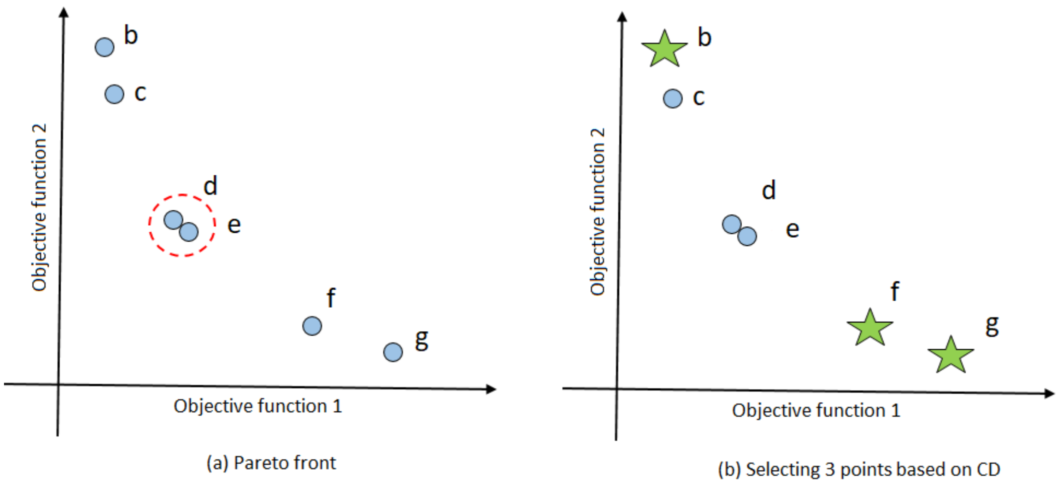


Figure 2. Selection of a number of solutions from the Pareto front based on crowding distance.

In order to solve this problem, a new concept of CD, known as the NCD, is defined (Figure 3). The proposed method requires the values of the objective function for the first and last member of PF in order to obtain the NCD value of a member (see Equation (4)). Despite the traditional CD method, the values of objective functions for the preceding and succeeding members are not required in this concept calculation.

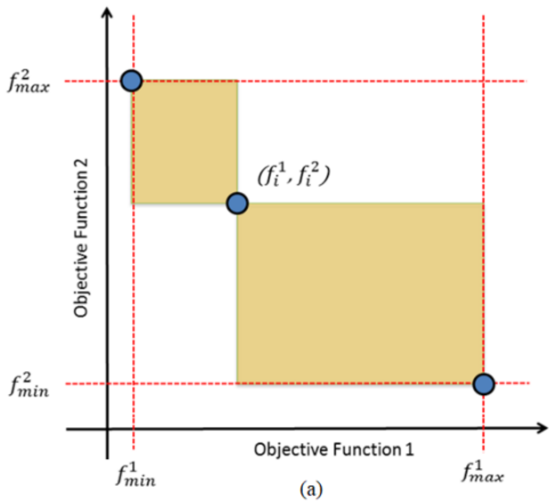


Figure 3. Offering a new method to control the diversity of PF.

In order to show the assurance of the diversity in PF in the proposed method, five members of the PF curve are selected in an example. Figure 4 demonstrates the order of choice of solutions with numbers 1 to 5. In the NCD method, the initial and final Pareto front solutions will always be selected (solutions 1 and 2). In order to select the third solution in the NCD method, solutions 1 and 2 are the beginning and end of the Pareto front. Similarly, in order to select the fourth solution, solutions 1 and 3 are the beginning and end of the Pareto front (Figure 4(c)).

$$NCD_i = \frac{(f_i^1 - f_{\min}^1)(f_{\max}^2 - f_i^2) + (f_i^2 - f_{\min}^2)(f_{\max}^1 - f_i^1)}{(f_{\max}^1 - f_{\min}^1)(f_{\max}^2 - f_{\min}^2)} \tag{4}$$

Figures 4(b) and 4(c) illustrate the result of the implementation of the traditional CD and the new proposed method (NCD) on the members of PF in Figure 4(a). The assurance of the diversity of the selected members of PF in the proposed method is visible compared to the idea of the traditional CD. The best five members of PF based on the conventional CD method and the proposed concept (NCD) are selected in Figures 4(b) and 4(c), respectively. The blue points in Figure 4 are the surplus members that are to be removed. As clearly demonstrated in Figure 4(b), no favorable distribution of the solutions exists in the entire PF, and in some parts, no members are selected at all.

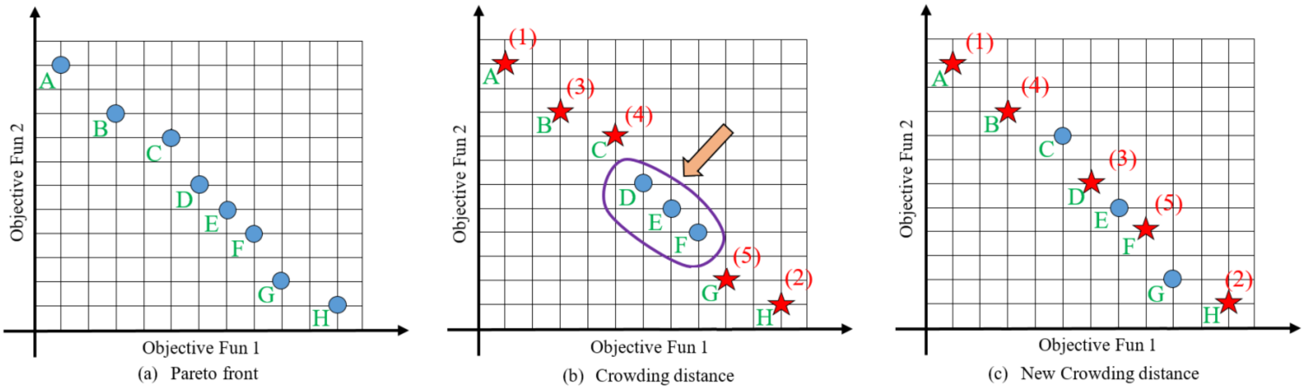


Figure 4. Comparison between method of conventional CD and the proposed NCD.

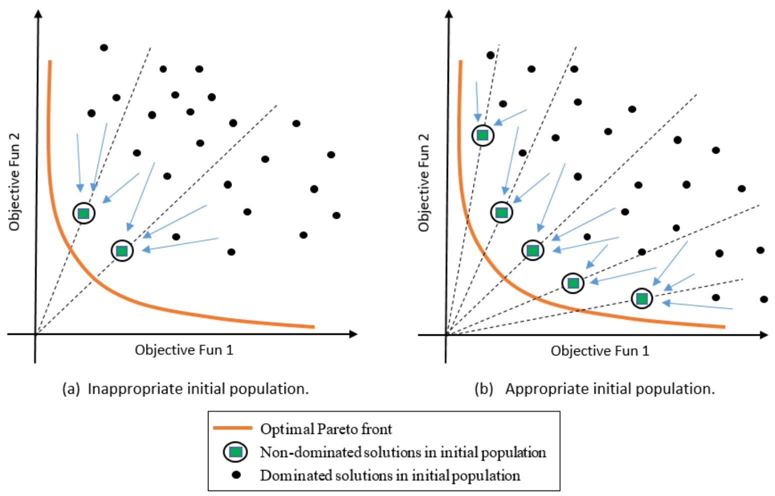


Figure 5. Impact of an appropriate initial population on speed of convergence.

4.2. A hybrid Initial Population Strategy

One of the most essential parts of soft computing is the initialization of an algorithm. Creating an appropriate initial population across the search space of the problem has a high impact on the algorithm convergence speed (Figure 5). As it could be seen in Figure 5, the initial population of the archive of the non-dominated solutions has a high impact on the convergence speed of the other members toward the Pareto-optimal front (POF).

4.2.1. Opposition-based Learning (OBL)

The concept of OBL was introduced for the first time in 2005, and it has been used to improve the performance in many scientific researches works such as evolutionary optimization algorithms and artificial neural networks [46]. The concept of OBL has been shown in a one- and two-dimensional space in Figure 6.

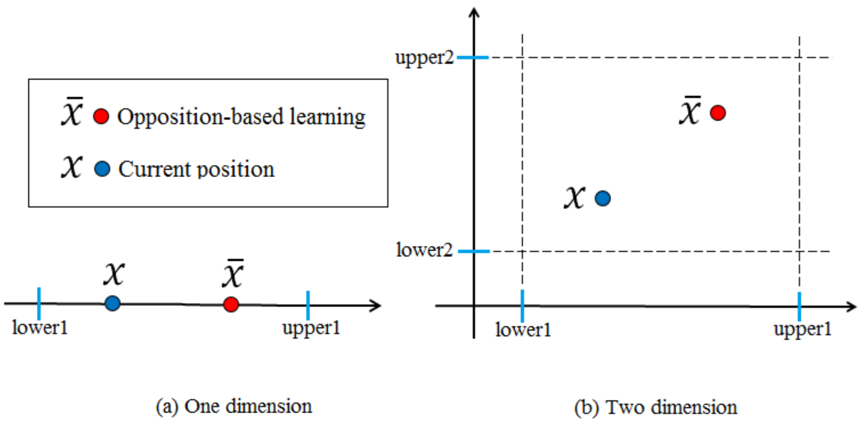


Figure 6. Opposition-based learning.

OBL reverses the current position of the particle based on the scope of the search area, and makes the space more searchable [21 ,43] (see Equation (5)).

$$\bar{x} = lower + upper - x \tag{5}$$

where *lower* and *upper* are the low and the high bound of the search space.

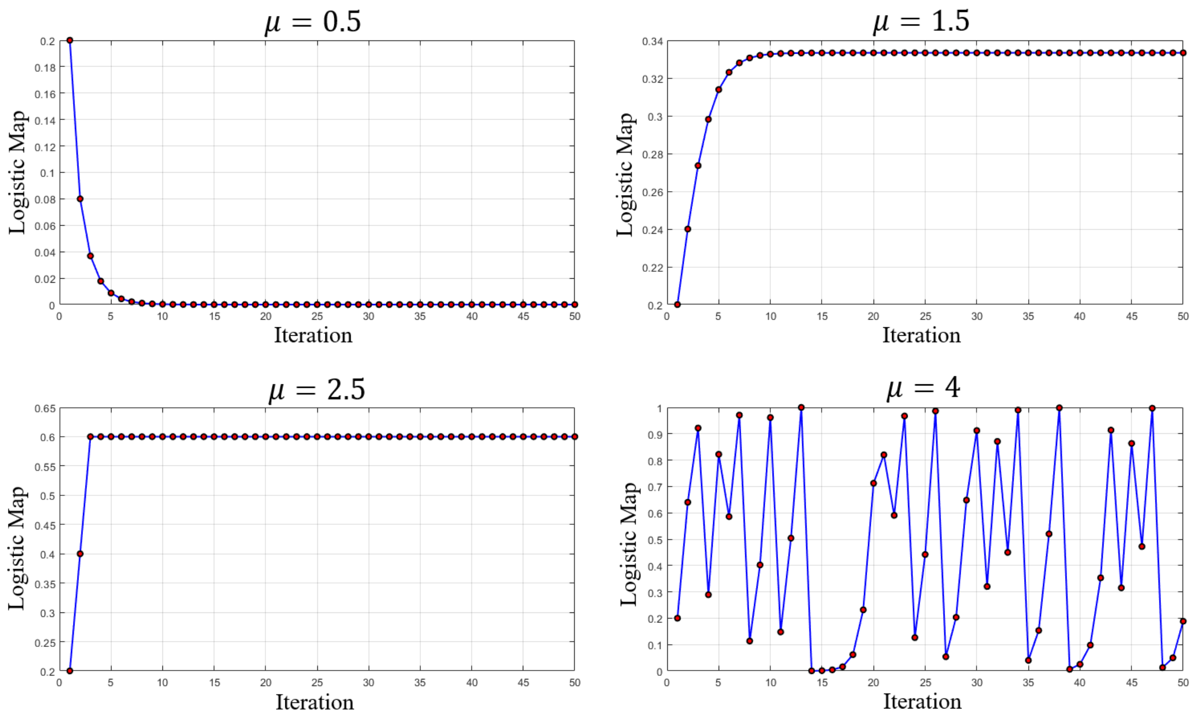


Figure 7. Impact of different values of μ parameter on Logistic map.

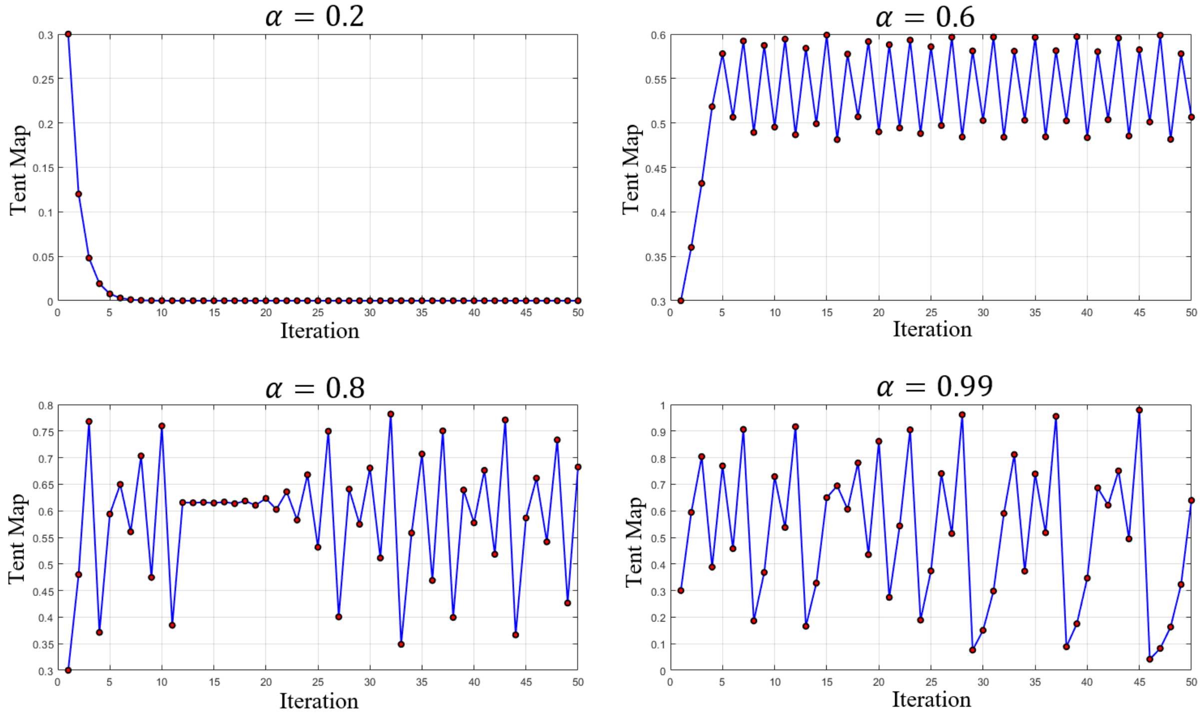


Figure 8. Impact of different values of α parameter on Tent map.

4.2.2. Chaotic Maps

The random number generation plays a very important role in the evolutionary optimization algorithms. Although the computer systems are not good generators, they are widely used in order to generate random numbers; however, their patterns are not easily recognizable. Chaotic is a kind of phenomenon that occurs in definable non-linear systems, and is highly sensitive to the initial conditions and quasi-random behaviors. A series of random functions are used in order to generate random numbers in the evolutionary algorithms; chaotic functions can be used to generate random numbers, and random behaviors can be replaced with chaotic behaviors.

The Logistic and Tent maps were used to generate random numbers in this research work; $\mu = 0.3$ and $\alpha = 4$ were used to adjust the parameters of these maps. As shown in Figure 7, the Logistic map has the most diversity in generating the random numbers when $\mu = 0.4$ (also for the Tent map, according to Figure 8).

The production of an initial population with the uniform distribution by Equation (6) was carried out:

$$x = lower + rand(upper - lower) \tag{6}$$

where *rand* represents a random number between *zero* and *one*, which has a very sensitive and essential role in the appropriate distribution of the initial population. Several chaotic maps to generate random numbers by Equation (7) were used.

$$x = lower + cm(upper - lower) \tag{7}$$

where *cm* is a chaotic map. So far, different maps, including Piecewise, Sine, Chebyshev, Circle, Gauss/mouse, Iterative, Logistic, Singer, Sinusoidal, and Tent have been provided [19].

Two chaotic maps of Tent and Logistic have been used in the methods proposed here, namely, chaotic maps that are utilized to obtain a sequence of random numbers; Equation (8) and (9) show the Tent and Logistic maps, respectively.

$$cm_{Tent}(t+1) = \begin{cases} 2\alpha \cdot cm_{Tent}(t) & cm_{Tent}(t) < 0.5 \\ 2\alpha \cdot (1 - cm_{Tent}(t)) & cm_{Tent}(t) \geq 0.5 \end{cases} \tag{8}$$

$$cm_{Logistic}(t+1) = \mu \cdot cm_{Logistic}(t) \cdot (1 - cm_{Logistic}(t)) \tag{9}$$

In the Tent map $\alpha = 0.99$ and in the Logistic map, $\mu = 4$; $cm_{Tent}(0)$ and $cm_{Logistic}(0)$ are random numbers between *zero* and *one*. In this work, by providing the appropriately combined methods, the highly efficient initial population to start the algorithm was created. Figure 9 depicts the production of the initial population in the proposed algorithm. In Figure 6, *ps* is the population size, *P(ps)* is an initial population with the uniform distribution, *OP(ps)* is the opposite of *P(ps)*, *Ch(ps)* is an initial population with the chaotic maps, and *OCh(ps)* is the opposite of *Ch(ps)*.

The opposite population (*OP*) means that all members of the current population (*P*(*ps*)) are individually counted by their opposite based on

Equation 5, and the recently generated population is known as the opposite population.

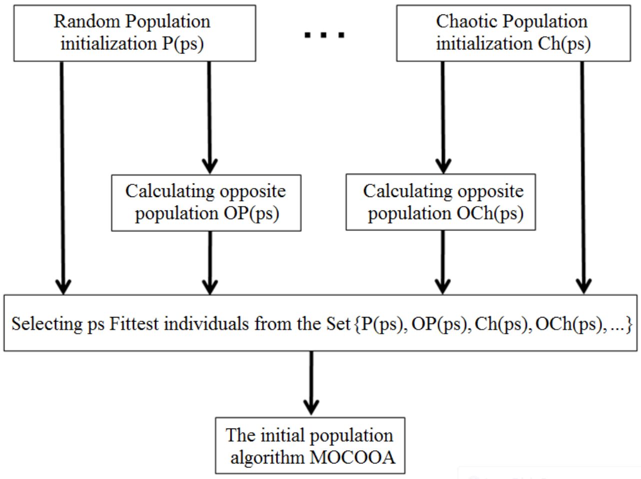


Figure 9. Production of initial population in the proposed algorithm.

4.3. Cost Function Value for MOPs

Solving MOPs, due to the contradiction between the objectives, there is no similar answer, indicating that all the objectives are the best. Finally, a group of non-dominated solutions, as the optimum solutions (near-optimal), were presented, which are known as the solutions archive of Pareto. In MOCOOA, an external archive is was to keep the non-dominated solutions obtained by the proposed algorithm, as the archive in each iteration of the algorithm is updated. In COOA, the concept of competition is defined based on the cost function value (*fitness*), and the power of a group is calculated via the following equation:

$$TC_n = fitness(imperialist_n) + \delta \cdot mean\{fitness(colonies\ of\ imperialist_n)\} \quad (10)$$

where $0 < \delta < 1$, the value of δ makes the colonies role to determine the total power of an empire. Herein, 0.3 was dedicated to δ in most of our implementations. According to Equation 10, two values must be calculated to define the fitness of a group. The first and second value of fitness are the strongest member of the group and the mean fitness value of the other members of the group, respectively. We have to pay more attention to the fitness value of the strongest member of the group in order to better understand the fitness of a group. On the other hand, affecting the fitness mean of the other members of the group may have a negative effect on the overall fitness of a group. The effect of the mean value of fitness on the other group members is controlled based on the δ parameter. The value of the δ parameter can be picked from the interval [0,1] but it is better to pick from the interval [0.1, 0.5]. Afterwards, all the empires compete with each other. In order to select the weakest empire and the weakest member in it, the TC_n value was used; the weakest member will be transferred to another empire based on the roulette wheel. It could be seen in Equation (10) that the definition of the cost function value of the population members in the proposed algorithm is required.

The *fitness* value of all the population members based on the quality and diversity of solutions with the following equation was calculated. The lower *fitness* value shows a better member.

$$fitness_i = (Rank_i \cdot m) - NCD_i \quad (11)$$

$Rank_i$ refers to the rank related to the *i*-th member of the population and *m* is the number of objectives; NCD_i, the value of the new crowding distance of *i*-th the population member is calculated using Equation 11.

4.4. A Global Search Strategy

For a better search around the external archive solutions and to improve the position of the members of each group, the new position of the member was calculated in the equation below. If the new position could dominate its current position, it will be replaced by:

$$xnew_i^d = x_i^d + alpha \cdot (leader^d - x_i^d) \quad (12)$$

x_i^d is the d-th dimension of the i-th member, alpha is a random number in the [0, 1] range and leader (global optimal solutions) is the position of an external archive member, which are randomly selected from the external archive for the i-th member. The alpha parameter value is picked randomly from the interval [0,1] for each member of the population so that a better search around leader is helped by this parameter.Given the fact that an idea for obtaining the value of a population member cost function was provided based on all the objective functions for all members, we could exactly run MOCOOA like COOA. The existing members within the external archive are always used as a leader (global optimal solutions) in the algorithms of PSO, CSO, ACO, and GWO. A flowchart of the proposed algorithm is shown in Figure 11. Also, the MOCOOA process is shown in Algorithm 1 (Appendix 1).

4.5. Convergence Control in Proposed Algorithm

The evolutionary optimization algorithms include the two phases of exploration and exploitation. In the exploration phase, a global search, and in the exploitation phase, a local search is performed. The evolutionary optimization algorithms usually start with the exploration phase and enter the exploitation phase over time. In the optimization process, ideally, the change curve of the cost function should decrease and reach its minimum over time. In an optimization problem, the global optimum solution is not known in advance, and the only thing that can help improve the optimization process in the evolutionary algorithm is to increase the diversity in the member of population. When the change curve of cost function is continuously reduced, it can be seen that the optimization process goes well, and when the change curve of cost function is in a stagnancy state (unchanged), the only thing that can be understood is that we are either trapped in a local optimum trap or that we have reached the global optimum solution. Hence, we face uncertainty, inevitably; given that we may be caught in the local optimum trap, we must somehow improve the optimization process. Many evolutionary optimization algorithms continuously switch between the two phases of exploration and exploitation during the optimization process in order to avoid being trapped in the local optimum.

In the proposed algorithm, there are two perspectives to control the exploration and exploitation phases. In the first perspective, within the proposed algorithm, several evolutionary algorithms including CSO, PSO, GWO, and ACO are used, and each one of these algorithms has the ability to balance between the two phases of exploration and exploitation independently. For example, in the CSO algorithm, the balance between the two phases of exploration and exploitation is done through the mixture rate (MR) parameter, and each particle can be randomly placed in one of the two phases of exploration and exploitation or in the PSO algorithm, the balance between these two phases is done through inertia weight. In the second perspective, according to the feedback obtained from the reduction of the change curve in cost function, a mechanism for balancing these two phases is considered. If the proposed algorithm gets stagnancy, for example, according to Figure 10(a), if after a certain number of steps, the number of species in the proposed algorithm is reduced to one species (here CSO) and the algorithm get stagnancy, the population of the proposed algorithm is reset. Two scenarios are considered for the population reset. In the first scenario, as shown in Figure 10(b), by maintaining the position of all members of the population, all the inactive species will be reactivated, and the members of the population will be divided among all the active species. This makes the algorithm search mechanism more powerful. In the second scenario, according to Figure 10(b), by maintaining the position of the top members in the current population of the proposed algorithm, a population with a uniform distribution is reproduced and the new population members are divided among all species, which increases the diversity of the population members. The priority of the first scenario is superior to the second scenario, and in the proposed algorithm, if after a few steps the first scenario cannot reduce the change curve of cost function, the second scenario will be applied.

5. Experimental Results

In this section, a simulation study is carried out through MATLAB in order to demonstrate the potentiality of MOCOOA for solving the benchmark and real-life multi-objective engineering design problems. In order to analyze and evaluate the simulation results, the proposed algorithm was compared with the other algorithms such as MOPSO [5], MOGWO [13], MOCSO [14], MOICA [7], NSGA-II [6], MOEA/IGD-NS [25], and BCE-IBEA [30].

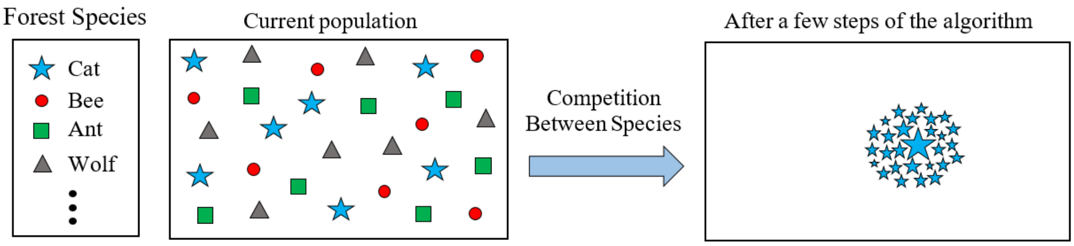
5.1. Performance Measure

In order to examine the performance of MOCOOA, some MOPs were utilized in experiments. In order to examine the performance of the proposed algorithm, the standard performance measures of MOEAs were employed, which represented the quantitative and qualitative comparisons with MOEAs. As to these metrics, POF for an MOP is required, and here, 500 uniformly spaced Pareto optimal solutions were used as the POF approximation. The two metrics of inverted generational distance (IGD) [3] and spread metric (Δ) [6] were used in order to evaluate the results.

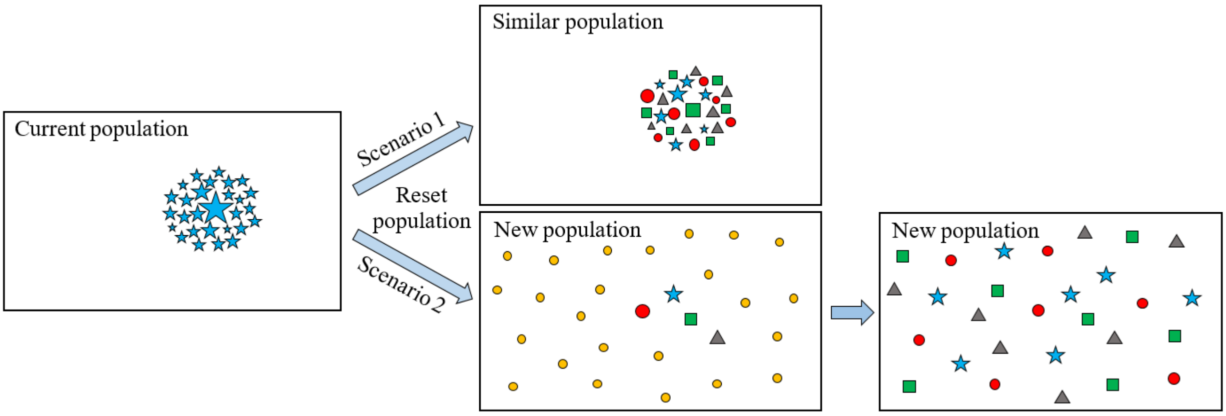
5.2. Default Parameter Settings

The default parameter settings for NSGA-II, MOPSO, MOCSO, MOGWO, MOICA, and MOCOOA are presented in Table 1. The initial population for solving different MOPs from 64 to 128 was considered.

In terms of setting the parameter values of different algorithms, it can be mentioned that the default parameter values are utilized for each algorithm.



(a) Competition between species



(b) Uniform distribution

Figure 10. Exploration and exploitation in the proposed algorithm.

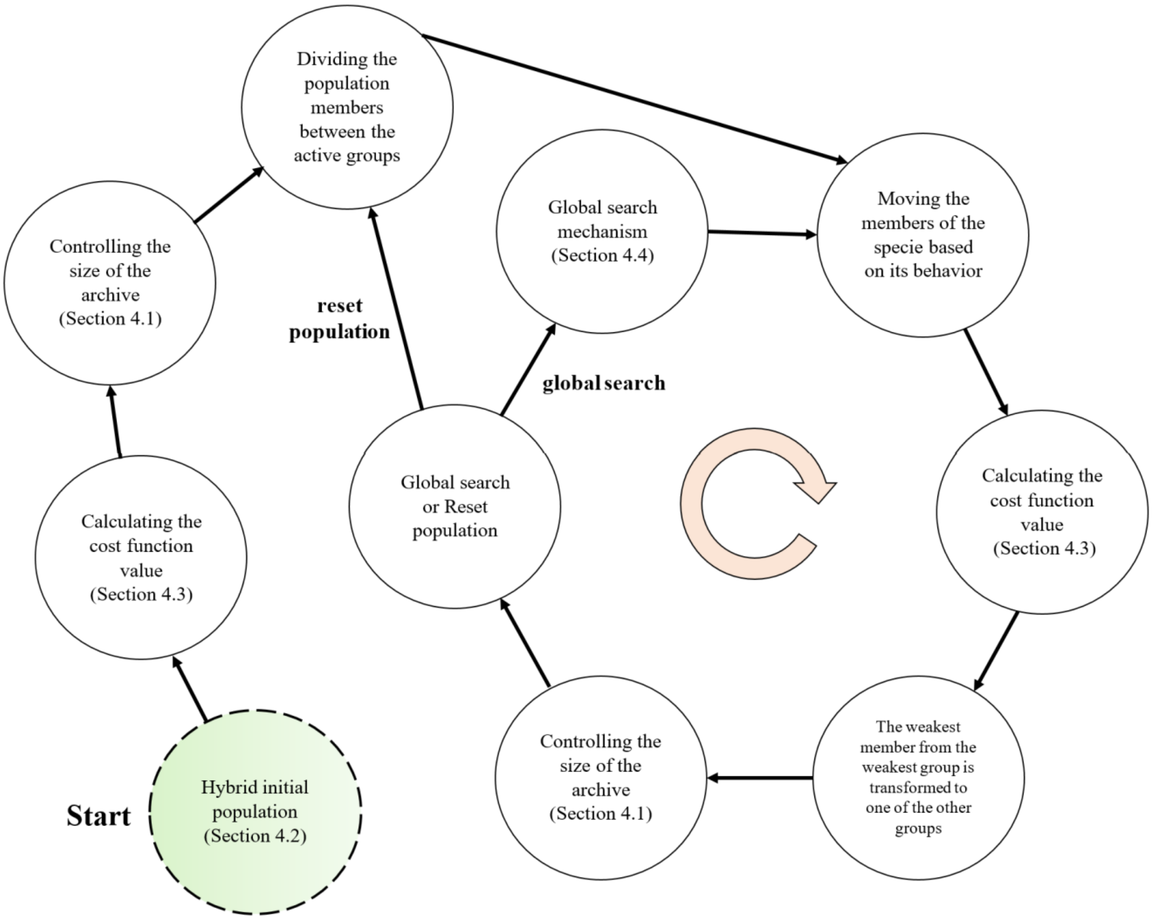


Figure 11. A flowchart of MOCOOA.

For example, the parameters $c_1 = 2.05$ and $c_2 = 2.05$ were employed in the PSO algorithm. Besides, the parameter ω (inertia weight) decreases linearly in each step of the algorithm. Meanwhile, for the CSO algorithm, both parameters of $c_1 = 2.05$ and ω were set similar to the ones for the PSO algorithm. Furthermore, for the other algorithms, the default values of the algorithm are utilized in the same way (according to Table 1). On the other hand, as the number of these algorithms were employed within MOCOOA, the parameter setting of these algorithms was similar to what was described.

5.3. Benchmark Test Functions

Several assessment functions were used in order to assess the proposed algorithm. Three multi-objective engineering design optimization problems were considered in the first part which included the solution of these problems, four-bar truss design, disk brake design, and two-bar truss design. In the second part, 10 CEC benchmark assessment functions (UF1-UF10) were used, and the details of these benchmark functions are reported in Appendix 1.

Table 1. Default parameter settings.

Algorithm	Parameter	Value
MOPSO	c_1, c_2	2.05
	α (grid inflation)	0.1
	β	4
	nGrid	10
	ω	0.9 TO 0.1
MOCSO	SMP	10-15
	SRD	0.25
	c_1	2.05
	ω	0.4
	MR	0.5
NSGA-II	Crossover probability	0.8
	Mutation probability	1/dimension
MOGWO	$\alpha = 0.1$	
	$\beta = 4$	
	$nGrid = 10$	
MOICA	Assimilation coefficient = 2.0	
	Revolution Probability = 0.1	
	Probability of Revolution on a Specific Variable = 0.15	

5.3. Multi-objective Engineering Design Problems

In this subsection, the proposed MOCOOA is used for three real problems in the engineering design [22,29], whose two problems with constraints (two-bar truss design problem and multi-plate disk brake design), and one problem without constraint (four-bar truss design problem) (see Figure 12).

5.3.1. Multi-objective Optimization of Four-bar Truss Design

The objective of this problem is to minimize the truss volume and joints' displacement simultaneously. This problem is unconstrained with continuous design variables [29]. Table 2 summarizes the final optimization results as to the mean and standard deviation of all the best solution metrics values, achieved for the four-bar truss design problem by the six algorithms over 20 independent runs of each algorithm. In order to validate the statistical difference between MOCOOA and other algorithms, according to Table 2, it is clear that MOCOOA is superior to the other optimization methods based on the mean and standard deviation values of the performance of all the metrics.

$$Minimize\ f_1(x) = L(2x_1 + \sqrt{2}x_2\sqrt{x_3} + x_4) \tag{13}$$

$$Minimize\ f_2(x) = \frac{F.L}{E}\left(\frac{2}{x_1} + \frac{2\sqrt{2}}{x_2} - \frac{2\sqrt{2}}{x_3} + \frac{2}{x_4}\right) \tag{14}$$

where

$$\begin{aligned}
&\left(\frac{F}{\sigma}\right) \le x_1 \le 3\left(\frac{F}{\sigma}\right), \quad \sqrt{2}\left(\frac{F}{\sigma}\right) \le x_2 \le 3\left(\frac{F}{\sigma}\right), \\
&\sqrt{2}\left(\frac{F}{\sigma}\right) \le x_3 \le 3\left(\frac{F}{\sigma}\right), \quad \left(\frac{F}{\sigma}\right) \le x_4 \le 3\left(\frac{F}{\sigma}\right), \\
&F = 10kN, \quad \mathrm{E}=2\times10^5\,\frac{kN}{cm^2}, \\
&L = 200cm, \quad \sigma=10\,\frac{kN}{cm^2}
\end{aligned} \tag{15}$$

Figure 13 demonstrates the Pareto fronts obtained by MOCOOA, and other algorithms for the four-bar truss design problem after 8000 evaluations of the objective function. In each figure, the true Pareto front is shown as a continuous line. In Figure 13, the quality of the solution obtained is much better and more disciplined than the other optimization methods. According to Figure 13, the NSGA-II, MOPSO, MOGWO, and MOCOOA algorithms are capable of covering all the parts of the Pareto optimal front compared to the other optimization methods. MOCOOA achieved better non-dominated solutions in terms of metrics compared to the NSGA-II, MOGWO, and MOPSO algorithms.

5.3.2. Multi-objective Multi-plate Disk Brake Design

The multi-plate disk brake design problem is mainly applied in airplanes for an effective braking while landing. In this problem, all the design variables are continuous [22]. Table 2 summarizes the final optimization results as to the mean and standard deviation of all the best solution metrics values achieved for the multi-plate disk brake problem by the six algorithms over 20 independent runs of each algorithm. In order to validate the statistical difference between MOCOOA and other algorithms, according to Table 2, it is clear that MOCOOA is superior to the other optimization methods based on the mean and standard deviation values of the performance of all the metrics.

Figure 14 demonstrates the Pareto fronts obtained by MOCOOA and other algorithms for the multi-plate disk brake problem after 20000 evaluations of the objective function. In Figure 14, the quality of the obtained solution is much better and more disciplined than the other optimization methods. According this figure, the MOCSO, MOPSO, MOGWO, and MOCOOA algorithms are capable of covering all parts of the Pareto optimal front compared to the other optimization methods. MOCOOA achieved better non-dominated solutions in terms of metrics compared to the other algorithms.

$$Minimize\ f_1(x) = 4.9\times10^{-5}(x_2^2 - x_1^2)(x_4 - 1) \qquad (16)$$

$$Minimize\ f_2(x) = \frac{9.82\times10^6(x_2^2 - x_1^2)}{x_3 x_4 (x_2^3 - x_1^3)} \qquad (17)$$

subject to

$$\begin{aligned}
&g_1(x) = 20 + x_1 - x_2 \le 0 \\
&g_2(x) = 2.5(x_4 + 1) - 30 \le 0 \\
&g_3(x) = \frac{x_3}{3.14(x_2^2 - x_1^2)^2} - 0.4 \le 0 \\
&g_4(x) = \frac{2.22\times10^{-3} x_3 (x_2^3 - x_1^3)}{(x_2^2 - x_1^2)^2} - 1 \le 0 \\
&g_5(x) = 900 - \frac{2.66\times10^{-2} x_3 x_4 (x_2^3 - x_1^3)}{(x_2^2 - x_1^2)} \le 0
\end{aligned} \qquad (18)$$

where

$$\begin{aligned}
&55 \le x_1 \le 80, \quad 75 \le x_2 \le 100, \\
&10^3 \le x_3 \le 3\times10^3, \quad 2 \le x_4 \le 20,
\end{aligned} \qquad (19)$$

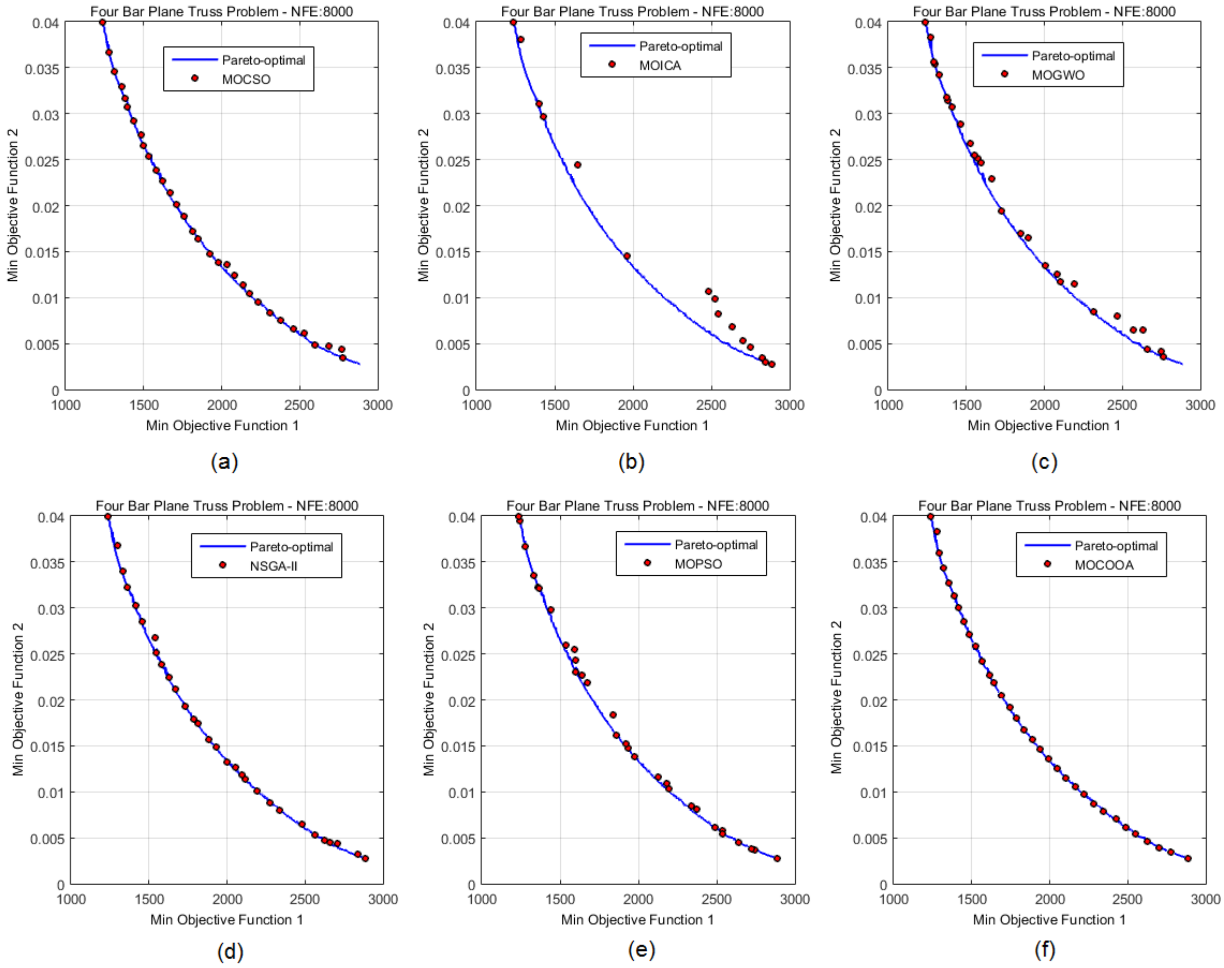


Figure 13. Comparison of MOCSO, MOICA, MOGWO, NSGA-II, MOPSO, and MOCOOA for the four-bar truss design problem.

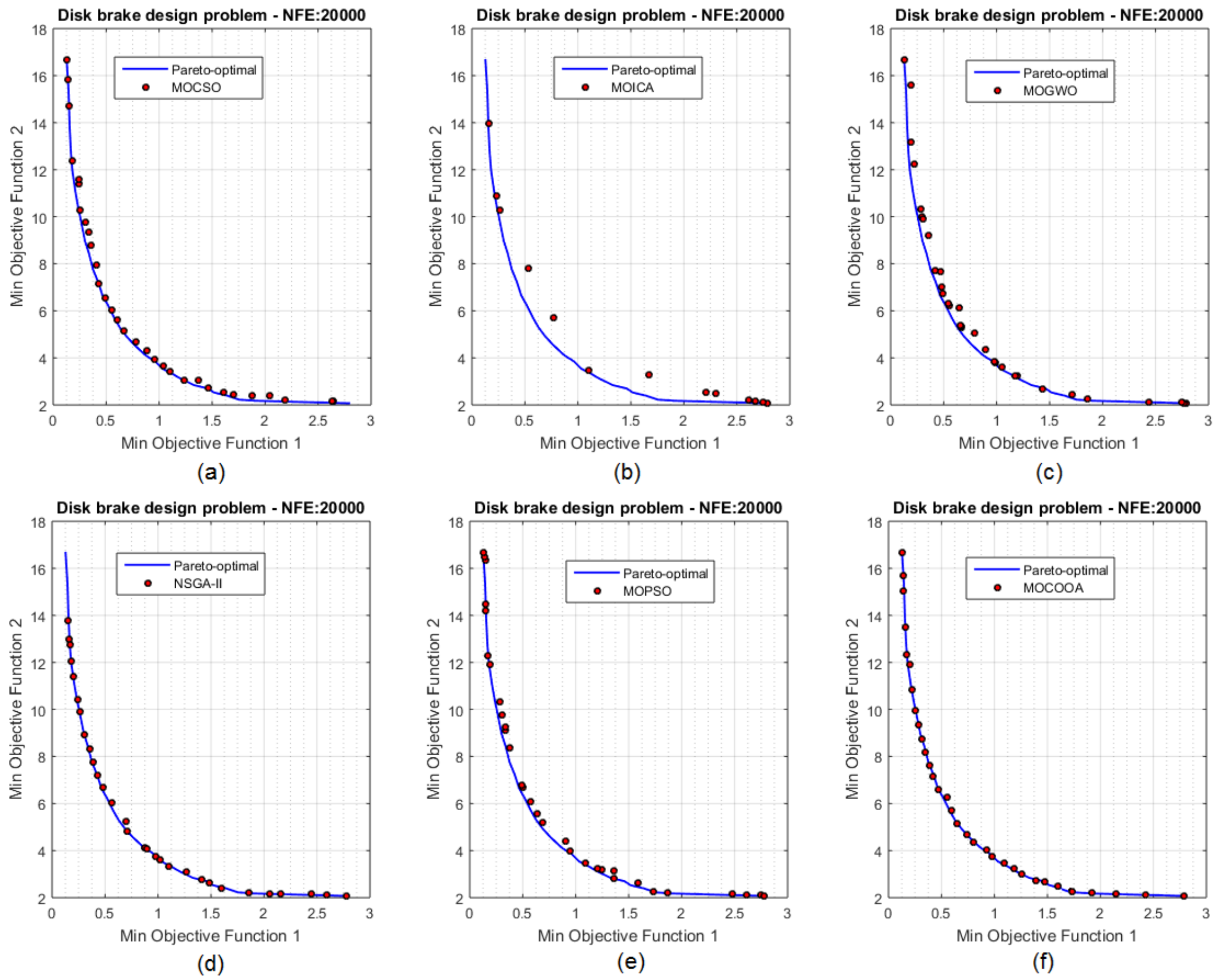


Figure 14. Comparison of MOCSO, MOICA, MOGWO, NSGA-II, MOPSO, and MOCOOA for the multi-plate disk brake problem.

5.3.3. Multi-objective Optimization of Two-bar Truss Design

The objective of this problem is to minimize the truss weight and joints' displacement simultaneously. This problem is bounded by two objective functions and four design variables. Also, all the design variables are continuous.

$$Minimize\ f_1(x) = 2\rho h x_2 \sqrt{1+x_1^2} \tag{20}$$

$$Minimize\ f_2(x) = \frac{Ph(1+x_1^2)^{1.5}(1+x_1^4)^{0.5}}{2\sqrt{2}Ex_2x_1^2} \tag{21}$$

subject to

$$\begin{aligned} g_1(x) &= \frac{P(1+x_1)(1+x_1^2)^{0.5}}{2\sqrt{2}x_2x_1} - \sigma_0 \le 0 \\ g_2(x) &= \frac{P(1-x_1)(1+x_1^2)^{0.5}}{2\sqrt{2}x_2x_1} - \sigma_0 \le 0 \end{aligned} \tag{22}$$

where

$$\begin{aligned} &0.1 \le x_1 \le 2.25, \quad 0.5 \le x_2 \le 2.5, \\ &P = 10^4 lb, \quad E = 3\times 10^7 \frac{lb}{in^2}, \quad h = 100in, \\ &\sigma_0 = 2\times 10^4 \frac{lb}{in^2}, \quad \sigma = 0.283\frac{lb}{in^3} \end{aligned} \tag{23}$$

Figure 15 demonstrates the Pareto fronts obtained by MOCOOA and the other algorithms for the two-bar truss design problem after 8000 evaluations of the objective function. In each figure, the true Pareto front is shown as a continuous line. In Figure 15, the quality of the obtained solution is much better and more disciplined than the other optimization methods. According to Figure 15 and Table 2, the NSGA-II, MOPSO, and MOCOOA algorithms are capable of covering all parts of the Pareto optimal front compared to the other optimization methods. MOCOOA achieved better non-dominated solutions in terms of metrics compared to the NSGA-II and MOPSO algorithms.

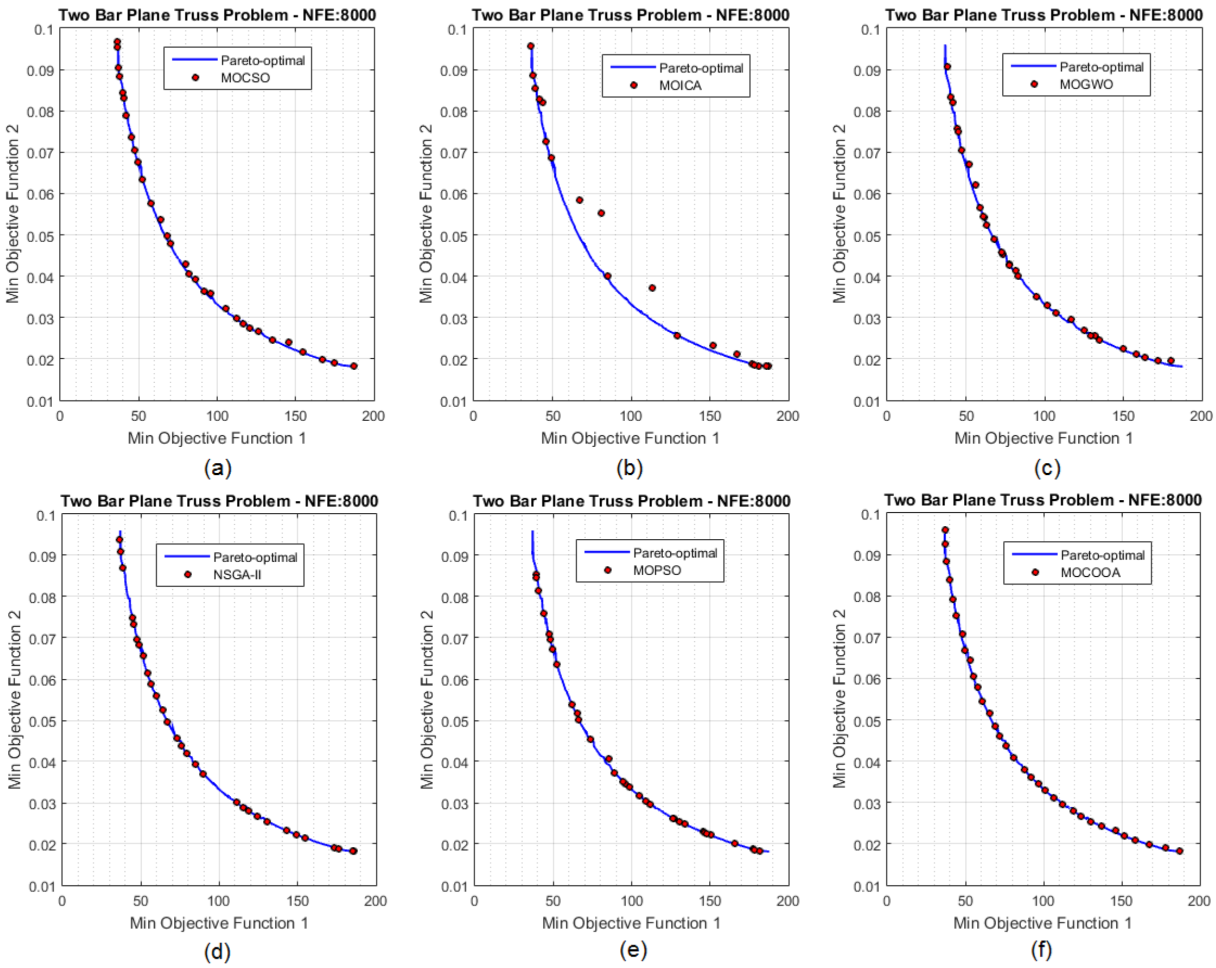


Figure 15. Comparison of MOCSO, MOICA, MOGWO, NSGA-II, MOPSO, and MOCOOA for the two-bar truss design problem.

Table 2. Comparison of MOCOOA and other algorithms.

Problem	Metric	NSGA-II	MOCOOA	MOPSO	MOCSO	MOICA	MOGWO
Four-bar truss design	*IGD*	0.4461 ± 0.0606	**0.4226 ± 0.0335**	0.4653 ± 0.0382	0.5798 ± 0.1299	2.7578 ± 0.0158	0.5092 ± 0.0281
	Δ	0.5041 ± 0.1149	**0.3165 ± 0.0443**	0.6217 ± 0.1254	0.4576 ± 0.011	1.0978 ± 0.0623	0.6679 ± 0.1064
Multi-disk brake design	*IGD*	0.0393 ± 0.0439	**0.0262 ± 0.0248**	0.0366 ± 0.0412	0.0386 ± 0.0282	0.0708 ± 0.0652	0.0346 ± 0.0348
	Δ	0.6534 ± 0.0375	**0.5662 ± 0.0481**	0.7232 ± 0.0809	0.6033 ± 0.0213	1.0999 ± 0.4675	0.8153 ± 0.0372
Two-bar truss design	*IGD*	0.1129 ± 0.0109	**0.1011 ± 0.0832**	0.1302 ± 0.0008	0.1345 ± 0.0421	0.1815 ± 0.0392	0.1231 ± 0.0572
	Δ	0.5965 ± 0.0314	**0.4017 ± 0.0191**	0.7213 ± 0.1293	0.4681 ± 0.0215	0.7779 ± 0.0395	0.7138 ± 0.0434

5.4. Comparison Among MOCOOA, MOEA/IGD-NS, and BCE-IBEA

In this sub-section, MOCOOA is compared with other well-known meta-heuristic algorithms, namely MOEA/IGD-NS [25] and BCE-IBEA [30]. The details of the CEC benchmark test functions are shown in [25]. The maximum number of fitness evaluation (NFE) in the case of all algorithms is 300,000 for the CEC benchmark test functions. For two-objective problems, the population size is 100, and for the three-objective optimization problems, it is 160. The MOEA/IGD-NS algorithm is an indicator-based MOEA, which uses an enhanced IGD metric to detect non-contributory solutions, which can accelerate convergence to PF at each step of the optimization process [25]. The BCE-IBEA algorithm is an indicator-based criterion MOEA, which embeds the Pareto criterion in the indicator-based evolutionary algorithm (IBEA) using an external archive storing well-distributed non-dominated solutions obtained in the evolution process [30]. The results listed in Table 3 were obtained by evaluating all the algorithms on 10 test functions. According to the results in tabulated, the good performance of MOCOOA, in comparison with the other algorithms, is perfectly shown. The bold values in Table 3 indicate the best results in the algorithms as to the mean values.

Table 3. Performance comparisons of IGD values on UF1-UF10.

Problem	Metric	MOEA/IGD-NS	BCE-IBEA	MOCOOA
UF1	IGD	8.69E-02	4.07E-02	**7.98E-03**
	Rank	3	2	**1**
UF2	IGD	**3.16E-02**	3.40E-02	3.25E-02
	Rank	**1**	3	2
UF3	IGD	5.07E-02	5.65E-02	**1.06E-02**
	Rank	2	3	**1**
UF4	IGD	4.29E-02	4.07E-02	**8.47E-03**
	Rank	3	2	**1**
UF5	IGD	2.08E-01	1.96E-01	**1.61E-01**
	Rank	3	2	**1**
UF6	IGD	2.10E-01	2.13E-01	**1.77E-01**
	Rank	2	3	**1**
UF7	IGD	4.57E-02	2.00E-02	**1.64E-02**
	Rank	3	2	**1**
UF8	IGD	3.25E-01	**2.07E-01**	5.88E-01
	Rank	2	**1**	3
UF9	IGD	3.90E-01	1.52E-01	**1.16E-01**
	Rank	3	2	**1**
UF10	IGD	**3.47E-01**	1.56E+00	8.07E-01
	Rank	**1**	3	2

The results of the mean IGD of these six MOEAs on CEC benchmark test functions (UF1-UF10) are listed in Table 3. Clearly, MOCOOA yields significantly better outcomes for seven out of 10 benchmark test functions, whereas MOEA/IGD-NS and BEC-IBEA are best on 2 and 1, respectively.

6. Conclusions

In the present work, in order to expand the single-objective version of the competitive optimization algorithm (COOA) to its multi-objective version (MOCOOA), three new contributions were presented. Primarily, a novel method was presented in order to prune the external archive and maintain the diversity in PF. Secondly, the appropriate and efficient initial population to start the proposed algorithm was defined. Thirdly, the concept of the cost function value for MOPs was redefined. In order to analyze and evaluate the simulation results, the proposed algorithm was compared with the other algorithms including MOPSO, MOGWO, MOCSO, MOICA, MOEA/IGD-NS, BCE-IBEA, and NSGA-II. The results of MOCOOA, compared to the other methods, revealed a faster convergence toward POF with maintaining the diversity of the members' diversity in the final PF for solving MOPs. Nevertheless, the proposed MOCOOA could still be improved, and other new algorithms of swarm intelligence could be used which have recently been published in the journals as a new species within MOCOOA, and the power of this algorithm could be evaluated in order to solve different optimization problems. The optimization problems from the dynamic domain could also be solved by improving and modifying the proposed algorithm.

References

[1] G. Zhang, H. Wang, W. Zhao, Z. Guan, and P. Li, "Application of Improved Multi-objective Ant Colony Optimization Algorithm in Ship Weather Routing," *Journal of Ocean University of China*, 20(1), pp.45-55, 2021.

[2] A. Kaveh, P. Rahmani, and A.D. Eslamlou, "An efficient hybrid approach based on Harris Hawks optimization and imperialist competitive algorithm for structural optimization," *Engineering with Computers*, pp.1-29, 2021.

[3] Y. Sun, G. G. Yen, and Z. Yi, "IGD indicator-based evolutionary algorithm for many-objective optimization problems," *IEEE Transactions on Evolutionary Computation*, Vol. 23 (2), pp. 173-87, 2018.

[4] N. Yerpula, "Cat Swarm Optimization Algorithm for Antenna Array Synthesis," Turkish Journal of Computer and Mathematics Education (TURCOMAT), 12(2), pp.1466-1474, 2021.

[5] F. Han, W.T. Chen, Q.H. Ling, and H. Han, "Multi-objective particle swarm optimization with adaptive strategies for feature selection," *Swarm and Evolutionary Computation*, 62, p.100847, 2021.

[6] K. Deb, A. Pratap, S. Agarwal, and T. A. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE transactions on evolutionary computation*, Vol. 6 (2), pp. 182-97, 2002.

[7] M. Khanali, A. Akram, J. Behzadi, F. Mostashari-Rad, Z., Saber, K.W. Chau, and A. Nabavi-Pelesaraei, "Multi-objective optimization of energy use and environmental emissions for walnut production using imperialist competitive algorithm," *Applied Energy*, 284, p.116342, 2021.

[8] F. A. Fortin and M. Parizeau, "Revisiting the NSGA-II crowding-distance computation," *InProceedings of the 15th annual conference on Genetic and evolutionary computation*; pp. 623-630, 2013.

[9] X. Zhou, J. Lu, J. Huang, M. Zhong, and M. Wang, "Enhancing artificial bee colony algorithm with multi-elite guidance," *Information Sciences*, 543, pp.242-258, 2021.

[10] K. Shang, H. Ishibuchi, L. He, and L.M. Pang, "A survey on the hypervolume indicator in evolutionary multi-objective optimization," *IEEE Transactions on Evolutionary Computation*, 25(1), pp.1-20, 2020.

[11] X. Ma, F. Liu, Y. Qi, M. Gong, M. Yin, L. Li, L. Jiao, and J. Wu, "MOEA/D with opposition-based learning for multi-objective optimization problem," *Neurocomputing*, Vol. 146, pp. 48-64, 2014.

[12] Y. Gao, L. Peng, F. Li, M. Liu, and W. Li, "Multi-objective opposition-based learning fully informed particle swarm optimizer with favour ranking," *In2013 IEEE International Conference on Granular Computing (GrC)*, pp. 114-119, 2013.

[13] Y. Yang, B. Yang, S. Wang, T. Jin, S. Li, "An enhanced multi-objective grey wolf optimizer for service composition in cloud manufacturing," *Applied Soft Computing*, 87, p.106003, 2020.

[14] D. Yan, H. Cao, Y. Yu, Y. Wang, and X. Yu, "Single-objective/multi-objective cat swarm optimization clustering analysis for data partition," *IEEE Transactions on Automation Science and Engineering*, 17(3), pp.1633-1646, 2020.

[15] L. M. Antonio and C. A. Coello, "Coevolutionary multi-objective evolutionary algorithms: Survey of the state-of-the-art," *IEEE Transactions on Evolutionary Computation*, Vol. 22 (6), pp. 851-65, 2017.

[16] Y. Sharafi, M. A. Khanesar, and M. Teshnehlab, "COOA: Competitive optimization algorithm," *Swarm and Evolutionary Computation*, Vol. 30, pp. 39-63, 2016.

[17] B. Guan, Y. Zhao, and Y. Li, "An improved ant colony optimization with an automatic updating mechanism for constraint satisfaction problems," *Expert Systems with Applications*, 164, p.114021, 2021.

[18] F. Wang, H. Zhang, and A. Zhou, "a particle swarm optimization algorithm for mixed-variable optimization problems," *Swarm and Evolutionary Computation*, 60, p.100808, 2021.

[19] S. Talatahari and M. Aziz, "Chaos Game Optimization: a novel metaheuristic algorithm," *Artificial Intelligence Review*, 54(2), pp.917-1004, 2021.

[20] T. Joyce and J. M. Herrmann, "A review of no free lunch theorems, and their implications for metaheuristic optimization," *InNature-inspired algorithms and applied optimization;Springer, Cham*, pp. 27-51, 2018.

[21] X. Yu, W. Xu, and C. Li, "Opposition-based learning grey wolf optimizer for global optimization," *Knowledge-Based Systems*, 226, p.107139, 2021.

[22] G. Dhiman and M. Garg, MoSSE: "a novel hybrid multi-objective meta-heuristic algorithm for engineering design problems," *Soft Computing*, 24(24), pp.18379-18398, 2020.

[23] S. Kukkonen and K. Deb, "A fast and effective method for pruning of non-dominated solutions in many-objective problems," *InParallel Problem Solving from Nature-PPSN IX;Berlin, Heidelberg*, pp. 553-562, 2006.

[24] L. T. Bui and T. T. Nguyen, "A competitive co-evolutionary approach for the multi-objective evolutionary algorithms," *IEEE Access*, 8, pp.56927-56947, 2020.

[25] Y. Tian, X. Zhang, R. Cheng, and Y. Jin, "A multi-objective evolutionary algorithm based on an enhanced inverted generational distance metric," *In2016 IEEE congress on evolutionary computation (CEC)*, pp. 5222-5229, 2016.

[26] S. Zapotecas-Martíne, A. López-Jaimes, and A. García-Nájera, "Libea: A lebesgue indicator-based evolutionary algorithm for multi-objective optimization," *Swarm and evolutionary computation*, Vol. 44, pp. 404-19, 2019.

[27] N.T. Tam, T.H. Hung, and H.T.T. Binh, "a decomposition-based multi-objective optimization approach for balancing the energy consumption of wireless sensor networks," *Applied Soft Computing*, 107, p.107365, 2021.

[28] W. Peng and Q. Zhang, "A decomposition-based multi-objective particle swarm optimization algorithm for continuous optimization problems," *In2008 IEEE international conference on granular computing*, pp. 534-537, 2008.

[29] S.A.N. Alexandropoulos, C.K. Aridas, S.B. Kotsiantis, and M.N. Vrahatis, "Multi-objective evolutionary optimization algorithms for machine learning: A recent survey," *In Approximation and optimization* (pp. 35-55). Springer, Cham, 2019.

[30] M. Li, S. Yang, and X. Liu, "Pareto or non-Pareto: Bi-criterion evolution in multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, Vol. 20(5), pp. 645-665, 2015.

[31] H. Kaur, A. Rai, S.S. Bhatia, and G. Dhiman, "MOEPO: a novel multi-objective emperor penguin optimizer for global optimization: special application in ranking of cloud service providers," *Engineering Applications of Artificial Intelligence*, 96, p.104008, 2020.

[32] R. A. El Sehiemy, F. Selim, B. Bentouati, and M.A. Abido, "A novel multi-objective hybrid particle swarm and salp optimization algorithm for technical-economical-environmental operation in power systems," *Energy*, 193, p.116817, 2020.

[33] K. Shang, H. Ishibuchi, L. He, and L.M. Pang, "A survey on the hypervolume indicator in evolutionary multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, 25(1), pp.1-20, 2020.

[34] J.G. Falcón-Cardona and C.A.C. Coello, "Indicator-based multi-objective evolutionary algorithms: A comprehensive survey," *ACM Computing Surveys (CSUR)*, 53(2), pp.1-35, 2020.

[35] X. Zhang, H. Liu, and L. Tu, "A modified particle swarm optimization for multimodal multi-objective optimization," *Engineering Applications of Artificial Intelligence*, 95, p.103905, 2020.

[36] W. A. N. G. Na, S. U. Yuchao, C.H.E.N. Xiaohong, L.I. Xia, and L.I.U. Dui, "a∈-indicator-based shuffled frog leaping algorithm for many-objective optimization problems," *Journal of Systems Engineering and Electronics*, 31(1), pp.142-155, 2020.

[37] C. Audet, J. Bigeon, D. Cartier, S. Le Digabel, and L. Salomon, "Performance indicators in multiobjective optimization," *European journal of operational research*, 2020.

[38] G. Dhiman, K. K. Singh, M. Soni, A. Nagar, M. Dehghani, A. Slowik, A. Kaur, A. Sharma, E.H. Houssein, and K. Cengiz, "MOSOA: a new multi-objective seagull optimization algorithm," *Expert Systems with Applications*, 167, p.114150, 2021.

[39] Q. Xu, Z. Xu, and T. Ma, "a survey of multiobjective evolutionary algorithms based on decomposition: Variants, challenges and future directions," *IEEE Access*, 8, pp.41588-41614, 2020.

[40] X. Ma, Y. Yu, X. Li, Y. Qi, and Z. Zhu, "A survey of weight vector adjustment methods for decomposition-based multiobjective evolutionary algorithms," *IEEE Transactions on Evolutionary Computation*, 24(4), pp.634-649, 2020.

[41] Y. Xie, J. Qiao, D. Wang, and B. Yin, "a novel decomposition-based multiobjective evolutionary algorithm using improved multiple adaptive dynamic selection strategies," *Information Sciences*, 556, pp.472-494, 2021.

[42] Z. Liao, W. Gong, L. Wang, X. Yan, and C. Hu, "a decomposition-based differential evolution with reinitialization for nonlinear equations systems," *Knowledge-Based Systems*, 191, p.105312, 2020.

[43] M. S. Nasrabadi, Y. Sharafi, and M. Tayari, "A parallel grey wolf optimizer combined with opposition-based learning," *In 2016 1st Conference on Swarm Intelligence and Evolutionary Computation (CSIEC)* (pp. 18-23). IEEE, 2016.

[44] J. Shi, M. Gong, W. Ma, and L. Jiao, "A multi-population coevolutionary strategy for multi-objective immune algorithm," *The Scientific World Journal*, 2014.

[45] V. Kumar and D. Kumar, "A systematic review on firefly algorithm: past, present, and future," *Archives of Computational Methods in Engineering*, 28(4), pp.3269-3291, 2021.

[46] S. Mahdavi, S. Rahnamayan, and K. Deb, "Opposition based learning: A literature review," *Swarm and evolutionary computation*, 39, pp.1-23, 2018.

[47] S. Lotfi, and F. Karimi, "A Hybrid MOEA/D-TS for solving multi-objective problems," *Journal of AI and Data Mining*, 5(2), pp.183-195, 2017.

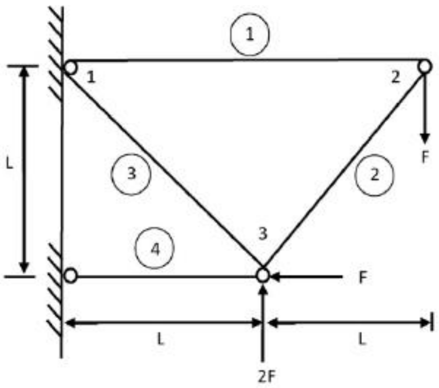
[48] H. Motameni, "PSO for multi-objective problems: Criteria for leader selection and uniformity distribution," *Journal of AI and Data Mining*, 4(1), pp.67-76, 2016.

[49] N. Nekooghadirli, R. Tavakkoli-Moghaddam, and V.R. Ghezavati, "Efficiency of a multi-objective imperialist competitive algorithm: A bi-objective location-routing-inventory problem with probabilistic routes," *Journal of AI and Data Mining*, 2(2), pp.105-112, 2014.

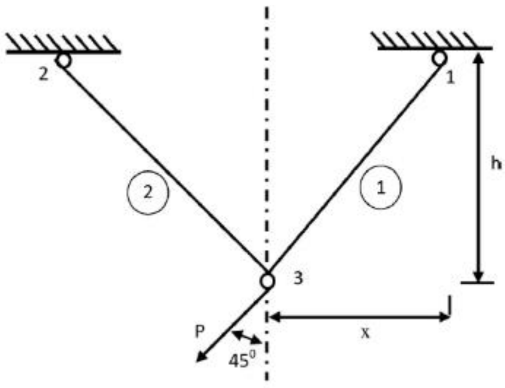
Appendix 1:

Table 4. Details of CEC UF benchmark test functions.

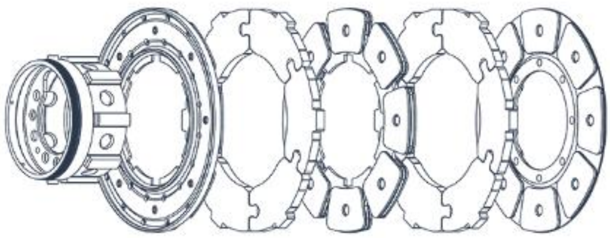
Fun.	Search space	Number of objectives	Number of variables	Characteristics of PF
UF1	$[0,1]\times[-1,1]^{n-1}$	2	30	Concave
UF2	$[0,1]\times[-1,1]^{n-1}$	2	30	Concave
UF3	$[0,1]^{n}$	2	30	Concave
UF4	$[0,1]\times[-2,2]^{n-1}$	2	30	Convex
UF5	$[0,1]\times[-1,1]^{n-1}$	2	30	21-point front
UF6	$[0,1]\times[-1,1]^{n-1}$	2	30	One isolated point and two disconnected parts
UF7	$[0,1]\times[-1,1]^{n-1}$	2	30	Continuous straight line
UF8	$[0,1]^{2}\times[-2,2]^{n-2}$	3	30	Parabolic
UF9	$[0,1]^{2}\times[-2,2]^{n-2}$	3	30	Planar
UF10	$[0,1]^{2}\times[-2,2]^{n-2}$	3	30	Parabolic



(a) Four-bar truss problem



(b) Two-bar truss problem



(c) Multi-plate disk brake problem

Figure 12. Engineering design problems [22, 29].

Algorithm 1: MOCOOA: multi-objective competitive optimization algorithm

---

1. Producing the initial population based on the methods outlined in section 4.2.
2. Non-dominated sorting of all the population members will be carried out.
3. Choosing the initial population and creating the initial non-dominated archive and calculating the cost function value (or *fitness*) with Equation 11.
4. Controlling the size of the archive with the proposed method in section 4.1 (if necessary).
5. Dividing the members of the initial population between the groups such as PSO, CSO, ACO and, GWO based on the cost function value.
6. Moving the members of each algorithm based on its behavior.
7. Calculating the cost function value of all population members by Equation 11.
8. The weakest member of the weakest group is selected based on the power of each group and is given to one of the other groups according to the roulette wheel.
9. The groups with no members are removed.
10. The non-dominated population of the current members, external archive, and new non-dominated population of the archive are updated.
11. Controlling the size of the archive with the proposed method in section 4.1 (if necessary).
12. If the final conditions have not been fulfilled, go to step 13; otherwise, go to step 16.
13. If the population reset conditions occur, go to step 14; otherwise, go to step 15.
14. Aggregating the population of the active groups members and the members of the external archive, for choosing the initial population based on the non-dominated sorting, and calculating the cost function value of the initial population with the Equation 11, and then step 5.
15. The global search strategy with the proposed method in section 4.4, and go to step 6.
16. Reporting the results.

یک رویکرد چند هدفه مبتنی بر الگوریتم بهینه‌سازی رقابتی و کاربردهای مهندسی آن

یوسف شرفی[۱]، محمد تشنه لب[۲،*] و مجتبی احمدیه خانه سر

[۱]دانشکده مهندسی کامپیوتر، واحد علوم و تحقیقات، دانشگاه آزاد اسلامی، تهران، ایران.

[۲]دانشکده مهندسی برق، گروه سیستمها و کنترل، دانشگاه صنعتی خواجه نصیر طوسی، تهران، ایران.

ارسال ۲۰۲۱/۰۵/۰۴؛ بازنگری ۲۰۲۱/۰۷/۰۹؛ پذیرش ۲۰۲۱/۰۸/۱۰

چکیده:

در این تحقیق یک الگوریتم بهینه‌سازی تکاملی چند هدفه جدید بر اساس الگوریتم بهینه‌سازی رقابتی[۱] (COOA) برای حل مسائل بهینه‌سازی چند هدفه[۲] (MOPs) ارائه شده است. الگوریتم بهینه‌سازی رقابتی بر اساس رقابت زیستی الهام گرفته از طبیعت، مابین جاندارانی همانند پرندگان، گربه‌ها، زنبورها و مورچه‌ها عمل می‌کند. این مطالعه شامل نوآوری‌های اصلی به شرح ذیل می‌باشد. اول، یک روش جدید برای هرس آرشیو خارجی ارائه شده است که تنوع پراکندگی در جبهه پارتو[۳] (PF) را حفظ می‌کند. دوم، یک رویکرد ترکیبی از روش‌های یادگیری مبتنی بر تضاد و نگاشت آشوب برای حفظ تنوع در فضای جستجوی جمعیت اولیه پیشنهاد شده است. سوم، یک روش جدید برای تبدیل یک مسئله بهینه‌سازی چند هدفه به یک مسئله بهینه‌سازی تک هدفه ارائه شده است. نتایج شبیه‌سازی الگوریتم پیشنهادی در مقایسه با سایر الگوریتم های بهینه سازی چندهدفه نشان می‌دهد که روش پیشنهادی می‌تواند کاندیدای بهتری برای حل مسائل بهینه‌سازی چند هدفه باشد.

کلمات کلیدی: بهینه‌سازی چندهدفه، الگوریتم بهینه‌سازی رقابتی، جمعیت اولیه، مسائل طراحی مهندسی، فاصله ازدحامی پیشنهادی.

1 Competitive Optimization Algorithm

2 Multi-objective Optimization Problems

3 Pareto Front