Applied Soft Computing 126 (2022) 109287

Contents lists available at ScienceDirect

Applied Soft Computing

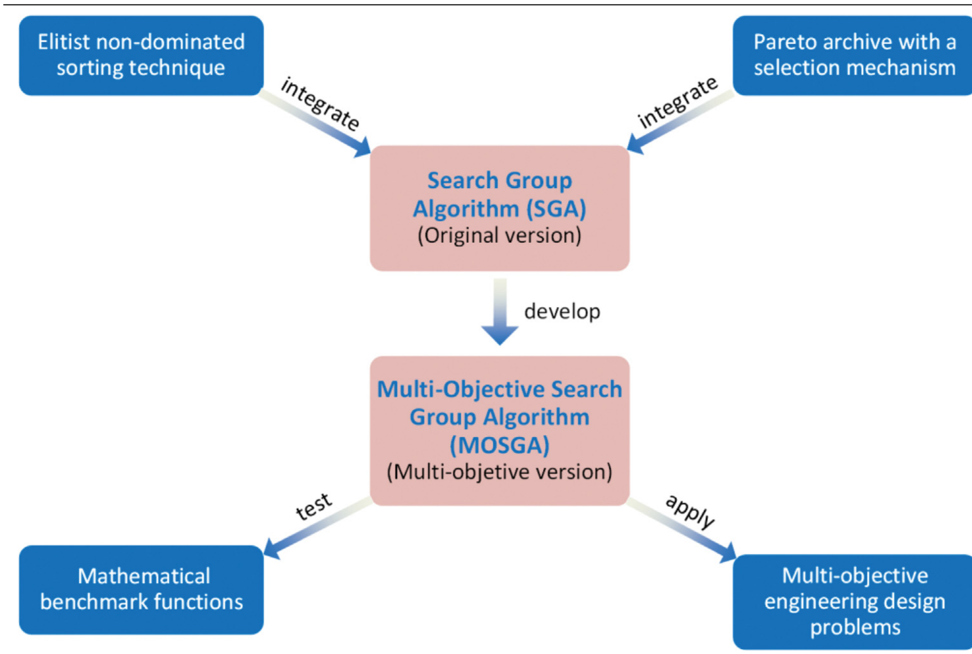
journal homepage: www.elsevier.com/locate/asoc

# Multi-Objective Search Group Algorithm for engineering design problems

Truong Hoang Bao Huy [a], Perumal Nallagownden [b], Khoa Hoang Truong [c,e], Ramani Kannan [b], Dieu Ngoc Vo [d,e,*], Nguyen Ho [f]

[a] Institute of Engineering and Technology, Thu Dau Mot University, Binh Duong Province, Viet Nam
[b] Department of Electrical and Electronics Engineering, Universiti Teknologi PETRONAS, Malaysia
[c] Department of Power Delivery, Ho Chi Minh City University of Technology (HCMUT), 268 Ly Thuong Kiet Street, District 10, Ho Chi Minh City, Viet Nam
[d] Department of Power Systems, Ho Chi Minh City University of Technology (HCMUT), 268 Ly Thuong Kiet Street, District 10, Ho Chi Minh City, Viet Nam
[e] Vietnam National University Ho Chi Minh City, Linh Trung Ward, Thu Duc City, Ho Chi Minh City, Viet Nam
[f] FPT University, Hanoi, Viet Nam

## GRAPHICAL ABSTRACT




## ARTICLE INFO

*Article history:*
Received 27 April 2020
Received in revised form 14 April 2022

## ABSTRACT

This study proposes a new multi-objective version of the Search Group Algorithm (SGA) called the Multi-Objective Search Group Algorithm (MOSGA). The MOSGA is the combination of the conventional SGA integrated with an elitist non-dominated sorting technique, enabling it to define Pareto optimal

* Corresponding author at: Department of Power Systems, Ho Chi Minh City University of Technology (HCMUT), 268 Ly Thuong Kiet Street, District 10, Ho Chi Minh City, Viet Nam.
*E-mail address:* vndieu@hcmut.edu.vn (D.N. Vo).

https://doi.org/10.1016/j.asoc.2022.109287
1568-4946/© 2022 Published by Elsevier B.V.

Accepted 4 July 2022
Available online 11 July 2022

*Keywords:*
Engineering design problems
Multi-objective optimization
Non-dominated solution
Pareto optimal solution
Search Group Algorithm

solutions via mutation, offspring generation, and selection. The Pareto archive with a selection mechanism is used to preserve and enhance the convergence and diversity of solutions. The MOSGA is validated on twenty-five prominent case studies, including nineteen unconstrained multi-objective benchmark problems, six constrained multi-objective benchmark problems, and five multi-objective engineering design problems to validate its capability and effectiveness. The statistical results are compared to the outcomes of other well-regarded algorithms using the same performance metrics. The comparative results show that MOGSA is robust and superior in handling a wide variety of multi-objective problems.

© 2022 Published by Elsevier B.V.

## 1. Introduction

Multi-objective optimization (MOO) is a multi-criteria decision-making tool, which considers the simultaneous optimization of problems that have multiple objective functions [1]. The MOO is utilized in most disciplines such as economics, science, and engineering to make optimal decisions based on the trade-off for different objectives. The presence of MOO is of great importance in real-world problems [2]. In principle, the resolution of multi-objective problems (MOPs) leads to a set of trade-off solutions called the Pareto optimal set [3,4].

*A priori* and *a posteriori*, and *interactive* are three main classes that use the stochastic optimization algorithm to handle MOPs. In *a priori* class, multiple objectives are aggregated into a single objective [5,6]. This phase highlights the importance of each objective from the perspective of decision-makers. After the aggregation process, single-objective algorithms can be used to find the optimal solution without algorithm modifications. This approach is relatively simple, with a low computational cost. Nonetheless, *a priori* methods can lead to certain drawbacks. It requires algorithms to run many times to attain the Pareto optimal set. Moreover, these methods face difficulty in an evenly distributed set of solutions. Another disadvantage is that such methods are very sensitive to non-convex Pareto optimal fronts. In *a posteriori* class, the formulations of MOPs are retained and optimized simultaneously [7]. Hence, the algorithms need to be modified to solve multiple objectives. Such methods can obtain a set of Pareto optimal solutions in one simulation run. After the optimization, decision-making can be made. This underscores the essentiality of the diversity of solutions across all objectives to offer decision-maker with a broad range of options. These methods can deal with any kind of Pareto optimal front efficiently. The algorithms based on *a posteriori* method are highly prevalent in the literature. In *interactive* class, preferences of decision-makers are examined and integrated during the process of MOO [8]. Such methods keep the multi-objective formulation but periodically pause the optimization process and fetch the decision-making preferences. This supports algorithms not searching non-promising areas of the search domain. Nevertheless, the *interactive* method needs human involvement, making it more complicated and slower than *a priori* and *a posteriori* classes.

In 1984, the first concept of MOO using evolutionary algorithms (EAs) was proposed by David Schaffer [9]. EAs are stochastic search and optimization techniques that simulate the process of natural evolution. EAs have been recognized as being well suited to MOPs due to their features. Some of their advantages are: firstly, their population-based nature allows obtaining non-dominated set in a single run; secondly, EAs are capable of solving large and very complex search spaces, and they have low requirements for problem characteristics. The handling MOPs via EAs is called the multi-objective evolutionary algorithms (MOEAs). Over the past two decades, there was extraordinary growth in the research and application of MOEAs to solve MOPs in diverse fields such as economics, science, and engineering. According to their mechanisms, MOEAs can be divided into three main groups: Pareto-dominance-based, Decomposition-based, and Indicator-based. Pareto-based MOEAs are characterized by straightforward mechanisms, which became one of the most used strategies for achieving a relatively good approximation of the true Pareto front.

Vector Evaluated Genetic Algorithms (VEGA) is commonly known as the first Pareto-based MOEA [10]. Based on the original Genetic Algorithm (GA), VEGA was proposed to make it capable of solving MOPs. This algorithm divided the population into several subpopulations, and the number of subpopulations was equal to the number of objective functions. Each subpopulation was responsible for finding one objective. Although its concept was straightforward, non-dominated solutions obtained by VEGA were usually not distributed homogeneously along the Pareto front, especially in the trade-off regions. The literature showed that the Non-Dominated Sorting Genetic Algorithm II (NSGA-II) [11] is the most popular MOEA. Based on the well-regarded GA algorithm, NSGA-II was developed to overcome difficulties in the first version (NSGA). These difficulties are the lack of a sharing parameter, lack of considering elitism, and high computational cost of non-dominated sorting. This method used a fast non-dominated sorting approach, a diversity preservation method, and a crowded-comparison operator to alleviate the aforementioned issues. Multi-objective Particle Swarm Optimization (MOPSO) is the second most popular MOEA suggested by Coello Coello & Lechuga [12]. The Pareto dominance concept was utilized to define the flight direction of a particle. It used a mutation operator to increase randomness and improve the diversity of trial solutions. MOPSO had a high convergence rate that can be more likely to be terminated early with the incorrect Pareto front. Multi-Objective Differential Evolution (MODE) [13] was developed based on the basic DE algorithm. Generally, MODE adopted the non-dominated sorting and ranking selection methods. The non-dominated sorting was done on the combined population of parents and newly generated offspring. The aforementioned algorithms employed the concept of Pareto dominance in their mechanism. This concept provides such algorithms with practical means to handle MOPs. Since there were multiple optimum solutions in the context of MOO, most algorithms utilized an archive (or repository) to save the best solutions obtained and improve this archive during the optimization process.

In recent years, many new and effective Pareto-based MOEAs integrated with appropriate mechanisms have been endlessly proposed, namely Multi-Objective Grey Wolf Optimizer (MOGWO) [14], Multi-Objective Water Cycle Algorithm (MOWCA) [15,16], Multi-Objective Dragonfly Algorithm (MODA) [17], Multi-Objective Ant Lion Optimizer (MOALO) [18], Multi-objective Grasshopper Optimization Algorithm (MOGOA) [19], Multi-Objective Multi-Verse Optimizer (MOMVO) [20], Multiple Objective Symbiotic Organisms Search (MOSOS) [21], to name just a few. Despite the achievements in offering good solutions of these algorithms for complex MOPs, the No-Free-Lunch (NFL) [22] evidenced that no optimization method is able to deal with all MOPs effectively. Hence, there is certainly a need to improve the existing algorithms or suggest new algorithms to address a wide range of MOPs more effectively.

Recently, a new metaheuristic method, namely the Search Group Algorithm (SGA), was developed by Gonçalves et al. [23]. The mechanism of SGA is to create and develop a search group based on the potential solutions found. Its key advantage is balancing between exploration and exploitation phases to obtain feasible solutions within a search space. SGA yielded remarkable outcomes when dealing with different engineering problems, including truss structure optimization [24], automatic generation control [25], networked control systems [26], optimization of planar steel frames [27], optimal power system voltage regulation [28], and thermo-economic optimization of solar thermal systems [29,30]. There is a lot of potential for further research and exploitation of SGA as it is a fairly new and promising method.

With motivations from the above discussions, this paper proposes a new Multi-Objective Search Group Algorithm (MOSGA) for solving MOPs. The MOSGA is the modified approach to convert the SGA into an effective MOO algorithm. The main contributions of this paper may be given as follows:

- A new multi-objective method based on SGA with memeplex structure of NSGA-II is developed to solve the MOPs. SGA has the ability of high exploration and exploitation. SGA provide an exceptional balance between exploitation and exploration of the design domain. Two strategies of NSGA-II, namely the elitist non-dominated sorting approach and Pareto archive, are very effective and prominent to define and storing non-dominated solutions. The SGA search mechanism is integrated with the elitist non-dominated sorting approach and Pareto archive of NSGA-II to develop MOSGA. The aim of this incorporation strategy is to develop a robust method for finding solutions more efficiently in a multi-objective space.
- The performance of MOSGA is validated on a set of case studies with diverse features, including unconstrained and constrained multi-objective benchmark problems, and multi-objective engineering design problems. The optimal results of MOSGA are compared with those of various well-regarded MOO techniques based on different statistical analyses of five performance metrics, Wilcoxon rank-sum test, Friedman test, robustness analysis, and graphical representations of Pareto optimal fronts.
- Analysis results indicate that the MOSGA is generally successful to deal with the MOPs and obtains Pareto optimal fronts with high convergence and diversity for different benchmark test problems and multi-objective engineering problems. Comparative results highlight that MOSGA yielded very competitive results and tended to outperform other compared methods for most case studies.

Section 2 defines definitions related to this research. Section 3 presents detailed descriptions of both the SGA and the proposed MOSGA. Section 4 depicts the statistical analysis of MOSGA results in benchmark test problems. Subsequently, this section also represents the optimization results for multi-objective engineering design using the MOSGA and their statistical comparison and analysis. Lastly, Section 5 makes a conclusion for this paper.

## 2. Background

### 2.1. Multi-objective optimization

Multi-objective optimization problems (MOPs) have at least two conflicting objective functions that are minimized or maximized simultaneously. Hence, a MOP is generally stated as follows [5]:

$$\text{Find}: \quad \boldsymbol{x} = [x_1, x_2, \ldots, x_n]^T \tag{1}$$

$$\text{Minimize/ Maximize}: \quad F(\boldsymbol{x}) = [f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \ldots, f_m(\boldsymbol{x})] \tag{2}$$

$$\text{Subject to}: \quad g_j(\boldsymbol{x}) \leq 0, \quad j = 1, 2, \ldots, J \tag{3}$$

$$h_k(\boldsymbol{x}) = 0, \quad k = 1, 2, \ldots, K \tag{4}$$

$$x_i^{(L)} \leq x_i \leq x_i^{(U)}, \quad i = 1, 2, \ldots, n \tag{5}$$

where $\boldsymbol{x}$ is a solution vector of $n$ design variables, $F(\boldsymbol{x})$ is the objective vector of $m$ objective functions, the terms $g_j(\boldsymbol{x})$ denotes the $j$th inequality constraint, $h_k(\boldsymbol{x})$ denotes the $k$th equality constraints, $J$ and $K$ are the number of inequality and equality constraints, respectively, $x_i^{(L)}$ and $x_i^{(U)}$ are boundaries of the $i$th decision variable.

The single-objective optimization task achieves a single optimal solution that has the best objective function value. Meanwhile, multiple optimal solutions called Pareto optimal solutions exist in MOO. Most MOO algorithms employed the concept of domination in their search mechanism to handle multiple objectives and find Pareto optimal solutions. The concept of domination and related terms are described in the following definitions:

(a) Pareto dominance.

A solution vector $\boldsymbol{x} = [x_1, x_2,\ldots, x_m]^T$ dominates $\boldsymbol{y} = [y_1, y_2,\ldots, y_m]^T$ if the following condition is satisfied:

- $\forall i \in \{1, 2, \ldots, m\} : f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{y})$. Solution $x$ is no worse than $y$ in all objectives.
- $\exists i \in \{1, 2, \ldots, m\} : f_i(\boldsymbol{x}) < f_i(\boldsymbol{y})$. Solution $x$ is strictly better than $y$ in at least one objective.

(b) Pareto optimality.

A solution is defined as a Pareto optimal solution or non-dominated solution if no solution exists for the entire feasible search space found to dominate it, as follows:

$$\nexists \boldsymbol{y} \in \Omega | \boldsymbol{y} \prec \boldsymbol{x} \tag{6}$$

(c) Pareto optimal set.

Pareto optimal set ($PS$) is a set that comprises Pareto optimal solutions as follows:

$$PS = \{\boldsymbol{x}, \boldsymbol{y} \in \Omega | \nexists \boldsymbol{y} \prec \boldsymbol{x}\} \tag{7}$$

(d) Pareto optimal front.

Pareto optimal front ($PF$) is a set that comprises corresponding objective values of Pareto optimal set, as follows:

$$PF = \{F(\boldsymbol{x}) | \boldsymbol{x} \in PS\} \tag{8}$$

### 2.2. Elitist non-dominated sorting technique

To sort a population into different non-dominated fronts with computed crowding distance, MOSGA applies the elitist non-dominated sorting approach [11] which includes two methods: fast non-dominated sorting and crowding distance computation.

In the fast non-dominated sorting technique, two criteria are determined for each solution of the population, where domination count $n_i$ represents the number of solutions that dominate the solution $i$, and $S_i$, which is a set of solutions that is dominated by solution $i$. All solutions with a domination count $n_i$ of zero are put in the first non-dominated front. Second, for every solution $i$ with $n_i = 0$, it visits each solution $j$ in the set $S_j$ and reduces its domination count $n_j$ by one. If any solution $j$ has a domination count $n_j$ of zero, then it is put in the second front (a separate list $J$). Afterwards, the above procedure is repeated for each solution of the second front to determine the third front. This process is implemented until all fronts are achieved. Fig. 1 provides the fast non-dominated sorting method as Algorithm 1.

Crowding distance computation is then applied to preserve the diversity of solutions in a particular front. The crowding distance

Algorithm 1: Pseudocode of fast non-dominated sorting technique.

```
for each i ∈ I
    S_i = ∅
    n_i = 0
    for each j ∈ I
        if (i ≺ j) then            % If i dominates j
            S_i = S_i ∪ {j}        % Add j to the set of solutions dominated by i
        else if (j ≺ i) then
            n_i = n_i + 1          % Increment the domination counter of i
    if n_i = 0 then                % i belongs to the first non-dominated front
        i_rank = 1
        F_1 = F_1 ∪ {i}
k = 1                              % Initialize the non-dominated front counter
while F_k ≠ 0
    J = ∅                          % Used to store the members of the next non-dominated front
    for each i ∈ F_k
        for each j ∈ S_i
            n_j = n_j − 1
            if n_j = 0 then        % j belongs to the next non-dominated front
                j_rank = k + 1
                J = J ∪ {j}
    k = k + 1
    F_k = J
```

**Fig. 1.** Pseudocode of the fast non-dominated sorting method [11].

evaluates the density of solutions surrounding a particular solution in the population. Fig. 2 presents the schematic view of crowding distance computation. Firstly, the population is sorted in ascending order of the magnitude of each objective value. The boundary solutions for each objective are assigned an infinite distance value. All other intermediate solutions are assigned a crowding distance value as follows:

$$d_j^i = \sum_{j=1}^{m} \frac{f_j^{i+1} - f_j^{i-1}}{f_j^{\max} - f_j^{\min}} \tag{9}$$

where $m$ is the number of objectives, $f_j^{\min}$ and $f_j^{\max}$ denote the minimum and maximum values of the $j$th objective, respectively, $f_j^{i+1}$ and $f_j^{i-1}$ represent the $j$th objective values for two adjacent solutions ($i+1$ and $i-1$) of solution $i$, respectively. A solution located in a lesser crowded region has a higher crowding distance than others.

The crowded-comparison operator ($\prec_n$) is applied to compare two solutions in multi-objective space using two criteria: non-dominated rank ($r$) and crowding distance ($d$) as follows:

$$i \prec_n j \text{ if}(r_i < r_j) \text{ or } ((r_i = r_j) \text{ and } (d_i > d_j)) \tag{10}$$

Therefore, if two solutions belong to different non-dominated ranks, the solution of the better non-dominated rank is preferred. Otherwise, if both solutions have the same non-dominated rank, the solution with the higher crowding distance value is preferred.

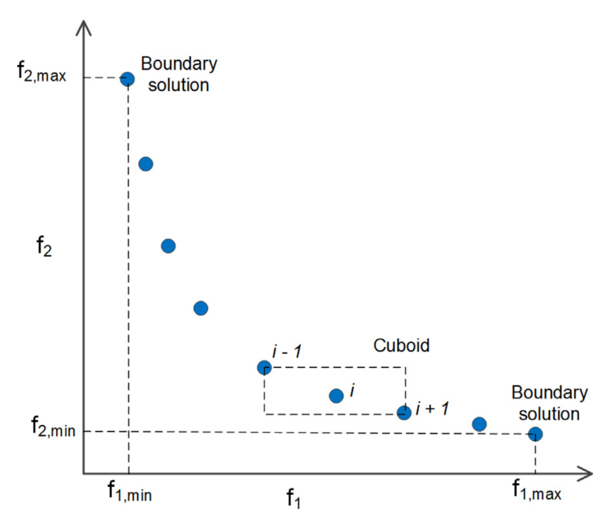



**Fig. 2.** Schematic view of crowding-distance computation [30].

### 2.3. Pareto archive selection

In MOO, an important task is to save obtained non-dominated solutions in an archive to maintain a non-dominated set. The archive is updated over the course of iterations based on a selection mechanism suggested by Deb et al. [11]. All newly generated

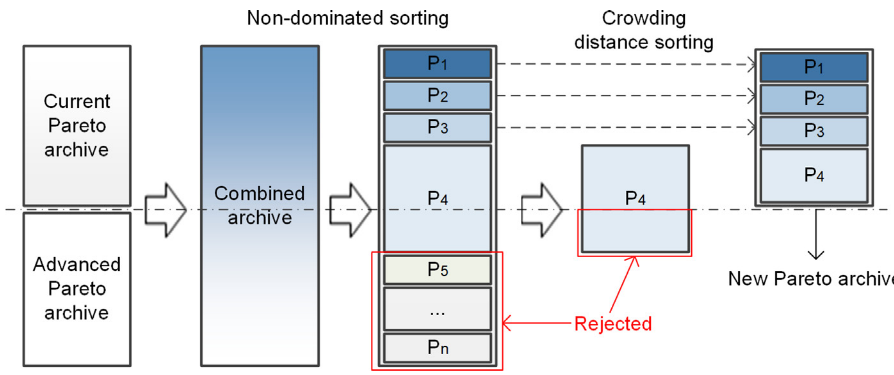



**Fig. 3.** Pareto archive selection procedure [30].

individuals are stored in an advanced Pareto archive. Two current and advanced archives are then combined, after which the combined archive size is bigger than the limit size ($n_{pop}$). To maintain the limited size of the archive, a selection mechanism is used to remove undesirable solutions and avoid the loss of potential candidate solutions. Fig. 3 presents the Pareto archive selection.

Firstly, the combined archive is ranked using fast non-dominated sorting into different non-domination fronts ($P_1$, $P_2$, …, $P_n$). The first entry, which is selected for the new Pareto archive comprised of the solutions belonging to the best non-dominated front ($P_1$). If the $P_1$ size is smaller than the limit size of the archive, all $P_1$ members will enter the new Pareto archive. Therefore, the remaining solutions for the new Pareto archive are selected from the subsequent non-dominated fronts in ranking order ($P_2, P_3, \ldots$). This process continues until the new archive has a sufficient number of fronts for $n_{pop}$ members—assuming that the front $P_k$ is the last non-dominated front, beyond which no other front can be accommodated. To choose precise solutions for the new archive, solutions from the last front ($P_k$) are chosen based on crowding distance value in descending order. The new Pareto archive of size $n_{pop}$ is used to generate a new search group in the local phase.

## 3. Multi objective search group algorithm

### 3.1. SGA

The main aim of SGA is to create a suitable balance between exploitation and exploration of the optimization process [27]. Both capabilities are important to obtain an optimum solution. The SGA optimization procedure has two phases: global and local, both phases comprising mutation, generation, and selection procedures [24]. In the global phase, the SGA forms a search group to explore potential areas, and then exploits the best individuals from these potential areas in the local phase. The SGA process is presented in the next subsections.

#### 3.1.1. Population initialization

The optimization process of the proposed SGA starts with the creation of an initial population **P** in the search space as the following equation:

$$P_{ij} = x_j^{\min} + (x_j^{\max} - x_j^{\min})U[0,1], \quad \text{for } i = 1, \ldots, n_{pop}, \quad j = 1, \ldots, n, \tag{11}$$

in which $P_{ij}$ indicates the $j$th design variable of the $i$th individual of **P**, $x_j^{\min}$ and $x_j^{\max}$ are limitation of the $j$th design variable, $U$[0,1] is a stochastic variable between a range [0,1], $n_{pop}$ represents the population size, and $n$ denotes the number of design variables.

#### 3.1.2. Selection of initial search group

After the population is initialized, the objective function value is estimated for individuals of **P**. Then, best $n_g$ individuals are selected from **P** to generate a search group **R** using a standard tournament selection. In this study, the tournament size is set to 4 for all experiments. Further details of the tournament selection can be found in the literature [31]. Search group members are ranked based on the comparison of their objective function value at every iteration.

#### 3.1.3. Mutation of search group

For enhancing the global searchability of the SGA, an inverse tournament selection is applied to select $n_{mut}$ members from **R** for mutation. Depending on the rank in the current search group, the designs having the worst objective value are more likely to be mutated. The strategy here is to create new designs away from current members' locations to explore newer areas of the design space. Mutation of each new individual is performed as follows:

$$x_j^{mut} = E[\boldsymbol{R}_{:,j}] + t\varepsilon\sigma[\boldsymbol{R}_{:,j}], \quad \text{for } j = 1, \ldots, n, \tag{12}$$

where $x_j^{mut}$ denotes the $j$th design variable of a mutated individual, $\boldsymbol{R}_{:,j}$ signifies the $j$th column search group matrix, $E$ and $\sigma$ denote the mean value and standard deviation operators, respectively, $\varepsilon$ is the convenient stochastic variable, and $t$ represents the mutation operator to control the distance for a newly created individual.

#### 3.1.4. Creation of families

Each search group member is defined as a family leader. A family is a set of family leader and individuals created by this family leader. Each family leader generates a family via perturbation by Eq. (13):

$$x_j^{new} = R_{ij} + \alpha\varepsilon, \quad \text{for } j = 1, \ldots, n, \tag{13}$$

where $\alpha$ is the perturbation constant which decreases at each iteration $k$ as follows:

$$\alpha^{k+1} = b\alpha^k \tag{14}$$

where $b$ is a parameter that is determined by a combination of the linear function.

Of note is that the perturbation constant $\alpha$ controls the SGA mechanism to explore and exploit the design domain. In first iterations, $\alpha$ is set to a value high enough to allow family leaders to create individuals in any region of the domain (exploration). SGA can explore new regions to find a global solution in a search domain. When $\alpha^k$ is gradually reduced over iterations, individuals formed by family leaders tend to stay in their neighborhood (exploitation). A second aspect worth mentioning is that better individuals create bigger families, which means that the size of each family is based on the ranking of its family leader in the current search group.

| Algorithm 2: Pseudocode of the SGA. |
|---|
| 1: Initialize the parameters of the proposed algorithm; |
| 2: Generate the initial population **P** using **Eq. (11)**; |
| 3: Create the initial search group $\mathbf{R}^k$ selecting $n_g$ individuals from the initial population using a tournament selection; |
| 4: Replace $n_{mut}$ individuals by new members created as described in **Eq. (12)**; |
| 5: Generate the families $\mathbf{F}_i$ using **Eq. (13)**; |
| 6: Select the new search group according to the rule: |
| – Global phase: search group $\mathbf{R}^{k+1}$ is formed by the best member of each family; |
| – Local phase: search group $\mathbf{R}^{k+1}$ is formed by the best $n_g$ individuals of the population. |
| 7: Update $\alpha^{k+1}$ accordingly to **Eq. (14)**; |
| 8: Make $k = k + 1$, if $k > it^{max}$, go to Step 9, otherwise return to Step 4; |
| 9: Solution found: $\boldsymbol{x}^* = \boldsymbol{R}_{1,:}$. |

**Fig. 4.** SGA's pseudocode.

### 3.1.5. Selection of a new search group

The SGA optimization procedure includes two phases: global and local. The global stage selects the best member of each family to create a new search group. Hence, the SGA explores most of the design domain to find promising regions. The local stage selects best $n_g$ individuals from all the families to create a new search group. Thus, the algorithm exploits promising regions to refine the current best solutions found. Algorithm 2 in Fig. 4 describes the SGA's pseudocode.

## 3.2. MOSGA

This study proposes a modified approach to convert the SGA into an effective MOO algorithm. The proposed search mechanism is based on the SGA where non-dominated solutions are defined through mutation, offspring generation, and selection processes. The present study integrates two new modules to develop this protocol, which is similarly used to develop NSGA-II. The first module is the elitist non-dominated sorting method, which is employed to determine the non-dominated ranks of solutions. The second module is a Pareto archive with a selection mechanism to store and maintain the diversity of non-dominated solutions obtained. This Pareto archive is also retrieved to create new search groups in the local phase of the MOSGA. These modules and the MOSGA process are presented in the next subsections.

To start the process of MOSGA, an initial population **P** is randomly created in a similar manner as that of SGA, according to Eq. (11). In initial population **P**, multiple objective functions for each individual are estimated after the initialization process. In terms of SOO, individuals of **P** are ranked based on a comparison of their objective function values. Nevertheless, in MOO, individuals of **P** are ranked into different non-dominated ranks with different crowded distance values via the elitist non-dominated sorting technique. A tournament selection is later used to select $n_g$ best individuals from **P** to generate search group **R** based on their rankings in the non-dominated ranks.

From the search group **R**, low-ranking members in non-dominated ranks are selected to be mutated. The mutation then proceeds according to Eq. (12). This step allows MOSGA to enhance the algorithm's global searchability and avoid being stuck in the local front. Unlike perturbation constant ($\alpha^k$), distance adjustment parameter ($t$) and convenient stochastic variable ($\varepsilon$) for the mutation process are not linearly reduced over the course of iterations. Parameter $\varepsilon$ provides random value in each iteration to enhance exploration not only in the first iterations but also in the last iterations. This approach is essential to avoid local front stagnation, especially in the last iterations. After the mutation process, search group members are determined as family leaders that create families via the perturbation constant. This parameter changes adaptively to turn the optimization process from exploration to exploitation. High value for $\alpha^k$ promotes extra searches across the design space (exploration), and low value encourages the search intensity in the vicinity (exploitation). This allows MOSGA to progress the convergence of solutions through iterations. The difference in feature from the one of the original SGA is that all newly generated individuals are saved in the advanced archive to be sorted later.

Finally, it is an essential stage of the MOSGA to select a new search group. It influences both convergence and diversity properties of solutions found. This process of both stages is done by using a tournament selection. To determine the best family member in the global phase, all members of each family are ranked based on the elitist non-dominated sorting technique. The best member of each family is selected to generate a new search group. The primary purpose of the global stage is to explore the entire search space. In the local phase, the best $n_g$ individuals from the Pareto archive are selected to create a new search group. This stage tends to exploit and refine the domain of the current best non-dominated solutions. In both global and local phases, selecting better individuals for a new search group enables the improvement of the convergence of solutions (the first goal of MOO). Using tournament selection, individuals from less crowded areas have a high probability of being selected to create a new search group. This helps improve the distribution and spread of less distributed regions in multi-objective search space. Hence, the diversity of MOSGA solutions (the second goal of MOO) is guaranteed.

Fig. 5 describes MOSGA's pseudocode as Algorithm 3. Its key advantages to deal with MOPs can be outlined as below:

- The elitist non-dominated sorting provides an appropriate approach to rank a population into non-dominated ranks with calculated crowded distance. Thereby, MOSGA is able to efficiently execute the next steps of its process (mutation, generation, and selection) based on ranking on non-dominated fronts.
- The mutation phase is implemented to endlessly explore new areas of the search domain. This enhances exploration capability and local minima avoidance of MOSGA.
- The adaptive conversion between exploration phase and exploitation phase is governed by the perturbation constant $\alpha^k$. Therefore, the convergence of obtained solutions is guaranteed.

**Algorithm 3: Pseudocode of the MOSGA.**

1: Initialize the parameters of the proposed algorithm;
2: Generate the initial population **P** using **Eq. (11)**;
3: Evaluate the values of multiple objective functions for each individual in the initial population **P**;
4: Sort initial population **P** into different non-domination fronts based on elitist nondominated sorting technique and store them in the Pareto archive;
5: Create the initial search group $\mathbf{R}^k$ selecting $n_g$ individuals from the initial population;
6: Replace $n_{mut}$ individuals by new members created as described in **Eq. (12)**;
7: Generate the families $\mathbf{F}_i$ using **Eq. (13)** and save all newly created individuals in the advanced Pareto archive;
8: Combine the current and advanced archives;
9: Select the best solutions for new Pareto archive based on Pareto archive selection mechanism;
10: Select the new search group according to the rule:
– Global phase: search group $\mathbf{R}^{k+1}$ is formed by the best member of each family;
– Local phase: search group $\mathbf{R}^{k+1}$ is formed by the best $n_g$ individuals from the Pareto archive.
11: Update $\alpha^{k+1}$ accordingly to **Eq. (14)**;
12: Make $k = k + 1$, if $k > it^{\max}$, go to Step 13, otherwise return to Step 6;
13: Solution found: Pareto optimal solutions in the last Pareto archive.

**Fig. 5.** MOSGA's pseudocode.

- Better families are generated by better individuals to acquire a better convergence for the optimization procedure.
- The selection of the next search group is performed based on two phases (global and local phases) to generate a satisfactory balance between exploitation and exploration capabilities.
- Individuals from less crowded regions have a higher probability of selection to form a new search group using tournament selection. MOSGA obtains a diversity of solutions based on this pattern.
- The Pareto archive efficiently saves the best non-dominated solutions found. After each iteration, the archive is updated using a selection mechanism to preserve the diversity of solutions.

## 4. Numerical examples and results

### 4.1. Experimental setup

In this subsection, twenty-five well-known multi-objective benchmark problems extracted from credible research studies are employed to validate the performance and capability of MOSGA. These problems contain objective functions with distinct features having distinctive dimensions of design variables, which are classified into two main categories:

- Unconstrained multi-objective problems: ZDT1, ZDT2, ZDT3, ZDT6, BINH1, DEB1, DEB2, DEB3, FON1, FON2, KUR, LAU, MUR, POL, SCH1, SCH2, VN1, VN2, and VN3 [2,32–41].
- Constrained multi-objective problems: BEL, BINH2, CONSTR, KITA, SRN, and TNK [5,42–46].
- Multi-objective engineering design problems: four-bar truss design problem, speed reducer design problem, disk brake design problem, welded beam design problem, and spring design problem.

Appendix A contains all mathematical formulations that are used to define multi-objective benchmark test problems. The MOSGA is developed in MATLAB programming software. The initial parameters of MOSGA, including population size ($n_{pop}$), number of search group members ($n_{pop}$), number of mutations ($n_{mut}$), perturbation constant ($\alpha^k$), and global iteration ratio ($GIR$), are selected as 100, 20, 5, 3, and 0.3, respectively, for all benchmark problems. MOSGA is compared to NSGA-II and MOPSO as two of the most prominent MOO algorithms, whereas MOMVO, MODA, and MOGOA symbolize the most recent MOO algorithms to verify the results. The MATLAB source codes of NSGA-II, MOPSO, MOMVO, MODA, and MOGOA are publicly available at MathWorks [47–51].

The population size for all MOO algorithms is set to 100. NSGA-II uses a crossover probability of 0.9, a mutation rate of $1/u$ (where $u =$ is the number of design variables), and distribution indexes for crossover and mutation operators as $\eta_c = 20$ and $\eta_m = 20$, respectively. The simulated binary crossover operator and polynomial mutation are used to generate the offspring for NSGA-II. MOPSO is run using a mutation rate of 0.5 and 30 divisions for the adaptive grid. For MOMVO, worm hole existence probability is increased linearly from 0.2 to 1. MODA is executed with Levy exponent of 1.5. For MOGOA, maximum and minimum values of decreasing coefficient are set to 1 and 0.00004, respectively.

There are clearly two goals in MOO: (1) find solutions that converge to the Pareto optimal front, and (2) maintain a diversity of solutions in the Pareto optimal set. Therefore, a set of performance metrics are needed to adequately benchmark the performance of MOO algorithms. In this paper, the term $PF_{true}$ is the true Pareto optimal front defined by functions that composed a MOP. Note that $PF_{true}$ is fixed and constant. Meanwhile, $PF_g$ signifies the term Pareto optimal front generated by a MOO algorithm. The following subsections describe the metrics used in the study.

#### 4.1.1. Generational distance

The Generational Distance (GD) indicator was proposed by Van Veldhuizen et al. [52] to assess ability of an algorithm to generate $PF_g$ that converges to the $PF_{true}$. The mathematical definition of GD indicator can be given as follows:

$$GD = \frac{\sqrt{\sum_{i=1}^{n_{pf}} {d_i}^2}}{n_{pf}} \tag{15}$$

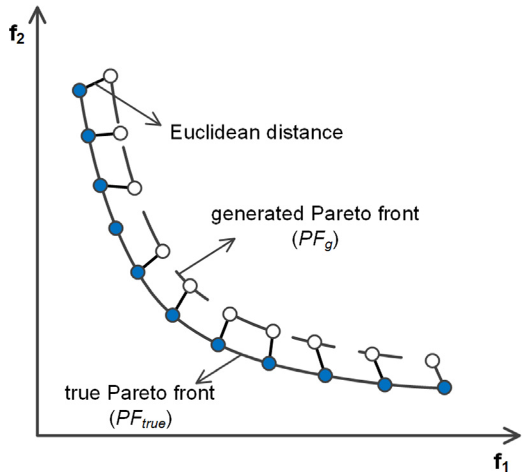



**Fig. 6.** Schematic view of the GD and IGD metrics.

where $d_i$ is the Euclidean distance between each solution in $PF_g$ and the nearest solution in $PF_{true}$ in the objective space, and $n_{pf}$ denotes the number of solutions in $PF_g$. A smaller GD value indicates a better convergence to $PF_{true}$.

*4.1.2. Inverted generational distance*

An inverse variation of the GD metric, the Inverted Generational Distance (IGD) metric, is given as follows [53]:

$$IGD = \frac{\sqrt{\sum_{i=1}^{n_t} (d_i')^2}}{n_t} \tag{16}$$

where $d_i'$ is the Euclidean distance between each solution in $PF_{true}$ and the nearest solution in $PF_g$ in the objective space, and $n_t$ denotes the number of solutions in $PF_{true}$. A smaller IGD value indicates a better convergence to $PF_{true}$. Fig. 6 depicts a schematic view of the GD and IGD metric.

*4.1.3. Spacing*

The Spacing (SP) indicator was suggested by Scott [54] to assess the distribution of solutions in $PF_g$. The mathematical equation of SP metric is:

$$SP = \sqrt{\frac{1}{n_{pf}-1}\sum_{i=1}^{n_{pf}} (d_i - \bar{d})^2} \tag{17}$$

where $d_i = \min_j(|f_1^i(x) - f_1^j(x)| - |f_2^i(x) - f_2^j(x)|),\ i, j = 1, 2, \ldots, n_{pf}$, and $\bar{d}$ is the mean value of all $d_i$. A smaller SP value shows a better distribution of solutions in $PF_g$. A schematic view of the SP indicator is depicted in Fig. 7.

*4.1.4. Spread*

The Spread ($\Delta$) indicator was suggested by Deb [5] to assess the diversity of solutions in $PF_g$. The mathematical equation of $\Delta$ metric is defined as follows:

$$\Delta = \frac{d_f + d_l + \sum_{i=1}^{n_{pf}} \left|d_i - \bar{d}\right|}{d_f + d_l + (n_{pf} - 1)\bar{d}} \tag{18}$$

where $d_i$ represents the Euclidean distance between neighboring solutions in $PF_g$, and $\bar{d}$ represents the mean value of all $d_i$, and $d_f$ and $d_l$ denote the Euclidean distances between the extreme solutions in $PF_{true}$ and $PF_g$. A schematic view of the $\Delta$ metric is portrayed in Fig. 8. A smaller $\Delta$ value indicates a better diversity (better extent of spread and distribution) of solutions in $PF_g$.

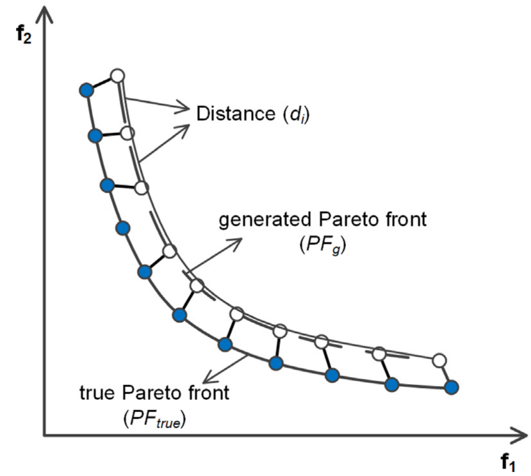



**Fig. 7.** Schematic view of the SP metric.

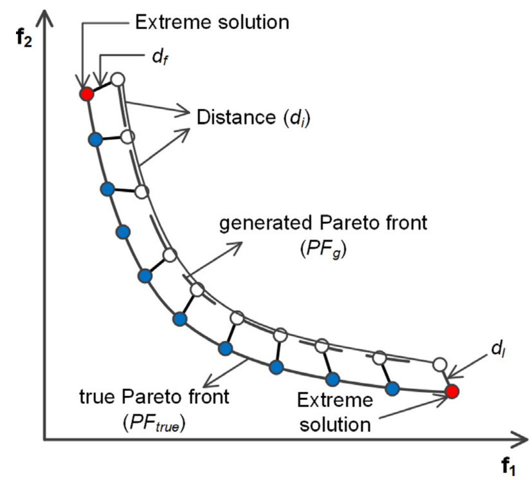



**Fig. 8.** Schematic view of the $\Delta$ metric.

The following is a calculation for the $\Delta$ metric for problems that have more than two objective functions [55]:

$$\Delta = \frac{\sum_{i=1}^{m} d(E_i, \Omega) + \sum_{X\in\Omega} |d(X, \Omega) - \bar{d}|}{\sum_{i=1}^{m} d(E_i, \Omega) + (|\Omega| - m)\bar{d}} \tag{19}$$

where $\Omega$ is a set of solutions, $E_i$ is $i$th extreme solutions in $PF_{true}$, $m$ is the number of objective functions and:

$$d(X, \Omega) = \min_{Y\in\Omega, Y\neq X} \|F(X) - F(Y)\| \tag{20}$$

$$\bar{d} = \frac{1}{|\Omega|}\sum_{X\in\Omega} d(X, \Omega) \tag{21}$$

*4.1.5. Hypervolume*

The Hypervolume indicator (HV) computes the volume covered by solutions in $PF_g$ in the objective space for a MOP where all objectives are minimized. A reference point $W$ is used to create a hypercube $v_i$ for each solution $i \in \Omega$, where solution $i$ is the diagonal corners of the hypercube $v_i$. A vector of the worst objective function values is generated to obtain the reference point. Subsequently, a union of all hypercubes is obtained, and its hypervolume can be computed by Eq. (22) [5]:

$$HV = \bigcup_{i=1}^{|\Omega|} v_i \tag{22}$$

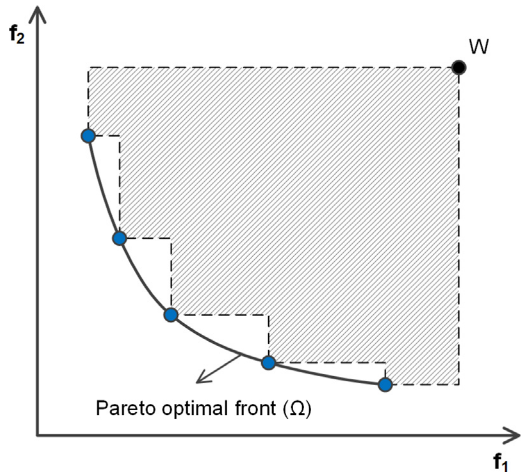



**Fig. 9.** Schematic view of the HV metric.

As shown in Fig. 9, the HV is depicted as a hatched area. Generally, a MOO algorithm with a high HV value is desirable.

In the context of benchmark function optimization, the optimization process of each method is independently run thirty times for each case study to analyze their statistical results. Moreover, the number of function evaluations (NFEs) that are considered as the stop criteria are set to 10,000 for reliable comparisons. To ensure fair and reliable comparisons, the maximum size of the Pareto archive is kept at 100 for all trials. For a comparative study, the statistical results for all considered methods are compared using four performance metrics (IGD, SP, $\Delta$, and HV). In this study, it is worth pointing out that all performance metrics are calculated in normalized objective space. Smaller values for IGD, SP, $\Delta$ and a higher value for HV show the better quality of the Pareto optimal front.

### *4.2. Multi-objective benchmark problems*

#### *4.2.1. Unconstrained multi-objective problems*

Table 1 gives the final statistical results for IGD, SP, $\Delta$, and HV metrics obtained by MOSGA, NSGA-II, MOPSO, MOMVO, MODA, and MOGOA, along with the required computational time for ZDT test suites. Table 1 indicates that MOSGA shows significant superiority over the NSGA-II, MOPSO, MOMVO, MODA, and MOGOA in terms of average and standard deviation values for the IGD metric. On the contrary, other algorithms all fail to obtain a near-optimal Pareto front with 10,000 NFEs because they are stuck in the local Pareto optimum. This is claimed by their large IGD values in Table 1. Besides, MOSGA not only obtains the best results for the IGD indicator but also ranks first for SP and $\Delta$ indicators. With respect to the HV metric, MOSGA is superior to other optimizers. Therefore, MOSGA shows the highest convergence and coverage for ZDT test suites.

Fig. 10 illustrates the graphical representations of true Pareto fronts and Pareto fronts obtained by MOSGA for ZDT test suites. All plots indicate that the Pareto optimal fronts yielded by MOSGA are distributed homogeneously along true Pareto optimal fronts.

Table 2 shows the statistical results of IGD, SP, $\Delta$, and HV indicators obtained by each algorithm for test functions BINH1, DEB1, DEB2, DEB3, FON1, and FON2. The statistical results for the IGD metric show that MOSGA outperforms other techniques, which proves that the algorithm of the present study has a better convergence on all of these test functions. It should be mentioned that the superiority of MOSGA is significant for test functions BINH1, DEB2, and FON2, which is inferred from considerable differences between the average IGD values yielded by MOSGA and those yielded by other optimizers. Moreover, NSGA-II acquires the second-best results from the IGD metric and ranks second after MOSGA for all these problems. On the contrary, MODA and MOGOA produce the worst statistical outcomes for the IGD metric.

According to Table 2, MOSGA achieves the best optimization results of SP and $\Delta$ metrics for most of these problems except for test function FON1. For test function FON1, the MOSGA has the second-best average SP values after the MOPSO; however, the best result of the $\Delta$ metric is obtained by MOSGA for this case. Additionally, MOPSO also shows competitive results for SP values compared to MOSGA for test functions BINH1 and DEB1; however, the $\Delta$ values obtained by the MOSGA are much better than those achieved by MOPSO. Hence, MOSGA offers Pareto optimal solutions that spread better than other algorithms. Meanwhile, MODA has the weakest performance of all in terms of SP and $\Delta$ metrics, which indicates that this method is unable to provide a well-distributed Pareto front. For the HV metric, MOSGA achieves better performance than other methods for most test functions.

Fig. 11 depicts graphical representations of Pareto fronts obtained by MOSGA for test functions BINH1, DEB1, DEB2, DEB3, FON1, and FON2. Fig. 11 reveals that MOSGA magnificently converges the true Pareto optimal fronts with appropriate distribution and spread for Pareto optimal solutions.

Table 3 illustrates the statistical optimization results of performance metrics for test functions KUR, LAU, MUR, POL, SCH1, and SCH2. Similar to Tables 1–2, the MOSGA in Table 3 attains the best statistical results for the IGD metric. Hence, MOSGA is able to offer superior convergence on these problems. Moreover, MOSGA, NSGA-II, and MOPSO produce competitive results in terms of the IGD metric for test functions MUR, POL, and SCH2. Meanwhile, the MODA has the weakest performance for the IGD metric in most cases. As per the outcomes of algorithms from Table 3, the MOSGA has the advantage of finding the best SP and $\Delta$ values for test functions KUR, LAU, MUR, POL, and SCH1, whereas MOPSO indicates the lowest average $\Delta$ value for test function SCH2 and the proposed algorithm proves second-best for this problem. Based on the SP and $\Delta$ metrics, the MODA has the worst statistical results, which shows that this algorithm faces difficulty in producing solutions with appropriate distribution and spread. Moreover, MOSGA outperforms other algorithms in terms of HV metric for five out of six test functions.

These observations from Table 3 can be confirmed by the graphical representations depicted in Fig. 12, from which Pareto fronts yielded by MOSGA not only converge on the true Pareto front quite well but also distribute properly. Moreover, Fig. 12 highlights that MOSGA successfully covers all parts of the true Pareto fronts uniformly for problems related to disconnected curves (i.e., test functions KUR, POL, and SCH2).

Table 4 illustrates the results of IGD, SP, $\Delta$, and HV metrics for test functions VN1, VN2, and VN3. Problems VN1, VN2, and VN3 out of twenty-five problems are selected to benchmark the performance of MOSGA to handle problems that have more than two objective functions. These problems contain three objective functions, which make them more challenging. Table 4 indicates that MOSGA provides the best performance out of all statistical metrics. This proves that MOSGA has better convergence and distribution ability than other techniques for problems with three objectives. Fig. 13 depicts the graphical representations of Pareto optimal fronts of functions VN1, VN2, and VN3 obtained by MOSGA to further confirm the above claim.

Based on the performance metrics from Tables 1–4, the present study concluded that MOSGA is the best of the six considered algorithms for dealing with unconstrained multi-objective test problems.

**Table 1**
Statistical results of MOO algorithms in terms of IGD, SP, $\Delta$, and HV metrics for ZDT test suites.

| Algorithms | IGD | | SP | | $\Delta$ | | HV | | Times |
|---|---|---|---|---|---|---|---|---|---|
| | Average | SD | Average | SD | Average | SD | Average | SD | |
| **ZDT1** | | | | | | | | | |
| MOSGA | **2.3968E−04** | **2.3485E−04** | **8.0527E−03** | **6.3937E−04** | **4.3094E−01** | **3.8630E−02** | **7.1804E−01** | 2.6305E−03 | 0.786 |
| NSGA-II | 3.8932E−02 | 3.1384E−03 | 3.9284E−02 | 1.7828E−02 | 7.4223E−01 | 3.9370E−02 | 0 | 0 | 3.452 |
| MOPSO | 2.3122E−02 | 4.9021E−03 | 3.2144E−02 | 2.6908E−02 | 8.5763E−01 | 4.3030E−02 | 7.0652E−02 | 6.9540E−02 | 1.285 |
| MOMVO | 7.3987E−03 | 2.7592E−03 | 1.3303E−02 | 4.1903E−03 | 9.1745E−01 | 5.7128E−02 | 4.7914E−01 | 4.4532E−02 | **0.679** |
| MODA | 4.9473E−02 | 6.4001E−03 | 3.8809E−02 | 1.9916E−02 | 1.1070E+00 | 7.6098E−02 | 2.0873E−04 | **1.1432E−03** | 25.356 |
| MOGOA | 3.1405E−02 | 7.0670E−03 | 1.3620E−02 | 5.1828E−03 | 1.1246E+00 | 5.2810E−02 | 1.9709E−02 | 3.3360E−02 | 89.371 |
| **ZDT2** | | | | | | | | | |
| MOSGA | **2.3260E−04** | **1.8444E−04** | **7.8326E−03** | **5.9250E−04** | **4.1976E−01** | 4.8904E−02 | **4.4190E−01** | **4.0192E−03** | 0.642 |
| NSGA-II | 7.0689E−02 | 3.7034E−03 | 5.0368E−02 | 2.1022E−02 | 8.4477E−01 | **3.9106E−02** | 0 | 0 | 6.682 |
| MOPSO | 2.3209E−02 | 5.8724E−03 | 3.5188E−02 | 6.1684E−02 | 9.5023E−01 | 9.1913E−02 | 1.1290E−02 | 2.9392E−02 | 0.681 |
| MOMVO | 1.5513E−02 | 6.9315E−03 | 2.4349E−02 | 2.3565E−02 | 1.0307E+00 | 5.5423E−02 | 1.2130E−01 | 5.7377E−02 | **0.462** |
| MODA | 8.0432E−02 | 7.6898E−03 | 2.7383E−02 | 2.5191E−02 | 1.0245E+00 | 4.1787E−02 | 0 | 0 | 26.260 |
| MOGOA | 3.3044E−02 | 7.9090E−03 | 8.4144E−03 | 2.0277E−02 | 1.0340E+00 | 5.6817E−02 | 1.6051E−03 | 6.4195E−03 | 93.922 |
| **ZDT3** | | | | | | | | | |
| MOSGA | **7.7038E−03** | **5.4142E−05** | **4.7996E−03** | **4.6090E−04** | **7.9463E−01** | **1.5661E−02** | **6.5709E−01** | **7.0740E−03** | **0.672** |
| NSGA-II | 1.6612E−02 | 1.0121E−03 | 3.0166E−02 | 1.2590E−02 | 8.0903E−01 | 1.6445E−02 | 4.7630E−02 | 1.7221E−02 | 3.531 |
| MOPSO | 1.2600E−02 | 1.8950E−03 | 2.5281E−02 | 1.1838E−02 | 9.0528E−01 | 3.8640E−02 | 1.2318E−01 | 4.7133E−02 | 1.280 |
| MOMVO | 8.6659E−03 | 1.5795E−03 | 7.0174E−03 | 1.9820E−03 | 9.8104E−01 | 4.2903E−02 | 4.0763E−01 | 5.3085E−02 | 0.695 |
| MODA | 2.1337E−02 | 2.7293E−03 | 3.1784E−02 | 1.7359E−02 | 1.1427E+00 | 8.1249E−02 | 7.3286E−03 | 1.3708E−02 | 25.417 |
| MOGOA | 1.8430E−02 | 4.9315E−03 | 9.2996E−03 | 3.9220E−03 | 1.1992E+00 | 6.8928E−02 | 5.6136E−02 | 6.2602E−02 | 90.627 |
| **ZDT6** | | | | | | | | | |
| MOSGA | **1.4593E−04** | **2.2273E−05** | **6.7931E−03** | **5.1023E−04** | **3.3362E−01** | **2.5230E−02** | **5.0462E−01** | **1.2629E−04** | 0.568 |
| NSGA-II | 1.8183E−01 | 7.3316E−03 | 5.6741E−02 | 1.6519E−02 | 8.6553E−01 | 2.7958E−02 | 0 | 0 | 7.669 |
| MOPSO | 6.0957E−02 | 3.7833E−02 | 3.3084E−01 | 2.6390E−01 | 1.0042E+00 | 7.7956E−02 | 4.4245E−02 | 7.2084E−02 | 0.663 |
| MOMVO | 4.0381E−03 | 6.2540E−03 | 2.2603E−01 | 1.6018E−01 | 1.0839E+00 | 1.2665E−01 | 3.7262E−01 | 6.7402E−02 | **0.268** |
| MODA | 1.5852E−01 | 4.8790E−02 | 1.4361E−01 | 1.7607E−01 | 1.1042E+00 | 1.2825E−01 | 3.1645E−02 | 1.2044E−01 | 13.011 |
| MOGOA | 7.4359E−02 | 6.7353E−02 | 5.1544E−02 | 5.6740E−02 | 1.0844E+00 | 7.5557E−02 | 1.4237E−01 | 1.9546E−01 | 30.793 |

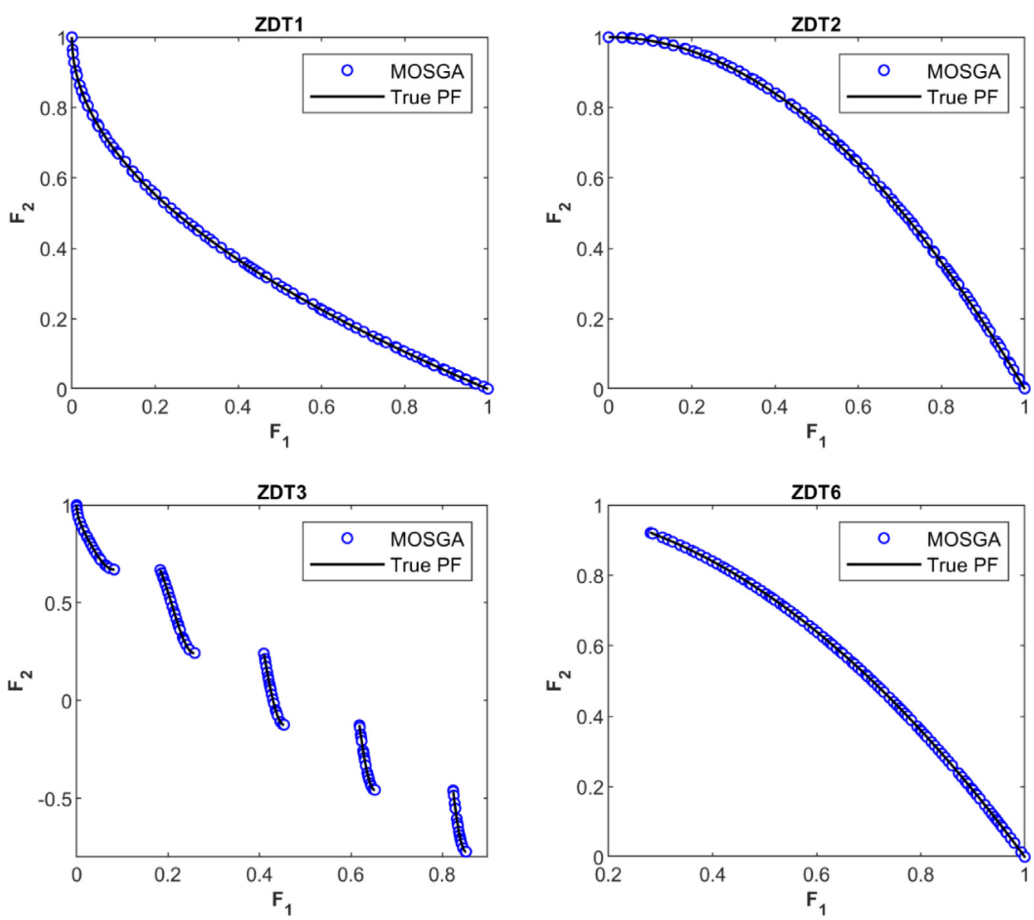



**Fig. 10.** Pareto optimal fronts generated by MOSGA for ZDT test suites.

**Table 2**
Statistical results of MOO algorithms in terms of IGD, SP, $\Delta$, and HV metrics for test functions BINH1, DEB1, DEB2, DEB3, FON1, and FON2.

| Algorithms | IGD | | SP | | $\Delta$ | | HV | | Times |
|---|---|---|---|---|---|---|---|---|---|
| | Average | SD | Average | SD | Average | SD | Average | SD | |
| **BINH1** | | | | | | | | | |
| MOSGA | **2.5152E−04** | **1.3499E−05** | **7.4010E−03** | **5.9414E−04** | **3.9194E−01** | **3.3539E−02** | **8.5862E−01** | **1.9522E−04** | **0.662** |
| NSGA-II | 2.8278E−04 | 2.1241E−05 | 9.1347E−03 | 8.0178E−04 | 4.9181E−01 | 4.1174E−02 | 8.5768E−01 | 2.4704E−04 | 2.240 |
| MOPSO | 3.2388E−04 | 6.1762E−05 | 7.6466E−03 | 1.7282E−03 | 4.3304E−01 | 4.9311E−02 | 8.5784E−01 | 5.9603E−04 | 4.226 |
| MOMVO | 8.9187E−04 | 1.7127E−04 | 1.7190E−02 | 3.4716E−03 | 1.1973E+00 | 6.1057E−02 | 8.4816E−01 | 2.8178E−03 | 1.376 |
| MODA | 1.4240E−03 | 2.9550E−04 | 1.8782E−02 | 4.6355E−03 | 1.5437E+00 | 1.7129E−01 | 8.4146E−01 | 4.1781E−03 | 5.835 |
| MOGOA | 1.1544E−03 | 1.9730E−04 | 1.5901E−02 | 4.2928E−03 | 1.5141E+00 | 3.4005E−02 | 8.4273E−01 | 3.3197E−03 | 6.746 |
| **DEB1** | | | | | | | | | |
| MOSGA | **1.4338E−04** | 1.5187E−05 | **8.1025E−03** | **6.3383E−04** | **4.0837E−01** | **2.5713E−02** | **4.4407E−01** | **1.5029E−04** | **0.603** |
| NSGA-II | 1.4746E−04 | **7.2864E−06** | 8.5035E−03 | 7.9044E−04 | 4.4693E−01 | 2.5750E−02 | 4.4394E−01 | 2.0664E−04 | 1.730 |
| MOPSO | 1.8270E−04 | 6.1820E−05 | 8.2998E−03 | 5.4773E−03 | 4.2018E−01 | 5.7188E−02 | 4.4089E−01 | 1.5742E−03 | 4.604 |
| MOMVO | 6.9269E−04 | 1.6590E−04 | 1.5712E−02 | 4.1987E−03 | 1.3954E+00 | 6.8152E−02 | 4.2443E−01 | 5.0092E−03 | 1.977 |
| MODA | 7.1969E−04 | 3.1579E−04 | 2.9839E−02 | 1.6896E−02 | 1.4522E+00 | 1.7843E−01 | 4.2110E−01 | 8.9193E−03 | 8.885 |
| MOGOA | 8.0953E−04 | 2.1193E−04 | 1.1650E−02 | 7.0603E−03 | 1.5438E+00 | 1.4359E−01 | 4.1575E−01 | 7.8046E−03 | 9.867 |
| **DEB2** | | | | | | | | | |
| MOSGA | **1.6890E−04** | **1.3785E−05** | **5.1828E−03** | **4.7572E−04** | **7.9376E−01** | **1.7862E−02** | **4.6542E−01** | **1.1679E−04** | **0.641** |
| NSGA-II | 1.9706E−04 | 2.4707E−05 | 9.8477E−03 | 1.8343E−02 | 8.7242E−01 | 1.2098E−01 | 4.6268E−01 | 2.4242E−03 | 2.436 |
| MOPSO | 1.4466E−03 | 5.3482E−04 | 3.5314E−02 | 1.6207E−02 | 1.1311E+00 | 7.4575E−02 | 4.4162E−01 | 1.1310E−02 | 1.379 |
| MOMVO | 6.6477E−04 | 1.7021E−04 | 2.6135E−02 | 5.1701E−02 | 1.2277E+00 | 6.9144E−02 | 4.5597E−01 | 3.6735E−03 | 1.074 |
| MODA | 7.0658E−04 | 2.1298E−04 | 4.4197E−02 | 6.7433E−02 | 1.3690E+00 | 7.1858E−02 | 4.5721E−01 | 3.9779E−03 | 9.210 |
| MOGOA | 3.2520E−03 | 3.3721E−03 | 1.0704E−02 | 8.0241E−03 | 1.3992E+00 | 1.5683E−01 | 4.2728E−01 | 3.6980E−02 | 8.083 |
| **DEB3** | | | | | | | | | |
| MOSGA | **1.3857E−04** | 2.6072E−05 | **7.2566E−03** | **5.9399E−04** | **4.6303E−01** | **3.6419E−02** | **2.6221E−01** | 2.5913E−04 | **0.548** |
| NSGA-II | 1.5323E−04 | **2.5987E−05** | 1.0281E−02 | 1.3927E−03 | 5.3417E−01 | 3.9427E−02 | 2.6211E−01 | **6.9920E−05** | 1.752 |
| MOPSO | 2.3636E−04 | 5.9114E−05 | 1.1009E−02 | 7.0060E−03 | 5.5729E−01 | 5.7794E−02 | 2.6201E−01 | 4.5291E−04 | 4.484 |
| MOMVO | 2.2870E−04 | 3.8984E−05 | 1.2457E−02 | 1.5442E−03 | 8.5800E−01 | 4.8636E−02 | 2.5936E−01 | 1.5937E−03 | 1.853 |
| MODA | 5.1740E−04 | 1.7061E−04 | 2.9466E−02 | 2.7458E−02 | 1.4464E+00 | 1.7607E−01 | 2.4882E−01 | 7.5396E−03 | 8.905 |
| MOGOA | 4.1660E−04 | 1.6835E−04 | 2.0047E−02 | 1.6191E−02 | 1.2084E+00 | 9.0217E−02 | 2.5350E−01 | 4.4177E−03 | 8.190 |
| **FON1** | | | | | | | | | |
| MOSGA | **3.1577E−04** | 2.0035E−05 | 6.9022E−03 | 6.1766E−04 | **3.9326E−01** | **2.8575E−02** | **2.2436E−01** | **1.5120E−04** | **0.645** |
| NSGA-II | 3.2792E−04 | **1.1396E−05** | 9.2744E−03 | **5.2335E−04** | 4.9458E−01 | 2.9845E−02 | 2.2322E−01 | 4.2397E−04 | 2.439 |
| MOPSO | 3.6524E−04 | 8.0952E−05 | **6.1671E−03** | 7.3897E−04 | 3.9686E−01 | 4.0563E−02 | 2.2334E−01 | 6.9387E−04 | 3.854 |
| MOMVO | 9.3366E−04 | 1.1693E−04 | 1.5689E−02 | 2.4918E−03 | 1.1246E+00 | 4.5227E−02 | 2.1808E−01 | 1.9490E−03 | 1.474 |
| MODA | 1.4624E−03 | 2.8309E−04 | 1.7153E−02 | 4.6914E−03 | 1.5020E+00 | 1.7215E−01 | 2.1280E−01 | 3.6453E−03 | 5.837 |
| MOGOA | 1.5761E−03 | 4.6353E−04 | 1.5433E−02 | 3.5249E−03 | 1.4809E+00 | 4.5348E−02 | 2.1159E−01 | 5.3303E−03 | 7.056 |
| **FON2** | | | | | | | | | |
| MOSGA | **1.9602E−04** | **1.2604E−05** | **6.8773E−03** | **7.1328E−04** | **4.0187E−01** | **3.6833E−02** | **4.2961E−01** | **2.5024E−04** | **0.742** |
| NSGA-II | 3.4895E−04 | 8.6231E−05 | 8.2606E−03 | 7.7427E−04 | 4.7539E−01 | 3.7910E−02 | 4.2232E−01 | 2.0742E−03 | 3.876 |
| MOPSO | 4.0872E−04 | 1.2197E−04 | 7.4465E−03 | 1.5653E−03 | 4.3766E−01 | 4.8457E−02 | 4.2350E−01 | 2.3850E−03 | 4.017 |
| MOMVO | 5.1908E−04 | 8.1980E−05 | 1.4136E−02 | 2.2334E−03 | 1.0397E+00 | 4.7523E−02 | 4.2014E−01 | 1.6817E−03 | 1.056 |
| MODA | 9.4585E−04 | 1.3947E−04 | 1.5992E−02 | 3.5337E−03 | 1.5407E+00 | 6.1461E−02 | 4.0238E−01 | 4.2837E−03 | 5.824 |
| MOGOA | 8.2475E−03 | 2.2668E−03 | 1.0124E−02 | 2.8035E−03 | 1.2847E+00 | 3.7533E−02 | 2.1763E−01 | 2.7077E−02 | 13.318 |

#### 4.2.2. Constrained multi-objective problems

Table 5 shows statistical optimization results of IGD, SP, $\Delta$, and HV metrics obtained by the MOSGA for test functions BEL, BINH2, CONSTR, KITA, SRN, and TNK. A death penalty function is applied as an approach for MOSGA to handle constraints. NSGA-II, MOPSO, MOMVO, MODA, and MOGOA are also carried out on the same test functions for comparison purposes. Based on Table 5, the MOSGA surpasses other reported algorithms to provide superior convergence on constrained multi-objective problems. Table 5 shows the superior results on average and the SD values of the IGD metric highlighted in bold. Likewise, MOSGA yields Pareto optimal solutions with the best distribution and spread. Indeed, the proposed MOSGA outperforms other algorithms as per the statistical results of SP and $\Delta$ indicators in Table 5. For the HV metric, the results obtained by MOSGA are also better than those of other methods, which shows that MOSGA has better convergence capability and robustness than others. Of the five studied algorithms, NSGA-II produces the second-best results for the IGD metric after MOSGA for most of the test functions (four out of six functions in Table 5). Nevertheless, solutions yielded by this algorithm display poor distribution and spread. This is inferred by its high values of SP and $\Delta$ metrics. On the contrary, MODA displays the worst convergence.

Fig. 14 illustrates graphical representations of Pareto fronts obtained by MOSGA on test functions BEL, BINH2, CONSTR, KITA, SRN, and TNK test functions. These figures show that some of the test problems have special Pareto optimal fronts. In particular, CONSTR is made up of a concave front connected to a linear front. Moreover, test function KITA has a continuous and concave front, whereas the front function of TNK is a discontinuous wave-like shape. Fig. 14 reveals that MOSGA is able to obtain solutions that converge fairly well on true Pareto fronts and spread the entire Pareto optimal regions evenly.

From the assessment, it is apparent that the MOSGA advantageously handles constraints and obtains Pareto optimal fronts with high convergence and distribution in different Pareto optimal regions.

#### 4.2.3. Statistical test

In this study, the Wilcoxon rank-sum test is used to validate whether the results of MOSGA are pointedly superior to those of other methods or not. A $p$-value $< 0.05$ and signed with "+" show a significant difference between two solution sets of the MOSGA and other methods. The results of the Wilcoxon rank-sum test at a significance level of 5% for all test problems for IGD, SP, $\Delta$, and HV metrics are presented in Table 6, 7, 8, and 9 , respectively.

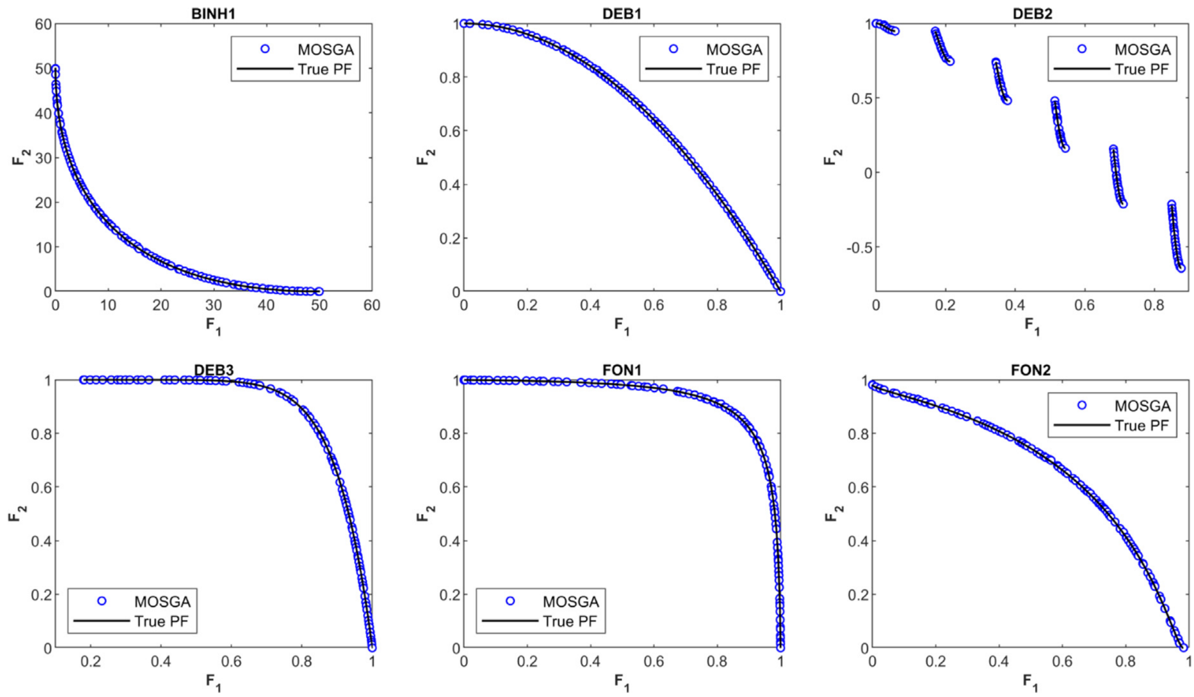



**Fig. 11.** Pareto optimal fronts generated by MOSGA for test functions BINH1, DEB1, DEB2, DEB3, FON1, and FON2.

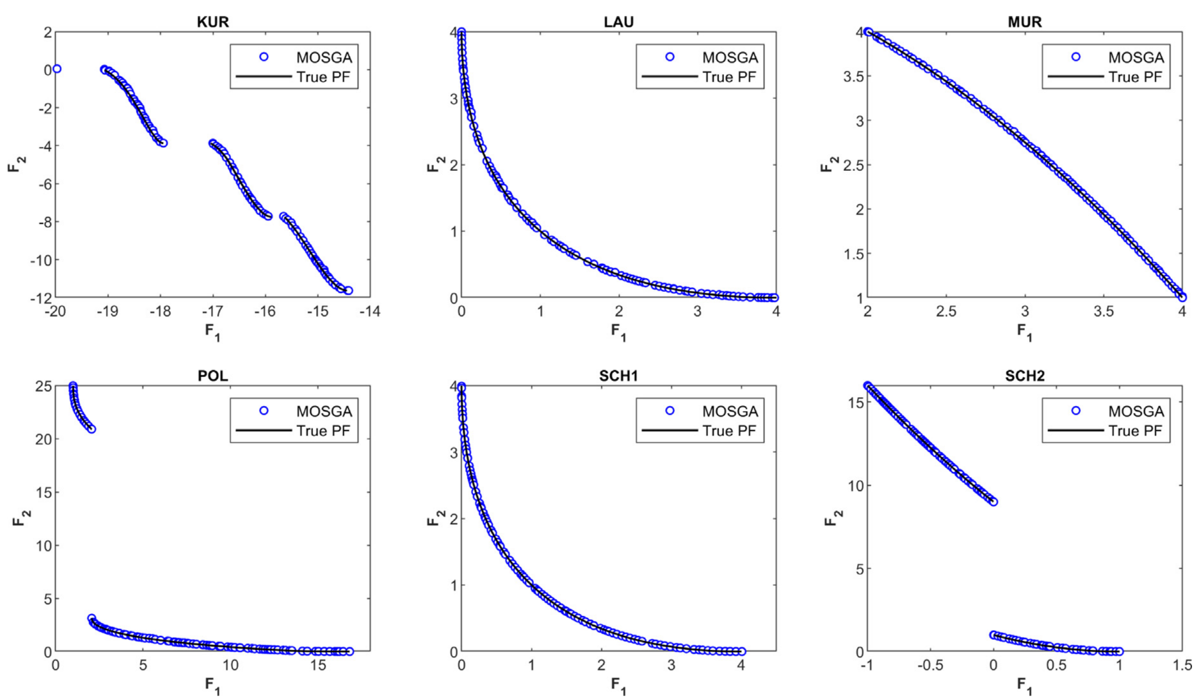



**Fig. 12.** Pareto optimal fronts generated by MOSGA for test functions KUR, LAU, MUR, POL, SCH1, and SCH2.

Table 6 shows that results of the MOSGA are statistically different from those of the NSGA-II, MOPSO, MOMVO, MODA, and MOGOA for 22, 23, 24, 25, and 25 test problems, respectively, for Wilcoxon rank-sum test in terms of IGD metric. Based on the statistical test for SP indicator, MOSGA obtains statistically different results from NSGA-II, MOPSO, MOMVO, MODA, and MOGOA for 20, 16, 23, 25, and 24, out of 25 test problems, respectively, as shown in Table 7. Regarding statistical test for $\Delta$ metric, MOSGA finds statistically different results from NSGA-II, MOPSO, MOMVO, MODA, and MOGOA for 24, 19, 25, 25, and 25 test problems, respectively. Similarly, Table 9 gives Wilcoxon rank-sum test for HV indicator, MOSGA is different from NSGA-II, MOPSO, MOMVO, MODA, and MOGOA for 23, 23, 25, 25, and 25 test problems, respectively, out of 25 test problems. Hence, MOSGA proves its pointedly superior performance in comparison with other methods for most test problems.

Another statistical test, namely the Friedman rank test, is conducted to further validate the comparative study of all six MOO algorithms. Table 10 shows the results obtained by the Friedman test based on IGD, SP, $\Delta$, and HV metrics for all benchmark problems. The best rank (with the smallest rank value) is shown in bold. As shown in Table 10, MOSGA has the best

**Table 3**
Statistical results of MOO algorithms in terms of IGD, SP, $\Delta$, and HV metrics for test functions KUR, LAU, MUR, POL, SCH1, and SCH2.

| Algorithms | IGD | | SP | | $\Delta$ | | HV | | Times |
|---|---|---|---|---|---|---|---|---|---|
| | Average | SD | Average | SD | Average | SD | Average | SD | |
| **KUR** | | | | | | | | | |
| MOSGA | **2.0447E−04** | **1.6988E−05** | **1.4701E−02** | **4.1759E−03** | **5.7899E−01** | **2.7119E−02** | **5.0106E−01** | **6.5985E−04** | **0.646** |
| NSGA-II | 2.9254E−04 | 5.9749E−05 | 1.5604E−02 | 4.7022E−03 | 7.3158E−01 | 6.1274E−02 | 4.9365E−01 | 3.5606E−03 | 3.806 |
| MOPSO | 8.4371E−04 | 4.5979E−04 | 2.0591E−02 | 1.3250E−02 | 7.7537E−01 | 1.0487E−01 | 4.7577E−01 | 8.1118E−03 | 2.198 |
| MOMVO | 4.0921E−04 | 5.9343E−05 | 1.5462E−02 | 5.6415E−03 | 9.5398E−01 | 6.1038E−02 | 4.8679E−01 | 6.2030E−03 | 0.771 |
| MODA | 2.1263E−03 | 1.1767E−03 | 2.4510E−02 | 8.7022E−03 | 1.3653E+00 | 1.2755E−01 | 4.2977E−01 | 2.2629E−02 | 5.789 |
| MOGOA | 1.8737E−03 | 1.0492E−03 | 1.6922E−02 | 7.9670E−03 | 1.3188E+00 | 4.8645E−02 | 4.1978E−01 | 3.5175E−02 | 13.238 |
| **LAU** | | | | | | | | | |
| MOSGA | **9.5302E−04** | **9.2279E−05** | **6.5310E−03** | **7.1360E−04** | **4.0842E−01** | **3.4917E−02** | **8.5758E−01** | **3.5545E−04** | **0.664** |
| NSGA-II | 3.3323E−03 | 8.1927E−04 | 6.7627E−03 | 4.9315E−03 | 1.2972E+00 | 8.4240E−02 | 8.4817E−01 | 4.6373E−03 | 5.955 |
| MOPSO | 1.8221E−03 | 5.9779E−04 | 1.0011E−02 | 4.1756E−03 | 5.1846E−01 | 4.3485E−02 | 8.5687E−01 | 7.3513E−04 | 4.278 |
| MOMVO | 1.5446E−03 | 2.4845E−04 | 1.1203E−02 | 1.7781E−03 | 7.4731E−01 | 5.5283E−02 | 8.5506E−01 | 1.0551E−03 | 0.919 |
| MODA | 5.0490E−03 | 2.4400E−03 | 1.7775E−02 | 6.8958E−03 | 1.3971E+00 | 1.8529E−01 | 8.3814E−01 | 1.1068E−02 | 5.679 |
| MOGOA | 2.5523E−03 | 4.1685E−04 | 1.6646E−02 | 3.0396E−03 | 1.1536E+00 | 5.4623E−02 | 8.4959E−01 | 1.1682E−03 | 5.966 |
| **MUR** | | | | | | | | | |
| MOSGA | **1.8128E−04** | 8.9043E−06 | **8.0259E−03** | **5.1952E−04** | **3.9104E−01** | **2.8267E−02** | **5.3516E−01** | **1.9547E−04** | **0.640** |
| NSGA-II | 1.8358E−04 | **7.7990E−06** | 8.6528E−03 | 7.0066E−04 | 4.3666E−01 | 3.8689E−02 | 5.3503E−01 | 2.7799E−04 | 1.546 |
| MOPSO | 3.1587E−04 | 5.0088E−05 | 9.7265E−03 | 4.0225E−03 | 4.8479E−01 | 5.8750E−02 | 5.2713E−01 | 1.7738E−03 | 4.052 |
| MOMVO | 8.4182E−04 | 1.9293E−04 | 1.6658E−02 | 4.4964E−03 | 1.3391E+00 | 4.3581E−02 | 5.1670E−01 | 3.7314E−03 | 1.754 |
| MODA | 1.2620E−03 | 3.8842E−04 | 3.0696E−02 | 1.2326E−02 | 1.5656E+00 | 1.0707E−01 | 5.0162E−01 | 9.6712E−03 | 11.337 |
| MOGOA | 7.8099E−04 | 1.4886E−04 | 1.1581E−02 | 3.3635E−03 | 1.3985E+00 | 1.4353E−01 | 5.1468E−01 | 5.7679E−03 | 8.595 |
| **POL** | | | | | | | | | |
| MOSGA | **1.3052E−04** | **7.9374E−06** | **4.5794E−03** | **3.6604E−04** | **8.0478E−01** | **1.8733E−02** | **9.2792E−01** | **6.4358E−05** | **0.865** |
| NSGA-II | 1.5198E−04 | 1.5288E−05 | 5.9905E−03 | 5.9112E−04 | 8.3914E−01 | 3.2257E−02 | 9.2737E−01 | 3.5953E−04 | 2.328 |
| MOPSO | 1.9091E−04 | 2.7504E−05 | 8.0079E−03 | 4.0612E−03 | 8.6933E−01 | 3.4532E−02 | 9.2650E−01 | 1.0854E−03 | 2.769 |
| MOMVO | 6.9586E−04 | 5.6720E−04 | 1.0939E−02 | 2.6868E−03 | 1.3953E+00 | 4.9008E−02 | 9.1316E−01 | 1.6602E−02 | 1.481 |
| MODA | 1.2263E−03 | 4.6091E−04 | 2.7470E−02 | 2.4552E−02 | 1.6048E+00 | 4.5320E−02 | 9.0584E−01 | 1.7893E−02 | 7.566 |
| MOGOA | 1.6161E−03 | 2.9129E−03 | 9.9781E−03 | 5.8007E−03 | 1.6035E+00 | 8.0097E−02 | 9.1013E−01 | 1.2774E−02 | 7.266 |
| **SCH1** | | | | | | | | | |
| MOSGA | **2.0733E−04** | **1.6745E−05** | **6.5812E−03** | **7.2462E−04** | **3.7374E−01** | **3.5378E−02** | **8.5862E−01** | **2.5487E−04** | 0.783 |
| NSGA-II | 5.1700E−03 | 2.3770E−03 | 2.5497E−02 | 3.1292E−02 | 1.7583E+00 | 6.9209E−02 | 7.4632E−01 | 7.3912E−02 | 3.047 |
| MOPSO | 2.5628E−04 | 4.6805E−05 | 7.1171E−03 | 1.0704E−03 | 4.1119E−01 | 3.6942E−02 | 8.5791E−01 | 4.3707E−04 | 3.853 |
| MOMVO | 3.2613E−04 | 3.2238E−05 | 9.9511E−03 | 1.0803E−03 | 7.2150E−01 | 5.5113E−02 | 8.5658E−01 | 6.6116E−04 | **0.413** |
| MODA | 3.5625E−03 | 4.5279E−03 | 4.1323E−02 | 8.9472E−02 | 1.4800E+00 | 2.3856E−01 | 7.9251E−01 | 9.6707E−02 | 3.022 |
| MOGOA | 4.4804E−04 | 4.6808E−05 | 1.2710E−02 | 1.3375E−03 | 9.5150E−01 | 5.9453E−02 | 8.5428E−01 | 8.8812E−04 | 6.096 |
| **SCH2** | | | | | | | | | |
| MOSGA | **1.4689E−04** | 1.4649E−05 | **4.9047E−03** | 5.3162E−04 | 7.2236E−01 | 3.3318E−02 | 6.7047E−01 | 1.8665E−04 | **0.700** |
| NSGA-II | 1.5642E−04 | **7.4560E−06** | 7.6413E−03 | **4.9367E−04** | 7.6321E−01 | **2.5593E−02** | 6.6978E−01 | 4.1616E−04 | 1.810 |
| MOPSO | 1.4974E−04 | 4.3907E−05 | 5.0704E−03 | 9.9953E−04 | **6.8061E−01** | 3.4977E−02 | **6.7062E−01** | **1.5612E−04** | 4.145 |
| MOMVO | 4.5866E−04 | 8.6367E−05 | 1.2191E−02 | 3.4652E−03 | 1.2505E+00 | 5.8840E−02 | 6.5888E−01 | 7.0019E−03 | 1.660 |
| MODA | 6.6544E−04 | 2.1744E−04 | 1.3919E−02 | 5.9735E−03 | 1.5175E+00 | 2.5221E−01 | 6.5147E−01 | 9.6141E−03 | 2.739 |
| MOGOA | 8.3324E−04 | 1.2944E−04 | 9.3044E−03 | 3.7307E−03 | 1.6127E+00 | 3.5297E−02 | 6.4541E−01 | 1.0222E−02 | 6.827 |

average ranking for all performance indicators, which shows that the proposed MOSGA is the most competitive algorithm of the considered algorithms.

*4.2.4. Robustness analysis and computational time*

The statistical analysis of IGD, SP, $\Delta$, and HV obtained by all algorithms for twenty-five benchmark functions are portrayed in Appendix B to evaluate the robustness of MOSGA in comparison to NSGA-II, MOPSO, MOMVO, MODA, and MOGOA for thirty independent runs. The significance of these results is represented as box plots. The rectangle box contains approximately the middle 50% of the performance metrics values, and the median value is represented as a red line. The top and bottom whiskers for each box show the boundary values, except for outliers. The outliers (exceptional values) are plotted using the (+) symbol. As shown in these figures, MOSGA yields the narrowest boxplots, which states that the range between the worst and the best values is relatively small for most test functions. Therefore, MOSGA proves superior to other MOO algorithms regarding stability and robustness.

Tables 1–5 also report the average CPU times of MOSGA, NSGA-II, MOPSO, MOMVO, MODA, and MOGOA. The proposed MOSGA has better computational time than other methods for the majority of test problems (19 out of 25). Moreover, MOSGA proves second-best for the remaining functions (i.e., test functions ZDT1, ZDT2, ZDT6, SCH1, KITA, and TNK) in terms of computational time.

*4.2.5. Comparison of MOSGA with recent methods in previous studies*

To further investigate the efficacy of the MOSGA, it is compared with some recent MOO algorithms in the literature. For fair comparisons with previous studies, population size and NFEs of MOSGA are set to 100 and 25,000, respectively. Other parameters of MOSGA are retained for same values from preceding section ($n_g = 20$, $n_{mut} = 5$, $\alpha^k = 3$, and $GIR = 0.3$).

Tables 11, 12, and 13 respectively show results of IGD, $\Delta$, and HV metrics obtained by MOSGA and other prominent methods in the past literature, including Multi-objective Evolutionary Algorithm based on Decomposition (MOEA/D) [56], Indicator-Based Evolutionary Algorithm (IBEA) [56], Multi-objective Vertex Search (MOVS) [57], and Multi-objective Artificial Algae Algorithm (MOAAA) [58] for 36 unconstrained problems. These unconstrained problems include the ZDT, WFG [59], DTLZ [60],

**Table 4**
Statistical results of MOO algorithms in terms of IGD, SP, $\Delta$, and HV metrics for test functions VN1, VN2, and VN3.

| Algorithms | IGD | | SP | | $\Delta$ | | HV | | Times |
|---|---|---|---|---|---|---|---|---|---|
| | Average | SD | Average | SD | Average | SD | Average | SD | |
| **VN1** | | | | | | | | | |
| MOSGA | **2.4026E−04** | **1.1484E−05** | **4.4173E−02** | **3.9184E−03** | **5.5688E−01** | 4.6825E−02 | **7.4916E−01** | **2.4708E−03** | **0.846** |
| NSGA-II | 2.6414E−04 | 3.1405E−05 | 6.2540E−02 | 6.1898E−03 | 8.4364E−01 | 9.0383E−02 | 7.4779E−01 | 4.3098E−03 | 1.824 |
| MOPSO | 2.7253E−04 | 2.9764E−05 | 4.8913E−02 | 1.1400E−02 | 5.7927E−01 | **4.0662E−02** | 7.2859E−01 | 8.5251E−03 | 6.938 |
| MOMVO | 4.1119E−04 | 8.9159E−05 | 4.8966E−02 | 8.2547E−03 | 9.0636E−01 | 6.6118E−02 | 7.0338E−01 | 3.4275E−02 | 2.053 |
| MODA | 3.4439E−04 | 5.8856E−05 | 7.6369E−02 | 1.4294E−02 | 1.2983E+00 | 8.1720E−02 | 7.2918E−01 | 9.0572E−03 | 7.541 |
| MOGOA | 3.8280E−04 | 5.0504E−05 | 6.0721E−02 | 1.1314E−02 | 1.2985E+00 | 8.3735E−02 | 7.1728E−01 | 1.1484E−02 | 7.107 |
| **VN2** | | | | | | | | | |
| MOSGA | **2.8526E−04** | 1.3243E−05 | **1.9369E−02** | **2.2910E−03** | **4.7504E−01** | **4.8070E−02** | **9.3451E−01** | **4.4318E−04** | **0.936** |
| NSGA-II | 2.9220E−04 | **1.0323E−05** | 2.7947E−02 | 3.2063E−03 | 9.7279E−01 | 8.1007E−02 | 9.3244E−01 | 1.0103E−03 | 2.158 |
| MOPSO | 5.8519E−04 | 8.9171E−05 | 2.7308E−02 | 1.0592E−02 | 5.0744E−01 | 9.4796E−02 | 9.0076E−01 | 1.2648E−02 | 4.477 |
| MOMVO | 3.8813E−04 | 6.0734E−05 | 3.2634E−02 | 6.5127E−03 | 1.0521E+00 | 6.1625E−02 | 9.2746E−01 | 3.7965E−03 | 1.516 |
| MODA | 5.4881E−04 | 9.5762E−05 | 5.6247E−02 | 2.8214E−02 | 1.4305E+00 | 1.2322E−01 | 9.2012E−01 | 5.3337E−03 | 6.052 |
| MOGOA | 4.7571E−04 | 5.6858E−05 | 3.2343E−02 | 9.0464E−03 | 1.3044E+00 | 9.1402E−02 | 9.2343E−01 | 3.2122E−03 | 7.156 |
| **VN3** | | | | | | | | | |
| MOSGA | **1.8974E−04** | **1.7277E−05** | **1.5378E−02** | 1.8929E−03 | **4.4079E−01** | **3.2831E−02** | **8.6225E−01** | **1.7509E−04** | **0.975** |
| NSGA-II | 2.0745E−04 | 5.2902E−05 | 1.6582E−02 | **1.6418E−03** | 7.2965E−01 | 9.0953E−02 | 8.5996E−01 | 5.2844E−04 | 2.000 |
| MOPSO | 2.3204E−04 | 7.3378E−05 | 1.5992E−02 | 3.1391E−03 | 4.5246E−01 | 5.4386E−02 | 8.6207E−01 | 6.9021E−04 | 3.827 |
| MOMVO | 9.4343E−04 | 1.0404E−03 | 2.5820E−02 | 5.3427E−03 | 1.0563E+00 | 6.9225E−02 | 8.5168E−01 | 4.9058E−03 | 1.387 |
| MODA | 4.4523E−04 | 1.2796E−04 | 3.4876E−02 | 9.8977E−03 | 1.3666E+00 | 1.0440E−01 | 8.4082E−01 | 7.0928E−03 | 6.853 |
| MOGOA | 7.0605E−04 | 2.9969E−04 | 2.5044E−02 | 5.6400E−03 | 1.3855E+00 | 6.2668E−02 | 8.3922E−01 | 7.2207E−03 | 7.614 |

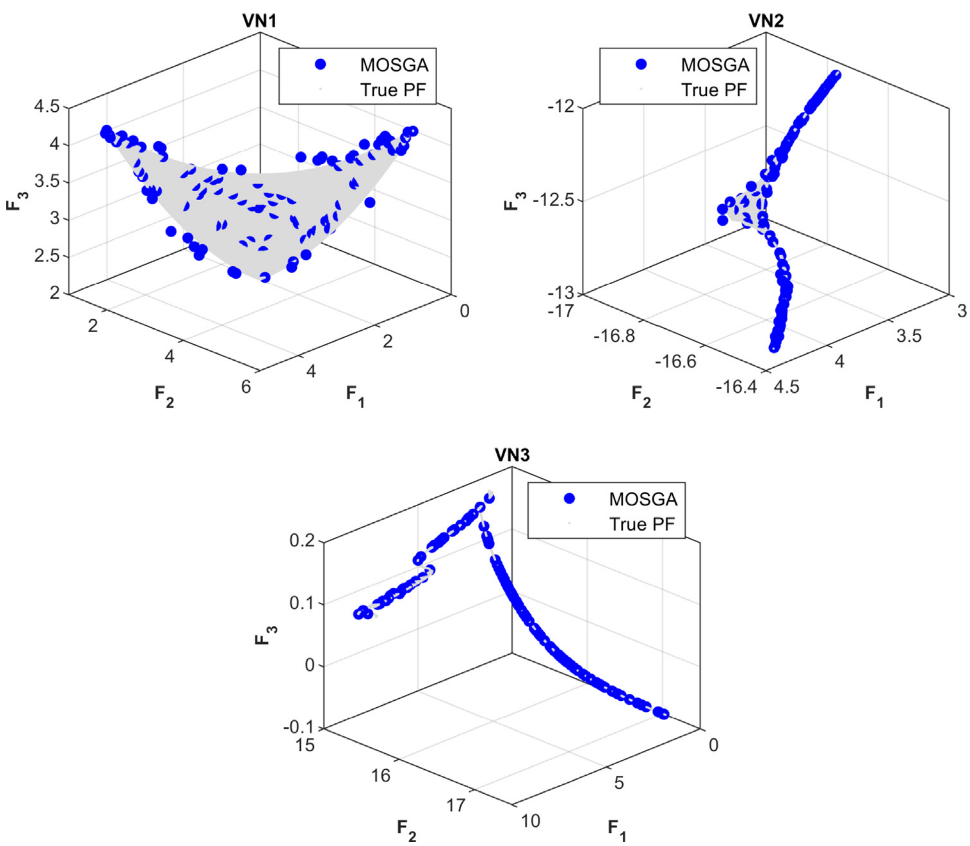



**Fig. 13.** Pareto optimal fronts generated by MOSGA for test functions VN1, VN2, and VN3.

LZ09 [61] test suites, OKA1, OKA2 [62], VN2, VN3, SCH1, FON2, KUR, and POL. In these test functions, 27 test problems contain two objectives and 9 test problems contain three objectives. Inspecting the results in Table 11, MOSGA outperforms other algorithms for IGD metric for 28 out of 36 test functions (i.e., 77.78%). For $\Delta$ metric, MOSGA produces the best results for 15 out of 36 problems. Furthermore, MOSGA shows very competitive results in the rest of the test problems. Table 13 also shows that MOSGA obtains pointedly better results than IBEA, MOEA/D, MOVS, and MOAAA for most test problems regarding the HV metric. Therefore, it is evident that MOSGA yields very competitive results and tends to outperform IBEA, MOEA/D, MOVS, and MOAAA for these test problems. Pareto fronts obtained by MOSGA for WFG and DTLZ test suites are illustrated in Appendix C. It can be observed

**Table 5**
Statistical results of MOO algorithms in terms of IGD, SP, $\Delta$, and HV metrics for test functions BEL, BINH2, CONSTR, KITA, SRN, and TNK.

| Algorithms | IGD | | SP | | $\Delta$ | | HV | | Times |
|---|---|---|---|---|---|---|---|---|---|
| | Average | SD | Average | SD | Average | SD | Average | SD | |
| **BEL** | | | | | | | | | |
| MOSGA | **1.8438E−04** | 2.3824E−05 | **7.8350E−03** | **5.4055E−04** | **3.8809E−01** | **2.7503E−02** | **5.8139E−01** | 3.4228E−04 | **1.023** |
| NSGA-II | 1.8662E−04 | **5.2104E−06** | 8.3841E−03 | 7.0896E−04 | 4.4638E−01 | 2.9484E−02 | 5.8078E−01 | **2.9727E−04** | 2.808 |
| MOPSO | 2.6993E−04 | 2.8677E−05 | 8.0553E−03 | 9.4557E−04 | 4.1962E−01 | 3.4954E−02 | 5.7508E−01 | 1.3858E−03 | 4.273 |
| MOMVO | 1.0343E−03 | 2.0894E−04 | 1.4691E−02 | 4.7037E−03 | 1.4242E+00 | 4.4463E−02 | 5.5596E−01 | 5.6070E−03 | 2.320 |
| MODA | 1.3265E−03 | 5.4157E−04 | 3.1296E−02 | 9.5831E−03 | 1.5608E+00 | 1.1574E−01 | 5.4768E−01 | 1.0595E−02 | 9.613 |
| MOGOA | 3.6936E−04 | 5.8797E−05 | 1.1063E−02 | 1.7810E−03 | 8.6272E−01 | 1.0324E−01 | 5.7508E−01 | 2.2390E−03 | 8.799 |
| **BINH2** | | | | | | | | | |
| MOSGA | **7.0076E−05** | 5.7899E−06 | 8.3983E−03 | 6.8836E−04 | **4.3699E−01** | 3.2655E−02 | **8.4313E−01** | **1.3995E−04** | **0.989** |
| NSGA-II | 7.1629E−05 | **3.4167E−06** | 7.9894E−03 | **6.6989E−04** | 4.4510E−01 | **2.2584E−02** | 8.4243E−01 | 2.3863E−04 | 2.289 |
| MOPSO | 1.1201E−04 | 3.2452E−05 | **7.4839E−03** | 1.8446E−03 | 4.4558E−01 | 3.6903E−02 | 8.4157E−01 | 4.5709E−04 | 4.131 |
| MOMVO | 2.7582E−04 | 4.2267E−05 | 1.5480E−02 | 2.6545E−03 | 1.3184E+00 | 5.3066E−02 | 8.2780E−01 | 2.8513E−03 | 1.973 |
| MODA | 3.6665E−04 | 1.4300E−04 | 2.3740E−02 | 7.3140E−03 | 1.5296E+00 | 1.3206E−01 | 8.2418E−01 | 6.9513E−03 | 6.894 |
| MOGOA | 2.6762E−04 | 7.9572E−05 | 1.1745E−02 | 3.7379E−03 | 1.3234E+00 | 2.3755E−01 | 8.2884E−01 | 5.9038E−03 | 8.385 |
| **CONSTR** | | | | | | | | | |
| MOSGA | **2.0530E−04** | **7.0184E−06** | **7.9547E−03** | **4.1069E−04** | **4.2636E−01** | **1.8244E−02** | **8.1439E−01** | **2.6238E−04** | **0.878** |
| NSGA-II | 3.2425E−04 | 5.3987E−05 | 9.2990E−03 | 1.4418E−03 | 7.0026E−01 | 3.4383E−02 | 8.0688E−01 | 1.6227E−03 | 3.599 |
| MOPSO | 4.8622E−04 | 1.6302E−04 | 1.2381E−02 | 3.5422E−03 | 6.4254E−01 | 9.0661E−02 | 8.0848E−01 | 2.1126E−03 | 2.969 |
| MOMVO | 5.5924E−04 | 1.0518E−04 | 1.4872E−02 | 2.4114E−03 | 1.1003E+00 | 5.8542E−02 | 8.0437E−01 | 2.9763E−03 | 1.650 |
| MODA | 6.3363E−04 | 1.7059E−04 | 1.9968E−02 | 4.8989E−03 | 1.0765E+00 | 1.2320E−01 | 7.9454E−01 | 6.3596E−03 | 6.608 |
| MOGOA | 1.8417E−03 | 8.1807E−04 | 1.0627E−02 | 5.2680E−03 | 1.5581E+00 | 5.7912E−02 | 7.7780E−01 | 1.0012E−02 | 8.278 |
| **KITA** | | | | | | | | | |
| MOSGA | **2.1913E−04** | **9.2652E−06** | **8.0256E−03** | 4.4398E−03 | **3.7540E−01** | 5.9872E−02 | **7.1075E−01** | **3.5143E−04** | 1.186 |
| NSGA-II | 8.7909E−04 | 1.8530E−04 | 1.2942E−02 | 2.7899E−02 | 1.0514E+00 | 8.6308E−02 | 6.8619E−01 | 5.4679E−03 | 4.154 |
| MOPSO | 6.4165E−04 | 1.8884E−04 | 3.0230E−02 | 2.0007E−02 | 6.9790E−01 | 1.2307E−01 | 7.0181E−01 | 4.1270E−03 | 3.320 |
| MOMVO | 4.2979E−04 | 5.7549E−05 | 1.3318E−02 | **3.7902E−03** | 7.2277E−01 | **4.8203E−02** | 7.0327E−01 | 1.5082E−03 | **0.994** |
| MODA | 2.8172E−03 | 1.0178E−03 | 8.5146E−02 | 5.7922E−02 | 1.3596E+00 | 2.2269E−01 | 6.2512E−01 | 3.2756E−02 | 5.906 |
| MOGOA | 2.3397E−03 | 8.3664E−04 | 1.4170E−02 | 1.6730E−02 | 1.4058E+00 | 1.2099E−01 | 6.4141E−01 | 2.6602E−02 | 6.974 |
| **SRN** | | | | | | | | | |
| MOSGA | **9.4943E−05** | 7.1727E−06 | **6.7433E−03** | 5.6370E−04 | **3.8925E−01** | 3.1421E−02 | **6.1867E−01** | **3.1657E−04** | **1.178** |
| NSGA-II | 9.6368E−05 | **3.7905E−06** | 8.2943E−03 | **5.5062E−04** | 4.3954E−01 | **2.5944E−02** | 6.1787E−01 | 4.5630E−04 | 4.216 |
| MOPSO | 1.2189E−04 | 1.6953E−05 | 7.8738E−03 | 1.5279E−03 | 4.3286E−01 | 3.6736E−02 | 6.1600E−01 | 9.8100E−04 | 4.349 |
| MOMVO | 3.6465E−04 | 8.7014E−05 | 1.3957E−02 | 2.6721E−03 | 1.2243E+00 | 7.3563E−02 | 6.0166E−01 | 5.0320E−03 | 1.878 |
| MODA | 5.5064E−04 | 7.5997E−05 | 1.8192E−02 | 5.8258E−03 | 1.5869E+00 | 3.9668E−02 | 5.8327E−01 | 5.0529E−03 | 7.422 |
| MOGOA | 4.8430E−04 | 7.1627E−05 | 1.6728E−02 | 4.8290E−03 | 1.5369E+00 | 4.8583E−02 | 5.9319E−01 | 4.9442E−03 | 8.281 |
| **TNK** | | | | | | | | | |
| MOSGA | **2.2510E−04** | **1.3403E−05** | **5.1168E−03** | **5.2752E−04** | **7.4258E−01** | **2.7507E−02** | **4.2484E−01** | **2.9818E−04** | 1.503 |
| NSGA-II | 1.8469E−03 | 6.7056E−04 | 7.1883E−03 | 5.9216E−03 | 1.6468E+00 | 5.7394E−02 | 3.9733E−01 | 5.1512E−03 | 8.835 |
| MOPSO | 1.6227E−03 | 5.5813E−04 | 1.8194E−02 | 7.5789E−03 | 9.3330E−01 | 5.0875E−02 | 3.9556E−01 | 7.5622E−03 | 2.379 |
| MOMVO | 4.6255E−04 | 6.9991E−05 | 9.0356E−03 | 1.5071E−03 | 9.2842E−01 | 4.1290E−02 | 4.1865E−01 | 1.9299E−03 | **0.771** |
| MODA | 3.3969E−03 | 1.0549E−03 | 5.9529E−02 | 4.7971E−02 | 1.1082E+00 | 2.6983E−01 | 3.5735E−01 | 1.6639E−02 | 6.161 |
| MOGOA | 2.6375E−03 | 1.0838E−03 | 6.2072E−03 | 2.0918E−03 | 1.3896E+00 | 9.2119E−02 | 3.9425E−01 | 1.2420E−02 | 6.657 |

that MOSGA fruitfully converges on the true Pareto fronts with a high diversity of solutions.

Table 14 presents comparisons of MOSGA and other methods in the literature, including Micro-Genetic Algorithm (Micro-GA), Pareto Archived Evolution Strategy (PAES), Multi-Objective Cuckoo Search (MOCS), Constrained Multi-Objective Particle Swarm Optimization (CMOPSO), MOWCA [15], MOALO [18], Multi-Objective Colliding Bodies Optimization (MOCBO), MOSOS [21], Strength Pareto Evolutionary Algorithm 2 (SPEA2), Multi-Objective Self-Adaptive Differential Evolution (MOSADE) [55], and Non-Dominated Sorting Grey Wolf Optimizer (NS-GWO) [63] for some constrained test functions. It is obviously noted from Table 14 that MOSGA also outperforms other MOO algorithms in the previous studies for the majority of test problems, especially for the GD metric.

### 4.3. Multi-objective engineering design problems

Although multi-objective benchmark test functions are very effective in testing an algorithm, dealing with real-world optimization problems is always more challenging. To assess its applicability, the present study further examines the proposed MOSGA with five multi-objective engineering design problems. Mathematical formulations of engineering design problems are given in Appendix D. Similar to the previous section, the MOSGA is independently run thirty times for each problem, and each run consisted of 10,000 NFEs. The optimization results of MOSGA are compared with outcomes from NSGA-II, MOPSO, MOMVO, MODA, and MOGOA. The control parameters of these algorithms and the size of the Pareto archive are kept the same to test the benchmark problems. Since true Pareto optimal fronts for engineering design problems are unavailable in the literature, six considered MOO algorithms are compared using SP and HV metrics.

#### 4.3.1. Four-bar truss design problem

Table 15 gives the statistical results of SP and HV indicators produced by each considered algorithm. Based on the average SP values in Table 15, MOSGA shows superior performance to maintain a suitable distribution of the obtained Pareto optimal solutions. In addition to achieving the best SP value, MOSGA has the advantage of finding a wide range of solutions having uniform spread that is shown in the highest HV values obtained by MOSGA. Moreover, the lower SD values for the SP and HV metrics indicate better stability of solutions generated by MOSGA compared with the other optimizers.

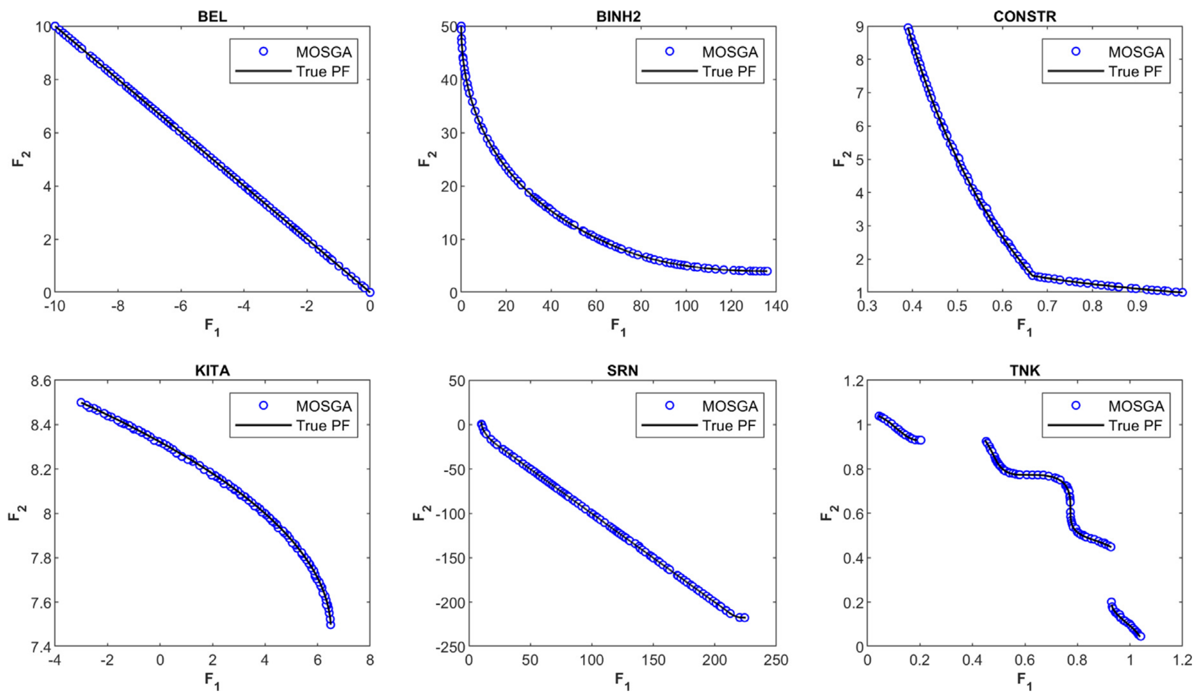



**Fig. 14.** Pareto optimal fronts generated by MOSGA for test functions BEL, BINH2, CONSTR, KITA, SRN, and TNK.

**Table 6**
Results of Wilcoxon rank-sum test of MOSGA versus other algorithms based on IGD metric.

| MOSGA versus | NSGA-II | | MOPSO | | MOMVO | | MODA | | MOGOA | |
|---|---|---|---|---|---|---|---|---|---|---|
| | p-value | Signed | p-value | Signed | p-value | Signed | p-value | Signed | p-value | Signed |
| ZDT1 | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| ZDT2 | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| ZDT3 | 3.02E−11 | + | 3.02E−11 | + | 1.96E−01 | – | 3.02E−11 | + | 3.02E−11 | + |
| ZDT6 | 3.02E−11 | + | 3.01E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| BINH1 | 1.31E−08 | + | 2.61E−10 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| DEB1 | 3.51E−02 | + | 8.88E−06 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| DEB2 | 8.84E−07 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| DEB3 | 1.12E−02 | + | 2.67E−09 | + | 3.16E−10 | + | 3.02E−11 | + | 3.02E−11 | + |
| FON1 | 2.53E−04 | + | 7.30E−04 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| FON2 | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| KUR | 1.61E−10 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| LAU | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| MUR | 4.83E−01 | – | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| POL | 1.70E−08 | + | 4.08E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| SCH1 | 3.02E−11 | + | 1.73E−07 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| SCH2 | 1.77E−03 | + | 5.37E−02 | – | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| VN1 | 3.37E−04 | + | 9.83E−08 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| VN2 | 1.54E−01 | – | 3.02E−11 | + | 3.69E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| VN3 | 4.55E−01 | – | 5.01E−02 | – | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| BEL | 7.30E−04 | + | 3.16E−10 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| BINH2 | 3.92E−02 | + | 6.70E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| CONSTR | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| KITA | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| SRN | 2.61E−02 | + | 6.72E−10 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| TNK | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |

Fig. 15 depicts a graphical representation of Pareto optimal front generated by MOSGA, in which extreme solutions obtained by MOSGA are (1727.7393, 0.0027) and (1174.1999, 0.0341). This Pareto front obtained by MOSGA for this problem is very consistent with the Pareto optimal front available in the literature [15].

### 4.3.2. Speed reducer design problem

Table 16 presents a comparative result of SP and HV indicators obtained by MOSGA, NSGA-II, MOPSO, MOMVO, MODA, and MOGOA. The statistical results in Table 16 denote that the average SP value obtained by MOSGA ranks first. Therefore, MOSGA finds a more evenly distributed Pareto optimal front compared with NSGA-II, MOPSO, MOMVO, MODA, and MOGOA. Likewise, MOSGA also surpasses the other methods for HV indicators. Thus, MOSGA offers a diverse set of Pareto optimal solutions having a good spread.

Moreover, Fig. 15 depicts the graphical representation of the Pareto optimal front generated by MOSGA, which clarifies the results presented in Table 16. The extreme Pareto solutions

**Table 7**
Results of Wilcoxon rank-sum test of MOSGA versus other algorithms based on SP metric.

| MOSGA versus | NSGA-II | | MOPSO | | MOMVO | | MODA | | MOGOA | |
|---|---|---|---|---|---|---|---|---|---|---|
| | p-value | Signed | p-value | Signed | p-value | Signed | p-value | Signed | p-value | Signed |
| ZDT1 | 3.02E−11 | + | 3.02E−11 | + | 3.01E−07 | + | 3.02E−11 | + | 9.06E−08 | + |
| ZDT2 | 3.02E−11 | + | 6.55E−04 | + | 8.77E−02 | – | 1.02E−05 | + | 3.16E−05 | + |
| ZDT3 | 3.02E−11 | + | 3.02E−11 | + | 2.03E−07 | + | 3.02E−11 | + | 2.92E−09 | + |
| ZDT6 | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 9.76E−10 | + | 1.29E−09 | + |
| BINH1 | 5.57E−10 | + | 6.84E−01 | – | 3.02E−11 | + | 3.02E−11 | + | 5.07E−10 | + |
| DEB1 | 1.44E−02 | + | 3.01E−04 | + | 1.07E−09 | + | 3.34E−11 | + | 2.61E−02 | + |
| DEB2 | 2.15E−10 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 5.61E−05 | + |
| DEB3 | 3.34E−11 | + | 8.50E−02 | – | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| FON1 | 3.02E−11 | + | 2.01E−04 | + | 3.02E−11 | + | 3.02E−11 | + | 1.96E−10 | + |
| FON2 | 5.53E−08 | + | 8.77E−02 | – | 3.02E−11 | + | 5.57E−10 | + | 1.29E−06 | + |
| KUR | 1.67E−01 | – | 6.67E−03 | + | 8.88E−01 | – | 4.80E−07 | + | 9.82E−01 | – |
| LAU | 3.55E−01 | – | 2.49E−06 | + | 3.34E−11 | + | 5.57E−10 | + | 3.02E−11 | + |
| MUR | 2.01E−04 | + | 9.23E−01 | – | 3.02E−11 | + | 3.02E−11 | + | 1.16E−07 | + |
| POL | 3.16E−10 | + | 1.89E−04 | + | 3.02E−11 | + | 3.02E−11 | + | 8.29E−06 | + |
| SCH1 | 1.68E−03 | + | 8.50E−02 | – | 3.69E−11 | + | 3.08E−08 | + | 3.02E−11 | + |
| SCH2 | 3.02E−11 | + | 8.53E−01 | – | 3.02E−11 | + | 6.72E−10 | + | 2.68E−06 | + |
| VN1 | 3.69E−11 | + | 1.15E−01 | – | 8.31E−03 | + | 3.02E−11 | + | 1.29E−09 | + |
| VN2 | 1.09E−10 | + | 1.86E−03 | + | 8.99E−11 | + | 2.87E−10 | + | 2.03E−09 | + |
| VN3 | 3.78E−02 | + | 7.62E−01 | – | 1.33E−10 | + | 3.02E−11 | + | 8.89E−10 | + |
| BEL | 3.50E−03 | + | 7.73E−01 | – | 1.56E−08 | + | 3.02E−11 | + | 1.07E−09 | + |
| BINH2 | 5.75E−02 | – | 2.27E−03 | + | 3.02E−11 | + | 3.02E−11 | + | 4.64E−05 | + |
| CONSTR | 3.52E−07 | + | 1.85E−08 | + | 3.02E−11 | + | 3.02E−11 | + | 2.42E−02 | + |
| KITA | 7.84E−01 | – | 2.87E−10 | + | 1.55E−09 | + | 4.50E−11 | + | 1.17E−03 | + |
| SRN | 1.96E−10 | + | 5.26E−04 | + | 3.02E−11 | + | 3.02E−11 | + | 4.50E−11 | + |
| TNK | 1.19E−01 | – | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 1.33E−02 | + |

**Table 8**
Results of Wilcoxon rank-sum test of MOSGA versus other algorithms based on $\Delta$ metric.

| MOSGA versus | NSGA-II | | MOPSO | | MOMVO | | MODA | | MOGOA | |
|---|---|---|---|---|---|---|---|---|---|---|
| | p-value | Signed | p-value | Signed | p-value | Signed | p-value | Signed | p-value | Signed |
| ZDT1 | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| ZDT2 | 3.02E−11 | + | 3.02E−11 | + | 3.00E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| ZDT3 | 3.18E−03 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| ZDT6 | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| BINH1 | 8.89E−10 | + | 5.87E−04 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| DEB1 | 2.00E−06 | + | 8.30E−01 | – | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| DEB2 | 3.35E−08 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| DEB3 | 2.60E−08 | + | 2.03E−09 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| FON1 | 9.92E−11 | + | 3.04E−01 | – | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| FON2 | 2.39E−08 | + | 1.95E−03 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| KUR | 6.07E−11 | + | 1.78E−10 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| LAU | 3.02E−11 | + | 3.82E−10 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| MUR | 1.17E−05 | + | 1.86E−09 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| POL | 2.13E−05 | + | 8.89E−10 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| SCH1 | 3.02E−11 | + | 3.56E−04 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| SCH2 | 5.46E−06 | + | 4.08E−05 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| VN1 | 3.02E−11 | + | 1.02E−01 | – | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| VN2 | 3.02E−11 | + | 3.48E−01 | – | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| VN3 | 5.49E−11 | + | 4.38E−01 | – | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| BEL | 7.77E−09 | + | 6.20E−04 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| BINH2 | 2.71E−01 | – | 4.12E−01 | – | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| CONSTR | 3.02E−11 | + | 3.34E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| KITA | 3.02E−11 | + | 5.49E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| SRN | 4.11E−07 | + | 2.60E−05 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| TNK | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 2.00E−06 | + | 3.02E−11 | + |

yielded by MOSGA are (6024.9005, 694.7161) and (3008.0756, 1076.2886).

#### 4.3.3. Disk brake design problem

Table 17 provides final statistical results for SP and HV metrics attained by each considered optimizer. Table 17 shows that MOSGA proves its superiority over NSGA-II, MOPSO, MOMVO, MODA, and MOGOA for the SP metric. Table 17 reveals that there is a considerable difference between the average SP values yielded by MOSGA and MOGOA as the second-best techniques with a corresponding value of 7.3538E−03 and 1.2720E−02, respectively. Thus, the solutions of MOSGA prove a better distribution than those of other algorithms. Additionally, the average HV values produced by NSGA-II, MOPSO, MOMVO, MODA, and MOGOA are not as good as the ones of MOSGA. Since HV is a criterion to assess the convergence and diversity of a MOO technique, these results indicate that MOSGA has more excellent search performance and obtains better Pareto optimal fronts.

Fig. 15 portrays the Pareto optimal front obtained by MOSGA. As a result, MOSGA finds extreme solutions with values between (2.7794, 2.0751) and (0.1274, 16.6549). The obtained Pareto front is either the same or better than the Pareto front found in [15],

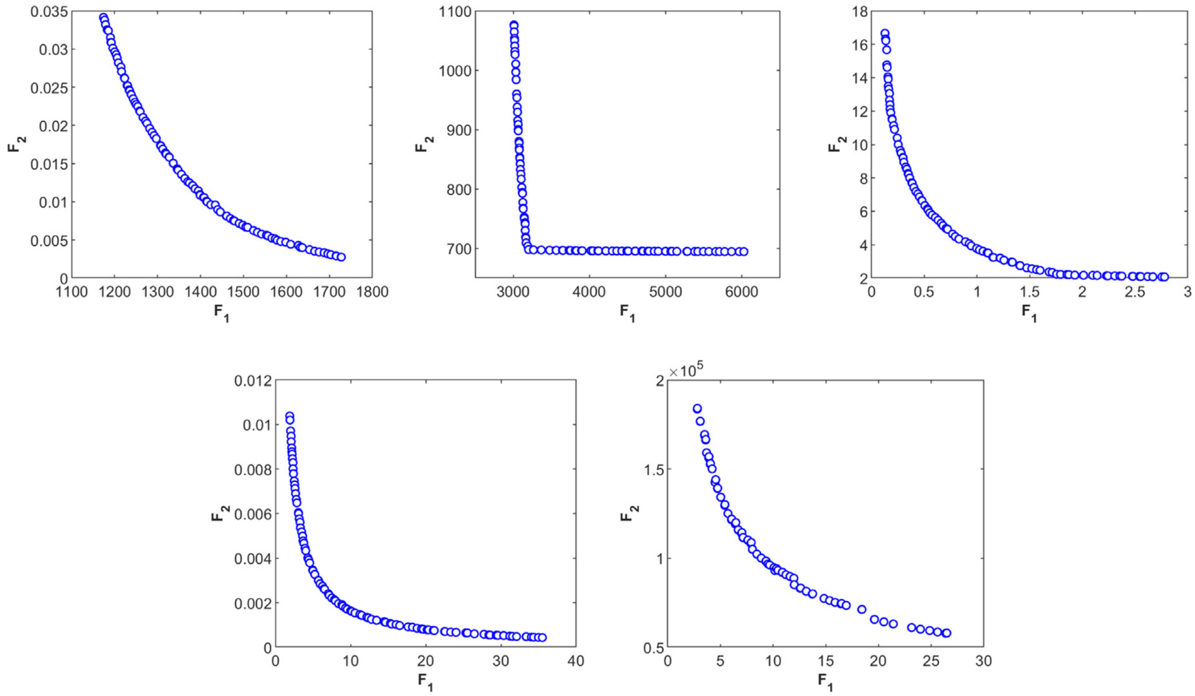



**Fig. 15.** Pareto optimal front generated by MOSGA: four-bar truss design problem, speed reducer design problem, disk brake design problem, welded beam design problem, and spring design problem.

**Table 9**
Results of Wilcoxon rank-sum test of MOSGA versus other algorithms based on HV metric.

| MOSGA versus | NSGA-II | | MOPSO | | MOMVO | | MODA | | MOGOA | |
|---|---|---|---|---|---|---|---|---|---|---|
| | p-value | Signed | p-value | Signed | p-value | Signed | p-value | Signed | p-value | Signed |
| ZDT1 | 1.21E−12 | + | 2.95E−11 | + | 3.02E−11 | + | 1.72E−12 | + | 1.62E−11 | + |
| ZDT2 | 1.21E−12 | + | 6.48E−12 | + | 3.02E−11 | + | 1.21E−12 | + | 2.37E−12 | + |
| ZDT3 | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 1.44E−11 | + | 2.80E−11 | + |
| ZDT6 | 1.21E−12 | + | 1.39E−11 | + | 3.02E−11 | + | 2.37E−12 | + | 1.62E−11 | + |
| BINH1 | 3.02E−11 | + | 6.72E−10 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| DEB1 | 1.38E−02 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| DEB2 | 6.70E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| DEB3 | 5.57E−03 | + | 1.15E−01 | – | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| FON1 | 3.02E−11 | + | 6.70E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| FON2 | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| KUR | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| LAU | 3.02E−11 | + | 5.19E−07 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| MUR | 5.37E−02 | – | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| POL | 6.70E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| SCH1 | 3.02E−11 | + | 3.50E−09 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| SCH2 | 1.78E−10 | + | 1.37E−03 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| VN1 | 3.87E−01 | – | 1.33E−10 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| VN2 | 3.69E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| VN3 | 3.02E−11 | + | 2.90E−01 | – | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| BEL | 2.60E−08 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| BINH2 | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| CONSTR | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| KITA | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| SRN | 5.46E−09 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |
| TNK | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + | 3.02E−11 | + |

**Table 10**
The average ranking of MOO algorithms by Friedman test for all benchmark problems.

| Algorithms | IGD | SP | $\Delta$ | HV |
|---|---|---|---|---|
| MOSGA | **1.2253** | **1.7213** | **1.2733** | **1.1453** |
| NSGA-II | 2.9080 | 3.1667 | 2.9240 | 3.0480 |
| MOPSO | 3.1387 | 3.0140 | 2.3393 | 3.0840 |
| MOMVO | 3.5533 | 4.1007 | 3.9753 | 3.5987 |
| MODA | 5.2693 | 5.2893 | 5.3600 | 5.2000 |
| MOGOA | 4.9053 | 3.7080 | 5.1280 | 4.9240 |

which confirms the exploratory ability of the MOSGA to detect accurate results.

### 4.3.4. Welded beam design problem

Table 18 illustrates results for SP and HV indicators yielded by the MOSGA, NSGA-II, MOPSO, MOMVO, MODA, and MOGOA. The results of the average SP values in Table 18 depict that MOSGA has an SP value of 7.0178E−03 and MOMVO has an SP value of 1.4265E−02, which place them in the first and second ranks, respectively. Likewise, MOSGA has the best performance among methods based on the average HV value. Hence, MOSGA has a

**Table 11**
Comparative results of MOSGA and previous studies in terms of IGD metric for ZDT, WFG, DTLZ, LZ09 test suites, OKA1, OKA2, VN2, VN3, POL, SCH1, FON2, and KUR.

| Test functions | MOSGA | | IBEA [56] | | MOEA/D [56] | | MOVS [57] | | MOAAA [58] | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Average | SD | Average | SD | Average | SD | Average | SD | Average | SD |
| ZDT1 | 1.8461E−04 | 8.5104E−06 | **1.64E−04** | **5.1E−06** | 5.31E−04 | 1.7E−04 | 1.99E−04 | 2.1E−05 | 2.30E−04 | 9.8E−06 |
| ZDT2 | **1.9024E−04** | **6.6743E−06** | 5.42E−04 | 3.7E−05 | 4.40E−04 | 1.6E−04 | 4.74E−04 | 1.0E−04 | 2.38E−04 | 1.4E−05 |
| ZDT3 | 7.6955E−03 | 1.9169E−05 | 1.51E−03 | 7.2E−04 | 1.70E−03 | 7.6E−04 | **1.30E−04** | **4.4E−06** | 1.58E−04 | 9.7E−06 |
| ZDT4 | **5.8685E−05** | **2.8192E−06** | 2.23E−02 | 3.1E−03 | 1.10E−02 | 7.4E−03 | 2.92E−02 | 1.3E−02 | 2.40E−03 | 2.9E−03 |
| ZDT6 | **1.3596E−04** | **8.5432E−06** | 2.55E−04 | 1.0E−05 | 1.42E−04 | 1.1E−05 | 2.19E−04 | 1.5E−05 | 2.51E−04 | 1.3E−05 |
| WFG1 | 6.0877E−04 | **1.5395E−04** | 3.67E−03 | 2.2E−03 | 6.70E−03 | 1.8E−03 | 2.81E−03 | 1.2E−03 | **4.33E−04** | 2.9E−04 |
| WFG2 | **5.5474E−05** | **2.1195E−06** | 3.76E−03 | 9.7E−04 | 4.38E−04 | 1.9E−05 | 1.31E−04 | 4.5E−06 | 1.25E−04 | 4.6E−06 |
| WFG3 | **5.8407E−05** | 2.2498E−06 | 1.30E−04 | 2.9E−06 | 1.36E−04 | **4.8E−07** | 1.44E−04 | 6.2E−06 | 1.47E−04 | 7.4E−06 |
| WFG4 | **6.5071E−05** | **3.6444E−06** | 5.22E−04 | 3.4E−05 | 2.28E−04 | 3.6E−05 | 1.56E−04 | 4.3E−06 | 1.54E−04 | 4.9E−05 |
| WFG5 | 2.5049E−04 | 2.2390E−06 | 3.71E−04 | 3.3E−05 | **1.24E−04** | **9.0E−07** | 1.32E−04 | 1.9E−05 | 1.35E−04 | 1.3E−05 |
| WFG6 | **2.1797E−04** | 1.4976E−04 | 8.54E−04 | 2.0E−04 | 2.28E−04 | **1.8E−06** | 2.38E−04 | 2.7E−05 | 2.30E−04 | 1.5E−05 |
| WFG7 | **6.9975E−05** | 2.8541E−06 | 5.14E−04 | 3.2E−05 | 1.48E−04 | **7.4E−07** | 1.53E−04 | 6.0E−06 | 1.52E−04 | 9.0E−05 |
| WFG9 | **2.1240E−04** | 3.0292E−04 | 4.83E−04 | 4.6E−05 | 2.41E−04 | **5.8E−06** | 2.38E−04 | 1.2E−05 | 2.30E−04 | 1.3E−05 |
| DTLZ1 | **1.2210E−03** | 1.8537E−03 | 4.17E−03 | **3.4E−04** | 7.94E−03 | 6.7E−03 | 4.01E−03 | 4.2E−03 | 4.59E−03 | 5.4E−03 |
| DTLZ2 | **7.4992E−04** | 8.0826E−06 | 1.40E−03 | 2.9E−05 | 5.72E−03 | **1.4E−07** | 7.61E−04 | 3.8E−05 | 7.61E−04 | 3.6E−05 |
| DTLZ4 | **7.7487E−04** | **2.2635E−05** | 5.11E−03 | 2.9E−03 | 1.16E−02 | 1.1E−03 | 1.19E−03 | 1.7E−04 | 1.23E−03 | 1.1E−04 |
| DTLZ5 | 8.4748E−05 | 5.6033E−06 | 1.02E−04 | 5.2E−06 | 1.48E−03 | **2.9E−09** | 2.05E−05 | 1.2E−06 | **1.96E−05** | 9.1E−07 |
| DTLZ6 | 8.4558E−05 | 5.6982E−06 | 4.49E−04 | 1.1E−04 | 3.82E−03 | **3.3E−10** | **5.10E−05** | 3.4E−06 | 5.70E−05 | 3.0E−06 |
| DTLZ7 | **6.1450E−04** | **5.5264E−05** | 1.59E−02 | 8.7E−03 | 2.84E−02 | 1.3E−03 | 2.22E−03 | 1.0E−04 | 2.27E−03 | 1.6E−04 |
| LZ09_F1 | **2.1989E−04** | 3.6404E−05 | 6.84E−04 | 3.3E−04 | 2.36E−04 | **1.5E−05** | 5.09E−04 | 6.0E−05 | 4.38E−04 | 2.3E−05 |
| LZ09_F2 | **1.0524E−03** | **8.6927E−05** | 8.26E−03 | 2.7E−03 | 4.89E−03 | 2.1E−03 | 5.67E−03 | 6.8E−04 | 3.43E−03 | 1.0E−03 |
| LZ09_F3 | **7.3878E−04** | **2.8999E−04** | 6.65E−03 | 3.2E−03 | 5.16E−03 | 3.1E−03 | 3.98E−03 | 9.1E−04 | 2.43E−03 | 2.9E−04 |
| LZ09_F4 | **9.8521E−04** | **3.8067E−04** | 6.10E−03 | 1.1E−03 | 2.98E−03 | 6.4E−04 | 3.84E−03 | 6.6E−04 | 2.53E−03 | 6.9E−04 |
| LZ09_F5 | **5.1803E−04** | **1.2566E−04** | 4.58E−03 | 2.2E−03 | 4.03E−03 | 2.5E−03 | 3.05E−03 | 3.4E−04 | 1.89E−03 | 3.0E−04 |
| LZ09_F6 | **1.9029E−03** | **3.4620E−04** | 1.62E−02 | 2.4E−03 | 1.92E−02 | 9.8E−04 | 5.78E−03 | 1.3E−03 | 6.19E−03 | 1.2E−03 |
| LZ09_F7 | **2.4141E−03** | **4.1362E−04** | 1.92E−02 | 5.5E−03 | 9.18E−03 | 6.3E−03 | 1.23E−02 | 4.0E−03 | 1.15E−02 | 3.9E−03 |
| LZ09_F8 | **2.4426E−03** | **4.2491E−04** | 1.69E−02 | 5.7E−03 | 1.06E−02 | 3.1E−03 | 1.20E−02 | 2.3E−03 | 1.23E−02 | 2.8E−03 |
| LZ09_F9 | **1.2852E−03** | **9.5022E−05** | 9.02E−03 | 2.6E−03 | 4.80E−03 | 1.8E−03 | 5.75E−03 | 5.7E−04 | 3.89E−03 | 1.2E−03 |
| OKA1 | **2.6251E−03** | 1.3872E−03 | 4.94E−03 | 2.4E−03 | 2.73E−03 | 9.2E−04 | 2.65E−03 | 9.6E−04 | 3.54E−03 | **4.4E−04** |
| OKA2 | **8.7693E−03** | 1.2262E−03 | 1.95E−02 | 9.1E−03 | 1.71E−02 | 9.1E−03 | 9.57E−03 | 1.2E−03 | 1.17E−02 | **5.6E−04** |
| VN2 | **3.1659E−04** | 2.7703E−05 | 1.47E−03 | 3.6E−04 | 2.91E−03 | **1.3E−06** | 3.52E−04 | 4.8E−05 | 3.24E−04 | 3.5E−05 |
| VN3 | **1.5954E−04** | 7.5581E−06 | 4.37E−03 | 2.6E−04 | 7.50E−03 | **1.5E−06** | 1.83E−04 | 2.0E−05 | 1.76E−04 | 2.6E−05 |
| POL | 1.2707E−04 | 5.6662E−06 | 1.07E−03 | 6.3E−04 | 7.87E−04 | 1.1E−03 | 1.05E−04 | 6.7E−06 | **9.42E−05** | **5.0E−06** |
| SCH1 | **1.9950E−04** | **1.1256E−05** | 2.61E−02 | 2.5E−02 | 2.65E−02 | 1.5E−02 | 5.84E−04 | 2.2E−05 | 3.13E−02 | 2.3E−02 |
| FON2 | **1.9220E−04** | 6.2726E−06 | 2.59E−04 | 7.5E−06 | 2.04E−04 | **4.8E−07** | 3.19E−04 | 1.4E−05 | 2.82E−04 | 1.0E−05 |
| KUR | 1.9671E−04 | 1.1600E−05 | 1.28E−03 | 1.7E−04 | 1.74E−04 | **1.5E−06** | 1.76E−04 | 9.0E−06 | **1.59E−04** | 8.1E−06 |

better possibility to detect a broader range of Pareto optimal solutions for both objective functions and offer more choices for decision-makers to choose their optimal design from a Pareto optimal set.

Fig. 15 clarifies the discussion in Table 18 by depicting the Pareto optimal front generated by MOSGA. From Fig. 15, the Pareto optimal solutions yielded by MOSGA are between the range of (35.4929, 0.000439) and (1.8991, 0.0103).

### *4.3.5. Spring design problem*

Table 19 shows a comparison of SP and HV indicators obtained by the MOSGA, NSGA-II, MOMVO, MOMVO, MODA, and MOGOA. According to Table 19, MOSGA outperforms other optimizers in providing a wider variety of Pareto optimal solutions with better distribution. The best average values and minimum deviation of SP and HV metrics attained by the MOSGA support this claim. It points out that MOSGA has superior convergence and diversity in comparison to other techniques. To clarify further, Fig. 15 portrays the Pareto optimal front obtained by MOSGA. The extreme Pareto optimal solutions yielded by MOSGA are (26.4782, 58018.6041) and (2.8019, 183667.1169).

Table 20 lists the average rankings of MOO algorithms based on the Friedman rank test for five multi-objective engineering problems. Results of the Friedman test show that MOSGA is the most efficient method. The MOSGA yields the best average ranking for the SP and HV indicators.

## *4.4. Sensitivity analysis for initial parameters of MOSGA*

Selecting the most efficient initial parameters is one of the most critical concerns for optimization algorithms. To assess the effect of initial parameters on the performance of MOSGA, we perform a sensitivity analysis of the number of search group members ($n_g$), perturbation constant ($\alpha^k$), and global search ratio (*GIR*).

Table 21 depicts the sensitivity of the perturbation coefficient ($\alpha^k$) in terms of values for different performance metrics, including the IGD, SP, $\Delta$, and HV metrics for test functions BINH1 and BINH2. Without loss of generality, the values of $\alpha^k$ are set 1, 2, 3, 4, and 5, respectively. Value for $\alpha^k$ has an impact on the exploration and exploitation phases in MOSGA. Generally, a high value of $\alpha^k$ reduces an appropriate search around the best solutions. Therefore, MOSGA may not be able to obtain the global optimal solution. Meanwhile, a low value for $\alpha^k$ reduces the search intensity. Hence, the MOSGA focuses on the vicinity of the best solutions without paying more attention to other domains. Consequently, MOSGA can be stuck in the local Pareto optimum for complex problems. As can be seen in Table 21, a value $\alpha^k$ of 3 obtains the best solutions with the best values for IGD, SP, $\Delta$, and HV indicators. Therefore, it can be deduced that a suitable value for $\alpha^k$ is 3.

The number of search group members ($n_g$) is the second parameter, which can affect the performance of MOSGA. The sensitivity of $n_g$ for IGD, SP, $\Delta$, and HV metrics for test functions BINH1 and BINH2 are presented in Table 22. Without loss of generality, the value of $n_g$ is tuned to change from 10 to 30 with a step size of 5. From these results, the value for $n_g$ of 20 can be considered a proper value, which works well in terms of IGD, SP, $\Delta$, and HV indicators.

An optimization engine with a proper balance between global and local search is vital. Thus, a sensitivity analysis of both phases on the performance of MOSGA is provided in Table 23. Without

**Table 12**
Comparative results of MOSGA and previous studies in terms of $\Delta$ metric for ZDT, WFG, DTLZ, LZ09 test suites, OKA1, OKA2, VN2, VN3, POL, SCH1, FON2, and KUR.

| Test functions | MOSGA | | IBEA [56] | | MOEA/D [56] | | MOVS [57] | | MOAAA [58] | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Average | SD | Average | SD | Average | SD | Average | SD | Average | SD |
| ZDT1 | 3.5363E−01 | 2.3411E−02 | **2.97E−01** | **1.8E−02** | 3.66E−01 | 4.8E−02 | 4.64E−01 | 3.1E−02 | 6.40E−01 | 5.1E−02 |
| ZDT2 | **3.2472E−01** | **1.3231E−02** | 3.37E+01 | 2.3E−02 | 3.25E−01 | 9.4E−02 | 5.19E−01 | 3.9E−02 | 6.79E−01 | 6.5E−02 |
| ZDT3 | 7.8518E−01 | 1.2423E−02 | 1.19E+00 | 6.9E−02 | 9.93E−01 | 3.2E−02 | **7.55E−01** | **9.0E−03** | 8.08E−01 | 2.4E−02 |
| ZDT4 | **3.8224E−01** | **3.5226E−02** | 1.11E+00 | 5.6E−02 | 9.69E−01 | 1.7E−01 | 1.23E+00 | 1.1E−01 | 5.85E−01 | 1.7E−01 |
| ZDT6 | **3.1877E−01** | 2.8611E−02 | 4.15E−01 | 4.8E−02 | 1.54E−01 | **8.3E−03** | 5.70E−01 | 4.5E−02 | 7.86E−01 | 3.5E−02 |
| WFG1 | 7.6689E−01 | **3.8033E−02** | 8.74E−01 | 6.9E−02 | 1.07E+00 | 1.5E−01 | 6.67E−01 | 1.9E−01 | **6.46E−01** | 9.6E−02 |
| WFG2 | 7.8831E−01 | 8.9313E−03 | 1.25E+00 | 7.9E−02 | 1.11E+00 | **5.6E−03** | 7.88E−01 | 8.7E−03 | **7.82E−01** | 1.02-02 |
| WFG3 | 3.4651E−01 | 3.1399E−02 | **2.55E−01** | 2.8E−02 | 3.45E−01 | **8.6E−04** | 3.24E−01 | 3.1E−02 | 3.16E−01 | 2.4E−02 |
| WFG4 | 3.3862E−01 | 2.4868E−02 | 5.13E−01 | 3.4E−02 | 5.10E−01 | 5.2E−02 | 3.55E−01 | **2.3E−02** | **3.27E−01** | 2.5E−02 |
| WFG5 | **3.9142E−01** | 2.7003E−02 | 5.85E−01 | 7.3E−02 | 4.53E−01 | **5.0E−03** | 3.98E−01 | 2.8E−02 | 4.29E−01 | 3.3E−02 |
| WFG6 | 3.8648E−01 | 2.9657E−02 | 5.26E−01 | 2.8E−02 | 4.13E−01 | **6.4E−03** | 3.49E−01 | 2.7E−02 | **3.30E−01** | 2.9E−02 |
| WFG7 | 3.9659E−01 | 2.9020E−02 | 5.15E−01 | 2.8E−02 | 4.14E−01 | **6.6E−03** | 3.75E−01 | 2.7E−02 | **3.44E−01** | 2.9E−02 |
| WFG9 | 3.7591E−01 | 4.0468E−02 | 5.00E−01 | 4.3E−02 | 4.48E−01 | **1.4E−02** | 3.31E−01 | 2.9E−02 | **3.29E−01** | 2.6E−02 |
| DTLZ1 | **4.9744E−01** | 1.3710E−01 | 1.64E+00 | 1.1E−01 | 1.09E+00 | 8.9E−02 | 7.58E−01 | **5.9E−02** | 7.38E−01 | 6.7E−02 |
| DTLZ2 | **4.7569E−01** | 4.1797E−02 | 5.83E−01 | 5.0E−02 | 1.00E+00 | **2.1E−05** | 6.58E−01 | 4.1E−02 | 6.70E−01 | 3.9E−02 |
| DTLZ4 | **4.9120E−01** | 5.1293E−02 | 7.04E−01 | 1.4E−01 | 1.02E+00 | 8.4E−02 | 6.54E−01 | 4.02-02 | 6.40E−01 | **3.6E−02** |
| DTLZ5 | 5.3191E−01 | 5.9106E−02 | 6.74E−01 | 3.9E−02 | 1.00E+00 | **2.9E−06** | 4.98E−01 | 5.8E−02 | **4.45E−01** | 4.7E−02 |
| DTLZ6 | 6.2543E−01 | 7.3467E−02 | 9.83E−01 | 1.7E−01 | 1.00E+00 | **2.6E−08** | **5.35E−01** | 3.9E−02 | 6.77E−01 | 4.4E−02 |
| DTLZ7 | **5.7549E−01** | 6.3992E−02 | 8.26E−01 | 1.1E−01 | 9.90E−01 | 2.4E−02 | 7.19E−01 | 3.8E−02 | 7.86E−01 | 4.7E−02 |
| LZ09_F1 | 5.4704E−01 | 4.7112E−02 | 7.66E−01 | 5.1E−02 | **3.13E−01** | **4.2E−02** | 4.13E−01 | 7.5E−02 | 3.90E−01 | 1.2E−01 |
| LZ09_F2 | 1.4560E+00 | **7.7391E−02** | 1.47E+00 | 1.3E−01 | **9.92E−01** | 1.4E−01 | 1.51E+00 | 1.2E−01 | 1.15E+00 | 1.2E−01 |
| LZ09_F3 | 5.9809E−01 | **2.5325E−02** | 1.12E+00 | 1.0E−01 | 7.01E−01 | 8.8E−02 | 8.11E−01 | 1.0E−01 | **5.95E−01** | 7.0E−02 |
| LZ09_F4 | 6.6043E−01 | 1.1133E−01 | 1.03E+00 | **4.8E−02** | 9.71E−01 | 1.7E−01 | 6.18E−01 | 7.9E−02 | **5.57E−01** | 7.1E−02 |
| LZ09_F5 | 5.5921E−01 | **1.8684E−02** | 1.09E+00 | 7.7E−02 | 6.64E−01 | 8.7E−02 | 6.89E−01 | 7.4E−02 | **5.53E−01** | 5.5E−02 |
| LZ09_F6 | **8.5872E−01** | 4.2198E−02 | 1.66E+00 | 4.1E−01 | 9.87E−01 | **2.8E−02** | 8.70E−01 | 7.8E−02 | 9.26E−01 | 1.0E−01 |
| LZ09_F7 | 1.4757E+00 | **1.0291E−01** | **1.12E+00** | 1.5E−01 | 1.18E+00 | 1.8E−01 | 1.29E+00 | 1.7E−01 | 1.29E+00 | 2.2E−01 |
| LZ09_F8 | 1.5498E+00 | 1.4076E−01 | **1.18E+00** | 1.6E−01 | 1.27E+00 | **7.4E−02** | 1.19E+00 | 1.6E−01 | 1.25E+00 | 2.9E−01 |
| LZ09_F9 | 1.7528E+00 | **9.9970E−02** | 1.67E+00 | 1.2E−01 | **9.77E−01** | 1.3E−01 | 1.66E+00 | 1.9E−01 | 1.31E+00 | 2.0E−01 |
| OKA1 | **9.1705E−01** | **5.1704E−02** | 1.63E+00 | 8.8E−02 | 1.17E+00 | 8.1E−02 | 1.03E+00 | 5.7E−02 | 1.33E+00 | 9.8E−02 |
| OKA2 | **1.2893E+00** | 1.5059E−01 | 1.45E+00 | 3.3E−01 | 1.52E+00 | **3.1E−01** | 1.44E+00 | 8.7E−02 | 1.55E+00 | 1.2E−01 |
| VN2 | **5.5021E−01** | **5.3611E−02** | 9.80E−01 | 5.6E−02 | 1.00E+00 | 7.3E−04 | 8.55E−01 | 9.8E−02 | 8.71E−01 | 8.2E−02 |
| VN3 | **4.8448E−01** | 6.3981E−02 | 8.09E−01 | 7.4E−02 | 1.00E+00 | **3.0E−05** | 7.33E−01 | 4.5E−02 | 7.00E−01 | 6.0E−02 |
| POL | **7.8359E−01** | **8.1745E−03** | 1.17E+00 | 4.3E−02 | 1.26E+00 | 4.7E−02 | 8.12E−01 | 2.1E−02 | 7.93E−01 | 1.8E−02 |
| SCH1 | **3.5224E−01** | **2.1944E−02** | 6.53E−01 | 2.2E−01 | 1.15E+00 | 3.4E−02 | 7.39E−01 | 3.5E−02 | 6.43E−01 | 2.8E−01 |
| FON2 | 3.8055E−01 | 2.7448E−02 | 4.06E−01 | 2.5E−02 | 1.46E−01 | **5.7E−04** | 4.10E−01 | 3.6E−02 | **2.89E−01** | 2.2E−02 |
| KUR | 5.7412E−01 | 2.2797E−02 | 8.66E−01 | 2.9E−02 | 7.31E−01 | 4.2E−03 | 5.65E−01 | 2.5E−02 | **4.98E−01** | **1.9E−02** |

loss of generality, the global iteration ratio (*GIR*) is set to 1, 0.3, and 0, respectively. When the global iteration ratio equals 1, MOSGA only executes the global phase without considering the local phase. Optimization is dedicated to the exploration of the search space to find promising regions. This leads to the diversification of non-dominated solutions, as demonstrated by the low value of the SP and $\Delta$ metrics. When the global iteration ratio equals 0, MOSGA ignores the global phase and executes only the local phase. Optimization is dedicated to the exploitation of the best solutions found. This allows for improved convergence of solutions; however, MOSGA is easily trapped in the local Pareto optimum for complex problems. From Table 23, a global iteration ratio of 0.3 could be defined as a proper value, which works well for IGD, SP, $\Delta$, and HV metrics. Therefore, the implementation of both phases at a reasonable ratio in MOSGA is essential to balance between exploitation and exploration.

### 4.5. Discussion

The MOSGA performance is studied on 25 different benchmark functions and five multi-objective engineering problems. MOSGA is compared with NSGA-II, MOMVO, MOMVO, MODA, and MOGOA for GD, IGD, SP, $\Delta$, and HV indicators. Moreover, MOSGA is also compared with other recent methods in previous studies on a different set of 36 benchmark functions and some constrained test functions using IGD, $\Delta$, and HV metrics. In this study, GD and IGD are used to assess the convergence and accuracy of the algorithm, while SP and $\Delta$ metrics are used to measure the distribution and spread of obtained solutions. Of all five performance measures, HV is a robust metric for assessing both the convergence and diversity ability of a MOO method. The obtained results of MOSGA are presented based on different statistical analyses, non-parametric statistical tests, robustness analysis, and graphical representations of Pareto optimal fronts. The statistical results of performance criteria reveal that the MOSGA is able to offer a superior quality of solutions in comparison to other methods. For all test functions, Pareto fronts yielded by MOSGA magnificently converge on true Pareto fronts with high diversity. Based on results of non-parametric statistical tests using Friedman rank test and Wilcoxon rank-sum test, MOSGA outperforms NSGA-II, MOMVO, MOMVO, MODA, and MOGOA for majority of performance metrics. In summary, the comparative results show that the MOSGA provides high convergence and diversity in comparison to other MOO algorithms.

Superior convergence is originated from MOSGA search mechanisms. Firstly, MOSGA benefits from high exploration. MOSGA search engine is based on a search group, allowing MOSGA to consider a set of best solutions obtained so far (i.e., family leaders) and oblige other solutions to update their positions based on these family leaders. This mechanism helps MOSGA to explore the search space more extensively and find more promising regions. Moreover, MOSGA performs mutation process to explore newer areas of the search domain in each iteration. MOSGA improves its exploration ability and local minima avoidance. The high exploitation capability of MOSGA is another reason for high convergence. Convergence is improved in MOSGA over the iterations since the perturbation coefficient ($\alpha^k$) is adaptively decreased to turn the optimization process from exploration to exploitation. This allows MOSGA to search locally and exploit promising areas

**Table 13**
Comparative results of MOSGA and previous studies in terms of HV metric for ZDT, WFG, DTLZ, LZ09 test suites, OKA1, OKA2, VN2, VN3, POL, SCH1, FON2, and KUR.

| Test functions | MOSGA | | IBEA [56] | | MOEA/D [56] | | MOVS [57] | | MOAAA [58] | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Average | SD | Average | SD | Average | SD | Average | SD | Average | SD |
| ZDT1 | **7.1811E−01** | 1.8710E−04 | 6.62E−01 | **9.0E−05** | 6.40E−01 | 7.6E−03 | 6.60E−01 | 2.2E−04 | 6.58E−01 | 3.4E−04 |
| ZDT2 | **4.4229E−01** | 2.7843E−04 | 3.27E−01 | **1.4E−04** | 3.10E−01 | 6.8E−03 | 3.26E−01 | 3.32-04 | 3.25E−01 | 4.6E−04 |
| ZDT3 | **6.5822E−01** | **1.3506E−04** | 5.09E−01 | 6.2E−03 | 4.43E−01 | 2.5E−02 | 5.15E−01 | 2.6E−04 | 5.15E−01 | 1.5E−04 |
| ZDT4 | **7.1880E−01** | **3.4918E−04** | 2.36E−01 | 9.3E−02 | 3.01E−01 | 1.8E−01 | 1.10E−01 | 1.5E−01 | 5.57E−01 | 1.2E−01 |
| ZDT6 | **5.0470E−01** | **1.1665E−04** | 3.96E−01 | 5.5E−04 | 4.01E−01 | 9.4E−04 | 3.99E−01 | 4.2E−04 | 3.98E−01 | 3.1E−04 |
| WFG1 | **6.6106E−01** | 1.2900E−02 | 4.73E−01 | 1.1E−01 | 3.22E−01 | 8.7E−02 | 4.90E−01 | 5.7E−02 | 6.20E−01 | **1.6E−02** |
| WFG2 | **6.3357E−01** | **8.4746E−05** | 5.50E−01 | 8.3E−04 | 5.55E−01 | 3.4E−04 | 5.57E−01 | 1.3E−04 | 5.57E−01 | 1.1E−04 |
| WFG3 | **5.7948E−01** | 2.8025E−04 | 4.94E−01 | 3.7E−04 | 4.93E−01 | **1.3E−04** | 4.93E−01 | 3.5E−04 | 4.92E−01 | 3.5E−04 |
| WFG4 | **3.4502E−01** | 3.9828E−04 | 2.09E−01 | **1.8E−04** | 2.04E−01 | 1.8E−03 | 2.10E−01 | 2.1E−04 | 2.09E−01 | 2.5E−04 |
| WFG5 | **3.1268E−01** | 1.9629E−04 | 2.17E−01 | 8.0E−04 | 2.18E−01 | **4.7E−05** | 2.20E−01 | 4.0E−03 | 2.18E−01 | 2.5E−03 |
| WFG6 | **3.1817E−01** | 2.1664E−02 | 1.97E−01 | 1.1E−02 | 2.09E−01 | **1.5E−04** | 2.07E−01 | 2.1E−03 | 2.08E−01 | 1.2E−03 |
| WFG7 | **3.4431E−01** | 4.1369E−04 | 2.08E−01 | 1.8E−04 | 2.09E−01 | **8.4E−05** | 2.09E−01 | 2.1E−04 | 2.09E−01 | 3.4E−04 |
| WFG9 | **3.2135E−01** | 4.3452E−02 | 2.24E−01 | 1.3E−03 | 2.22E−01 | **6.9E−04** | 2.23E−01 | 1.0E−03 | 2.24E−01 | 1.0E−03 |
| DTLZ1 | **7.5276E−01** | 2.1590E−01 | 1.74E−01 | **8.1E−02** | 2.01E−01 | 1.4E−01 | 6.37E−01 | 1.6E−01 | 4.22E−01 | 3.4E−01 |
| DTLZ2 | **5.3435E−01** | 3.3522E−03 | 4.12E−01 | 6.0E−04 | 7.55E−02 | **3.1E−06** | 3.75E−01 | 3.5E−03 | 3.71E−01 | 5.4E−03 |
| DTLZ4 | **5.3527E−01** | **3.7028E−03** | 2.42E−01 | 1.3E−01 | 8.15E−02 | 2.7E−02 | 3.82E−01 | 4.3E−03 | 3.71E−01 | 5.3E−03 |
| DTLZ5 | **1.9900E−01** | **1.1852E−04** | 9.17E−02 | 1.8E−04 | 1.43E−02 | 1.6E−07 | 9.28E−02 | 2.1E−04 | 9.28E−02 | 2.1E−04 |
| DTLZ6 | **1.9950E−01** | 1.4479E−04 | 7.22E−02 | 1.4E−02 | 1.45E−02 | **3.8E−09** | 9.408-02 | 1.6E−04 | 9.36E−02 | 1.8E−04 |
| DTLZ7 | **3.9943E−01** | 3.6629E−03 | 2.34E−01 | 4.6E−02 | 6.38E−02 | 4.9E−02 | 2.70E−01 | 5.1E−03 | 2.86E−01 | **3.5E−03** |
| LZ09_F1 | **6.9678E−01** | 2.6201E−03 | 6.55E−01 | 1.4E−03 | 6.61E−01 | **1.1E−04** | 6.50E−01 | 1.1E−03 | 6.51E−01 | 7.1E−04 |
| LZ09_F2 | **6.1353E−01** | **8.4377E−03** | 4.93E−01 | 4.4E−02 | 5.27E−01 | 3.8E−02 | 5.27E−01 | 1.5E−02 | 5.59E−01 | 2.3E−02 |
| LZ09_F3 | **6.6161E−01** | 8.1761E−03 | 5.85E−01 | 1.7E−02 | 5.96E−01 | 2.7E−02 | 5.78E−01 | **6.4E−03** | 5.97E−01 | 7.6E−03 |
| LZ09_F4 | **6.5713E−01** | 9.3259E−03 | 6.02E−01 | 5.9E−03 | 6.20E−01 | 5.2E−03 | 5.93E−01 | **4.5E−03** | 6.05E−01 | 6.1E−03 |
| LZ09_F5 | **6.7478E−01** | 4.1113E−03 | 6.03E−01 | 1.2E−02 | 6.09E−01 | 1.7E−02 | 5.97E−01 | **3.8E−03** | 6.12E−01 | 5.2E−03 |
| LZ09_F6 | **3.6755E−01** | 4.8070E−02 | 6.43E−02 | 7.5E−02 | 7.26E−02 | **1.2E−02** | 2.27E−01 | 3.0E−02 | 1.90E−01 | 3.4E−02 |
| LZ09_F7 | **4.4411E−01** | **3.7451E−02** | 3.93E−01 | 6.1E−02 | 4.43E−01 | 1.5E−01 | 3.78E−01 | 6.0E−02 | 3.19E−01 | 1.4E−01 |
| LZ09_F8 | **4.0389E−01** | 5.7868E−02 | 3.85E−01 | 5.5E−02 | 3.50E−01 | 9.6E−02 | 3.32E−01 | **4.5E−02** | 2.72E−01 | 1.1E−01 |
| LZ09_F9 | **3.3760E−01** | **5.1140E−03** | 1.63E−01 | 4.9E−02 | 1.67E−01 | 6.4E−02 | 2.05E−01 | 1.4E−02 | 2.26E−01 | 2.4E−02 |
| OKA1 | **6.7246E−01** | 9.6989E−03 | 5.67E−01 | 1.5E−02 | 6.00E−01 | 7.6E−03 | 6.04E−01 | **6.5E−03** | 5.56E−01 | 9.5E−03 |
| OKA2 | **2.8171E−01** | 2.2090E−02 | 6.29E−02 | 5.0E−02 | 5.25E−02 | 3.5E−02 | 1.23E−01 | 2.9E−02 | 5.77E−02 | **1.2E−02** |
| VN2 | **9.3304E−01** | 1.1161E−03 | 9.07E−01 | 7.4E−03 | 7.25E−01 | **4.2E−04** | 9.20E−01 | 1.1E−03 | 9.21E−01 | 1.1E−03 |
| VN3 | **8.6117E−01** | 5.7938E−04 | 8.29E−01 | 4.4E−03 | 5.66E−01 | **5.7E−05** | 8.32E−01 | 6.2E−04 | 8.33E−01 | 5.9E−04 |
| POL | **9.2797E−01** | **5.6186E−05** | 9.11E−01 | 1.3E−03 | 9.11E−01 | 1.3E−03 | 9.13E−01 | 1.1E−04 | 9.13E−01 | 6.9E−05 |
| SCH1 | **8.5878E−01** | **1.4293E−04** | 5.30E−01 | 2.1E−01 | 5.03E−01 | 1.9E−01 | 8.27E−01 | 3.4E−04 | 4.70E−01 | 2.2E−01 |
| FON2 | **4.2956E−01** | 2.3297E−04 | 3.11E−01 | 1.1E−04 | 3.12E−01 | **1.2E−05** | 3.08E−01 | 4.1E−04 | 3.09E−01 | 3.6E−04 |
| KUR | **5.0044E−01** | 3.9042E−04 | 3.94E−01 | 9.2E−04 | 4.00E−01 | **1.1E−04** | 4.00E−01 | 2.4E−04 | 4.00E−01 | 2.0E−04 |

in the search space. Since best obtained non-dominated solutions are stored in the Pareto archive, choosing a new search group from the Pareto archive emphasizes the exploitation of the best regions in the local phase. High exploitation also leads to high convergence. Thanks to the perturbation coefficient and proposed schemes in global and local phases, MOSGA has the advantage of striking a satisfactory balance between exploitation and exploration capabilities.

Another advantage is the high diversity (distribution and spread) of the MOSGA due to new search group selection and Pareto archive update mechanisms. MOSGA uses tournament selection to provide a high probability of choosing new family leaders from less crowded regions of the obtained non-dominated front. This promotes the search group to exploit and explore the less crowded regions of the search space and front. Moreover, a selection mechanism is used to discard non-dominated solutions from the most crowded regions when the Pareto archive is full. Hence, MOSGA boosts the diversity of solutions during the optimization process.

Although the proposed MOSGA obtains remarkable results for all benchmark test functions and real engineering problems, it also has a certain limitation. Since MOSGA is developed as a Pareto dominance-based algorithm, it can effectively solve MOPs with two and three conflict objective functions and obtain satisfactory results. However, when dealing with MOPs with more than three objective functions, a large number of non-dominated solutions are obtained at each iteration and the archive becomes full quickly. This results in the performance of the MOSGA possibly being less efficient when solving such problems. Hence, MOSGA is suitable for MOPs with two and three objective functions.

## 5. Conclusion

This paper has proposed the first multi-objective version of the SGA called MOSGA. SGA mechanisms are modified by integrating two new modules to develop the MOSGA. Initially, the elitist non-dominated sorting approach is applied to help determine non-dominated solutions via three significant stages: mutation, offspring generation, and selection. The Pareto archive selection mechanism is the second module to maintain and enhance the convergence and diversity of non-dominated solutions during optimization. The effectiveness of the MOSGA is demonstrated by solving twenty-five unconstrained and constrained benchmark problems. The performance metrics are used for comparison included IGD, SP, $\Delta$, and HV metrics. The statistical results prove that MOSGA provides a superior quality of solutions compared to five well-regarded algorithms considered in this study. MOSGA outperforms the other algorithms for all the test problems based on the convergence (IGD metric). Besides, MOSGA also surpasses the other algorithms for most problems based on diversity (SP and $\Delta$ metrics). For the HV metric, MOSGA also has the best performance. All Pareto optimal fronts yielded by MOSGA are well-converged on true Pareto optimal fronts with high diversity. Moreover, the study further tests MOSGA with five real engineering problems to verify its applicability. For all applications, MOSGA obtains superior solution quality compared to other algorithms. High convergence and diversity of obtained Pareto optimal fronts are due to the high exploitation and exploration

**Table 14**
Comparative results of MOSGA and previous studies in terms of GD, SP, and $\Delta$ metrics for test functions BINH2, CONSTR, KITA, SRN, and TNK.

| Algorithms | GD | | SP | | $\Delta$ | | Times |
|---|---|---|---|---|---|---|---|
| | Average | SD | Average | SD | Average | SD | |
| **BINH2** | | | | | | | |
| MOSGA | **1.1339E−04** | **1.5103E−05** | **7.9900E−03** | **4.3079E−04** | **3.9372E−01** | **2.5165E−02** | **2.217** |
| MOCBO [21] | 1.498E−01 | 7.60E−03 | – | – | 4.798E−01 | 7.210E−02 | 9.1544 |
| MOSOS [21] | 1.436E−01 | 6.24E−03 | – | – | 4.288E−01 | 6.250E−02 | 16.2664 |
| NS-GWO [63] | 1.6689E−01 | 5.97E−03 | – | – | 4.879E−01 | 8.965E−02 | 8.5685 |
| **CONSTR** | | | | | | | |
| MOSGA | **2.1479E−04** | **2.2059E−05** | **7.2431E−03** | **6.0451E−04** | 3.9081E−01 | 2.2611E−02 | **2.335** |
| MOWCA [15] | 8.7102E−04 | 5.7486E−05 | 3.5E−02 | 1.0E−03 | **1.745E−01** | 2.0441E−02 | – |
| CMOPSO [15] | 2.9894E−03 | 8.3111E−03 | – | – | 5.7586E−01 | 2.2894E−02 | – |
| MOCBO [21] | 5.202E−01 | 3.42E−03 | – | – | 7.235E−01 | 8.32E−03 | 5.2252 |
| MOSOS [21] | 5.162E−01 | 2.14E−03 | – | – | 7.122E−01 | 7.21E−03 | 10.011 |
| SPEA2 [55] | 4.8247E−03 | 3.7650E−04 | – | – | 4.280E−01 | 1.2258E−02 | – |
| MOSADE [55] | 4.8101E−03 | 3.7427E−04 | – | – | 3.546E−01 | 5.8029E−02 | – |
| NS-GWO [63] | 4.8955E−01 | 1.802E−02 | – | – | 6.598E−01 | **5.69E−04** | 12.895 |
| **KITA** | | | | | | | |
| MOSGA | **5.1505E−04** | **1.8588E−04** | **6.1059E−03** | **9.3840E−04** | **3.1062E−01** | **2.1173E−02** | **2.152** |
| Micro-GA [15] | 1.507E−01 | 8.97E−02 | 3.150E−01 | 4.217E−01 | – | – | – |
| PAES [15] | 1.931E−01 | 3.32E−02 | 1.101E−01 | 9.95E−02 | – | – | – |
| MOCS [15] | 2.74E−02 | 3.24E−02 | 1.592E−01 | 2.338E−01 | 1.0169E+00 | 1.147E−01 | – |
| MOWCA [15] | 4.9E−03 | 4.5E−03 | 4.85E−02 | 4.78E−02 | 3.764E−01 | 7.44E−02 | – |
| MOALO [18] | 4.20E−02 | 4.9E−02 | 2.9E−01 | 4.2E−01 | 6.0E−01 | 1.9E−01 | – |
| MOCBO [21] | 3.84E−02 | 9.20E−03 | – | – | 7.734E−01 | 8.324E−02 | 10.324 |
| MOSOS [21] | 3.68E−02 | 8.30E−03 | – | – | 6.832E−01 | 7.242E−02 | 14.382 |
| **SRN** | | | | | | | |
| MOSGA | **1.3779E−04** | **1.5733E−05** | **6.3641E−03** | **6.2374E−04** | 3.8862E−01 | 3.4467E−02 | **2.253** |
| CMOPSO [15] | 2.5331E−02 | 5.2561E−03 | – | – | 1.965E−01 | 2.4527E−02 | – |
| MOWCA [15] | 2.5836E−02 | 5.0102E−03 | 4.164E−01 | 7.79E−02 | 1.477E−01 | 1.3432E−02 | – |
| MOCBO [21] | 1.018E−01 | 1.56E−03 | – | – | 2.352E−01 | 1.93E−03 | 7.3251 |
| MOSOS [21] | 9.88E−02 | 1.47E−03 | – | – | 2.295E−01 | 1.76E−03 | 12.325 |
| SPEA2 [55] | 2.1059E−03 | 4.7502E−04 | – | – | **1.0134E−01** | 1.9082E−02 | – |
| MOSADE [55] | 2.0028E−03 | 2.0227E−04 | – | – | 1.0525E−01 | 9.3834E−03 | – |
| NS-GWO [63] | 6.987E−02 | 1.785E−02 | – | – | 2.001E−01 | **6.5E−04** | 7.2440 |
| **TNK** | | | | | | | |
| MOSGA | **4.2795E−04** | 6.3456E−05 | 4.7533E−03 | 5.8748E−04 | 7.1382E−01 | 3.2403E−02 | **2.498** |
| MOWCA [15] | 1.3067E−03 | 6.1979E−05 | – | – | 6.0211E−01 | 2.8148E−02 | – |
| CMOPSO [15] | 5.4811E−04 | 7.9634E−05 | – | – | 2.5871E−01 | 2.7272E−02 | – |
| MOALO [18] | 7.97E−04 | **5.4E−05** | **2.0E−03** | **1.0E−04** | 6.4E−01 | **1.2E−02** | – |
| MOCBO [21] | 1.528E−01 | 5.12E−03 | – | – | 1.242E−01 | 5.124E−02 | 11.010 |
| MOSOS [21] | 1.508E−01 | 4.04E−03 | – | – | 1.206E−01 | 4.236E−02 | 15.128 |
| SPEA2 [55] | 3.8175E−03 | 4.9142E−04 | – | – | 7.8373E−01 | 2.9969E−02 | – |
| MOSADE [55] | 3.7393E−03 | 3.7907E−04 | – | – | 7.5265E−01 | 2.9918E−02 | – |
| NS-GWO [63] | 1.4785E−01 | 3.35E−03 | – | – | **9.955E−02** | 2.568E−02 | 9.1156 |

**Table 15**
Results of MOO algorithms in terms of SP and HV metrics for four-bar truss design problem.

| Algorithms | SP | | HV | | Times |
|---|---|---|---|---|---|
| | Average | SD | Average | SD | |
| MOSGA | **8.6497E−03** | **4.2535E−04** | **7.6022E−01** | **2.9020E−04** | **0.9979** |
| NSGA-II | 2.2800E−02 | 3.1187E−02 | 7.5480E−01 | 2.4722E−03 | 2.5323 |
| MOPSO | 1.3103E−02 | 1.3089E−02 | 7.4815E−01 | 1.1422E−02 | 2.7469 |
| MOMVO | 1.4534E−02 | 2.1363E−03 | 7.5146E−01 | 1.9507E−03 | 1.3807 |
| MODA | 4.4022E−02 | 4.3926E−02 | 7.4247E−01 | 6.5398E−03 | 11.7750 |
| MOGOA | 1.8177E−02 | 9.8555E−03 | 7.3807E−01 | 1.3624E−02 | 13.5750 |

**Table 16**
Results of MOO algorithms in terms of SP and HV metrics for speed reducer design problem.

| Algorithms | SP | | HV | | Times |
|---|---|---|---|---|---|
| | Average | SD | Average | SD | |
| MOSGA | **7.3202E−03** | 3.8366E−03 | **9.6963E−01** | **3.7058E−03** | 0.9583 |
| NSGA-II | 1.5198E−02 | 3.9273E−03 | 9.5687E−01 | 4.4127E−03 | 17.7969 |
| MOPSO | 1.3004E−02 | 8.5665E−03 | 9.6881E−01 | 4.2488E−03 | 2.1125 |
| MOMVO | 1.7613E−02 | 6.4886E−03 | 9.5951E−01 | 4.4804E−03 | **0.8234** |
| MODA | 3.2642E−02 | 2.2231E−02 | 8.8848E−01 | 1.9508E−02 | 16.2354 |
| MOGOA | 8.8099E−03 | **2.5960E−03** | 9.4094E−01 | 3.7759E−02 | 24.1703 |

**Table 17**
Results of MOO algorithms in terms of SP and HV metrics for disk brake design problem.

| Algorithms | SP | | HV | | Times |
|---|---|---|---|---|---|
| | Average | SD | Average | SD | |
| MOSGA | **7.3538E−03** | **6.7964E−04** | **8.9578E−01** | **8.2402E−04** | **0.8943** |
| NSGA-II | 3.0740E−02 | 4.4559E−02 | 8.9190E−01 | 1.2560E−03 | 4.2214 |
| MOPSO | 1.8261E−02 | 1.2020E−02 | 8.8750E−01 | 2.4272E−03 | 3.2125 |
| MOMVO | 3.8599E−02 | 6.4632E−02 | 8.8622E−01 | 2.1166E−03 | 1.2214 |
| MODA | 3.1079E−02 | 2.1984E−02 | 8.7068E−01 | 6.7012E−03 | 10.2063 |
| MOGOA | 1.2720E−02 | 3.3030E−03 | 8.8217E−01 | 4.9237E−03 | 12.5995 |

**Table 18**
Results of MOO algorithms in terms of SP and HV metrics for welded beam design problem.

| Algorithms | SP | | HV | | Times |
|---|---|---|---|---|---|
| | Average | SD | Average | SD | |
| MOSGA | **7.0178E−03** | **6.5459E−04** | **9.2794E−01** | **2.0136E−03** | **0.9510** |
| NSGA-II | 2.9740E−02 | 2.5266E−02 | 9.1330E−01 | 6.5717E−03 | 4.7370 |
| MOPSO | 1.5732E−02 | 1.4279E−02 | 9.1556E−01 | 8.3464E−03 | 3.2474 |
| MOMVO | 1.4265E−02 | 5.7283E−03 | 9.1697E−01 | 5.7954E−03 | 1.2250 |
| MODA | 2.3166E−02 | 1.4875E−02 | 9.0618E−01 | 1.2988E−02 | 10.2797 |
| MOGOA | 5.6490E−02 | 8.5288E−02 | 8.7955E−01 | 5.3347E−02 | 13.8604 |

**Table 19**
Results of MOO algorithms in terms of SP and HV metrics for spring design problem.

| Algorithms | SP | | HV | | Times |
|---|---|---|---|---|---|
| | Average | SD | Average | SD | |
| MOSGA | **1.1955E−02** | **7.2016E−03** | **7.9475E−01** | **5.0796E−03** | 0.9802 |
| NSGA-II | 1.2663E−02 | 8.3666E−03 | 7.6466E−01 | 1.2696E−02 | 15.5917 |
| MOPSO | 1.7378E−02 | 1.6009E−02 | 7.9050E−01 | 8.8883E−03 | 1.3380 |
| MOMVO | 1.9543E−02 | 1.1880E−02 | 7.8495E−01 | 2.1550E−02 | **0.6870** |
| MODA | 1.3744E−01 | 1.0520E−01 | 6.4578E−01 | 7.7132E−02 | 7.0547 |
| MOGOA | 8.0208E−02 | 4.8762E−02 | 7.3452E−01 | 3.5101E−02 | 12.9464 |

**Table 20**
The average ranking of MOO algorithms by Friedman test for multi-objective engineering problems.

| Algorithms | SP | HV |
|---|---|---|
| MOSGA | **1.7200** | **1.2333** |
| NSGA-II | 3.3200 | 3.2533 |
| MOPSO | 3.0200 | 2.8000 |
| MOMVO | 3.9067 | 3.2800 |
| MODA | 5.1667 | 5.4667 |
| MOGOA | 3.8667 | 4.9667 |

capabilities of the MOSGA. The analysis of the results underscores the ability of the MOSGA is capable of solving problems with two and three objectives with convex, non-convex, and discontinuous Pareto optimal fronts. It is encouraged to develop and apply MOSGA to different real-world engineering problems for future works. Moreover, MOSGA should be extended and improved to solve many-objective problems.

## CRediT authorship contribution statement

**Truong Hoang Bao Huy:** Conceptualization, Methodology, Software, Writing – original draft. **Perumal Nallagownden:** Writing – review & editing. **Khoa Hoang Truong:** Resources. **Ramani Kannan:** Writing – review & editing. **Dieu Ngoc Vo:** Writing – review & editing, Supervision. **Nguyen Ho:** Resources.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.


## Acknowledgment

We acknowledge the support of time and facilities from Ho Chi Minh City University of Technology (HCMUT), VNU-HCM for this study.


## Appendix A. Multi-objective benchmark test problems

Tables A.1–A.3 presents the mathematical formulations of multi-objective benchmark test problems.

## Appendix B. Boxplot analysis of performance metrics

Figs. B.1–B.4 present the boxplot analysis of IGD, SP, $\Delta$, and HV obtained by all algorithms for twenty-five benchmark functions.

## Appendix C. Pareto optimal front of benchmark test functions

Figs. C.1 and C.2 present Pareto fronts obtained by MOSGA for WFG and DTLZ test suites.

## Appendix D. Multi-objective engineering design problems

### Four-Bar Truss Design Problem

The target of this classical engineering problem is to optimize the structural weight and joint displacement of a truss with four bars simultaneously. The cross-sectional areas of the structural members (1, 2, 3, and 4) are determined as four continuous

**Table 21**
Effects of perturbation constant ($\alpha^k$) on the IGD, SP, $\Delta$, and HV metrics for test functions BINH1 and BINH2.

| Parameters | IGD | | SP | | $\Delta$ | | HV | | Times |
|---|---|---|---|---|---|---|---|---|---|
| | Average | SD | Average | SD | Average | SD | Average | SD | |
| **BINH1** | | | | | | | | | |
| $\alpha^k = 1$ | 2.6256E−04 | 8.7493E−06 | 8.5644E−03 | 6.1561E−04 | 4.6536E−01 | 2.9696E−02 | 8.5825E−01 | 2.1762E−04 | 0.817 |
| $\alpha^k = 2$ | 2.5735E−04 | 1.1635E−05 | 7.7850E−03 | 5.4317E−04 | 4.2317E−01 | 3.2684E−02 | 8.5854E−01 | 1.9897E−04 | 0.829 |
| $\alpha^k = 3$ | **2.5152E−04** | 1.3499E−05 | **7.4010E−03** | 5.9414E−04 | **3.9194E−01** | 3.3539E−02 | **8.5862E−01** | 1.9522E−04 | 0.662 |
| $\alpha^k = 4$ | 2.5196E−04 | 1.4202E−05 | 7.9424E−03 | 6.3199E−04 | 4.1031E−01 | 3.7957E−02 | 8.5853E−01 | **1.9321E−04** | 0.547 |
| $\alpha^k = 5$ | 2.5190E−04 | **8.6468E−06** | 7.9540E−03 | **3.9971E−04** | 4.1244E−01 | **2.7087E−02** | 8.5854E−01 | 2.1388E−04 | **0.506** |
| **BINH2** | | | | | | | | | |
| $\alpha^k = 1$ | 7.6065E−05 | 1.4936E−05 | 9.4292E−03 | **4.0176E−04** | 5.5961E−01 | 3.2936E−02 | 8.4240E−01 | 2.0682E−04 | 1.127 |
| $\alpha^k = 2$ | 7.1019E−05 | 2.9504E−06 | 9.2464E−03 | 4.6750E−04 | 5.1121E−01 | **2.8786E−02** | 8.4254E−01 | 2.0770E−04 | **0.696** |
| $\alpha^k = 3$ | **7.0076E−05** | 5.7899E−06 | **8.3983E−03** | 6.8836E−04 | **4.3699E−01** | 3.2655E−02 | **8.4313E−01** | **1.3995E−04** | 0.989 |
| $\alpha^k = 4$ | 7.1042E−05 | 2.5249E−06 | 8.9313E−03 | 5.3394E−04 | 4.9476E−01 | 3.1094E−02 | 8.4265E−01 | 1.9215E−04 | 0.888 |
| $\alpha^k = 5$ | 7.2148E−05 | **2.4649E−06** | 9.2776E−03 | 5.0865E−04 | 5.2670E−01 | 3.1048E−02 | 8.4251E−01 | 2.0682E−04 | 1.243 |

**Table 22**
Effects of the number of search group members ($n_g$) on the IGD, SP, $\Delta$, and HV metrics for test functions BINH1 and BINH2.

| Parameters | IGD | | SP | | $\Delta$ | | HV | | Times |
|---|---|---|---|---|---|---|---|---|---|
| | Average | SD | Average | SD | Average | SD | Average | SD | |
| **BINH1** | | | | | | | | | |
| $n_g = 10$ | 2.5512E−04 | 1.2190E−05 | **6.1138E−03** | **4.9706E−04** | **3.7299E−01** | **1.6388E−02** | 8.5853E−01 | 2.0486E−04 | 0.838 |
| $n_g = 15$ | 2.5358E−04 | 2.1957E−05 | 6.5989E−03 | 5.8846E−04 | 3.7387E−01 | 2.9428E−02 | 8.5863E−01 | 2.1053E−04 | 0.598 |
| $n_g = 20$ | **2.5152E−04** | **1.3499E−05** | 7.4010E−03 | 5.9414E−04 | 3.9194E−01 | 3.3539E−02 | **8.5862E−01** | **1.9522E−04** | 0.662 |
| $n_g = 25$ | 2.5776E−04 | 1.4516E−05 | 7.6622E−03 | 6.0086E−04 | 4.1991E−01 | 3.0587E−02 | 8.5860E−01 | 2.1500E−04 | **0.515** |
| $n_g = 30$ | 2.6178E−04 | 9.6556E−06 | 8.1284E−03 | 6.1160E−04 | 4.4374E−01 | 2.9288E−02 | 8.5852E−01 | 1.8325E−04 | 0.593 |
| **BINH2** | | | | | | | | | |
| $n_g = 10$ | 7.1953E−05 | 4.5675E−06 | **7.0788E−03** | 7.2036E−04 | **4.1058E−01** | **2.5059E−02** | 8.4272E−01 | 2.6710E−04 | **0.845** |
| $n_g = 15$ | 7.1257E−05 | 3.2004E−06 | 7.7936E−03 | **3.7331E−04** | 4.2879E−01 | 2.5133E−02 | 8.4283E−01 | 2.3919E−04 | 0.947 |
| $n_g = 20$ | 7.0076E−05 | 5.7899E−06 | 8.3983E−03 | 6.8836E−04 | 4.3699E−01 | 3.2655E−02 | **8.4313E−01** | **1.3995E−04** | 0.989 |
| $n_g = 25$ | 6.9576E−05 | 3.9754E−06 | 8.5857E−03 | 6.0789E−04 | 4.6150E−01 | 3.3187E−02 | 8.4275E−01 | 2.8642E−04 | 0.893 |
| $n_g = 30$ | **6.8571E−05** | **1.9359E−06** | 8.9575E−03 | 5.6170E−04 | 4.8835E−01 | 3.1339E−02 | 8.4265E−01 | 1.9750E−04 | 0.905 |

**Table 23**
Effects of the global iteration ratio (*GIR*) on the IGD, SP, $\Delta$, and HV metrics for test functions BINH1 and BINH2.

| Parameters | IGD | | SP | | $\Delta$ | | HV | | Times |
|---|---|---|---|---|---|---|---|---|---|
| | Average | SD | Average | SD | Average | SD | Average | SD | |
| **BINH1** | | | | | | | | | |
| $GIR = 1$ | 2.5969E−04 | 2.6335E−05 | **6.4980E−03** | 9.6961E−04 | **3.5687E−01** | 3.8197E−02 | 8.5833E−01 | 2.6648E−04 | 1.309 |
| $GIR = 0.3$ | **2.5152E−04** | 1.3499E−05 | 7.4010E−03 | 5.9414E−04 | 3.9194E−01 | 3.3539E−02 | **8.5862E−01** | 1.9522E−04 | 0.662 |
| $GIR = 0$ | 2.5173E−04 | **8.6047E−06** | 7.7312E−03 | **5.5441E−04** | 4.1236E−01 | **2.5284E−02** | 8.5858E−01 | **1.4081E−04** | **0.484** |
| **BINH2** | | | | | | | | | |
| $GIR = 1$ | 7.4687E−05 | 7.9149E−06 | **7.1561E−03** | 6.3944E−04 | **3.9546E−01** | **2.7269E−02** | 8.4248E−01 | 4.5496E−04 | 1.468 |
| $GIR = 0.3$ | 7.0076E−05 | 5.7899E−06 | 8.3983E−03 | 6.8836E−04 | 4.3699E−01 | 3.2655E−02 | **8.4313E−01** | **1.3995E−04** | 0.989 |
| $GIR = 0$ | **6.8981E−05** | **5.5783E−06** | 8.5827E−03 | **5.5625E−04** | 4.5101E−01 | 3.2407E−02 | 8.4279E−01 | 2.8576E−04 | **0.776** |

design variables. The problem is mathematically described below [64,65]:

$$\text{Minimize}: \begin{cases} f_1(x) = L(2x_1 + \sqrt{2x_2} + \sqrt{x_3} + x_4) \\ f_2(x) = \dfrac{FL}{E}\left(\dfrac{2}{x_2} + \dfrac{2\sqrt{2}}{x_2} - \dfrac{2\sqrt{2}}{x_3} + \dfrac{2}{x_4}\right) \end{cases} \tag{D.1}$$

$$\text{Variable range}: \left(\frac{F}{\sigma}\right) \le x_1 \le 3 \times \left(\frac{F}{\sigma}\right) \tag{D.2}$$

$$\sqrt{2} \times \left(\frac{F}{\sigma}\right) \le x_2 \le 3 \times \left(\frac{F}{\sigma}\right) \tag{D.3}$$

$$\sqrt{2} \times \left(\frac{F}{\sigma}\right) \le x_3 \le 3 \times \left(\frac{F}{\sigma}\right) \tag{D.4}$$

$$\left(\frac{F}{\sigma}\right) \le x_4 \le 3 \times \left(\frac{F}{\sigma}\right) \tag{D.5}$$

$$\text{where}: \quad F = 10\ \text{kN}, \quad E = 2 \times 10^5\ \text{kN/cm}^2 \tag{D.6}$$

$$L = 200\ \text{cm}, \quad \sigma = 10\ \text{kN/cm}^3 \tag{D.7}$$

### Speed Reducer Design Problem

This problem was extensively studied in the field of mechanical optimization. The weight of the gear assembly and the transverse deflection of the shaft are minimized simultaneously. This problem includes seven design variables as well as eleven constraints, which is formulated as follows [66]:

$$\text{Minimize}: \begin{cases} f_1(x) = 0.7854x_1x_2^2(3.3333x_3^2 \\ \quad +14.9334x_3 - 43.0934) \\ \quad -1.508x_1(x_6^2 + x_7^2) + 7.4777(x_6^3 + x_7^3) \\ \quad +0.7854(x_4x_6^2 + x_5x_7^2) \\ f_2(x) = \dfrac{\sqrt{(745x_4/x_2x_3)^2 + 1.69 \times 10^7}}{0.1x_6^3} \end{cases} \tag{D.8}$$

**Table A.1**
Definitions for test functions ZDT1, ZDT2, ZDT3, ZDT6, BINH1, DEB1, DEB2, DEB3, FON1, and FON2.

| Problem | Definition | Constraints |
|---|---|---|
| ZDT1 | $f_1(x) = x_1$<br>$f_2(x) = g(x)\left[1 - \sqrt{\frac{f_1}{g(x)}}\right]$<br>$g(x) = 1 + \frac{9}{n-1}\sum_{i=2}^{n} x_i$ | $n = 30$<br>$0 \le x_i \le 1$<br>$i = 1, 2, \dots, 30$ |
| ZDT2 | $f_1(x) = x_1$<br>$f_2(x) = g(x)\left[1 - \left(\frac{f_1}{g(x)}\right)^2\right]$<br>$g(x) = 1 + \frac{9}{n-1}\sum_{i=2}^{n} x_i$ | $n = 30$<br>$0 \le x_i \le 1$<br>$i = 1, 2, \dots, 30$ |
| ZDT3 | $f_1(x) = x_1$<br>$f_2(x) = g(x)\left[1 - \sqrt{\frac{f_1}{g(x)}} - \frac{f_1}{g(x)}\sin(10\pi f_1)\right]$<br>$g(x) = 1 + \frac{9}{n-1}\sum_{i=2}^{n} x_i$ | $n = 30$<br>$0 \le x_i \le 1$<br>$i = 1, 2, \dots, 30$ |
| ZDT6 | $f_1(x) = 1 - \exp(-4x_1)\sin^6(6\pi x_1)$<br>$f_2(x) = g(x)\left[1 - \left(\frac{f_1}{g(x)}\right)^2\right]$<br>$g(x) = 1 + 9\left(\frac{\sum_{i=2}^{n} x_i}{n-1}\right)^{0.25}$ | $n = 10$<br>$0 \le x_i \le 1$<br>$i = 1, 2, \dots, 10$ |
| BINH1 | $f_1(x, y) = x^2 + y^2$<br>$f_2(x, y) = (x-5)^2 + (y-5)^2$ | $-5 \le x, y \le 10$ |
| DEB1 | $f_1(x) = x_1$<br>$f_2(x) = g(x) \cdot h(x)$<br>$g(x) = 1 + x_2^2$<br>$h(x) = \begin{cases} 1 - \left(\frac{f_1(x)}{g(x)}\right)^2, & \text{if } f_1 \le g; \\ 0, & \text{otherwise} \end{cases}$ | $0 \le x_i \le 1$<br>$i = 1, 2$ |
| DEB2 | $f_1(x) = x_1$<br>$f_2(x) = g(x) \cdot h(x)$<br>$g(x) = 1 + 10x_2$<br>$h(x) = 1 - \left(\frac{f_1(x)}{g(x)}\right)^2 - \frac{f_1(x)}{g(x)}\sin(12\pi f_1)$ | $0 \le x_i \le 1$<br>$i = 1, 2$ |
| DEB3 | $f_1(x) = 1 - e^{(-4x_1)}\sin^4(10\pi x_1)$<br>$f_2(x) = g(x) \cdot h(x)$<br>$g(x) = 1 + x_2^2$<br>$h(x) = \begin{cases} 1 - \left(\frac{f_1(x)}{g(x)}\right)^{10}, & \text{if } f_1 \le g; \\ 0, & \text{otherwise} \end{cases}$ | $0 \le x_i \le 1$<br>$i = 1, 2$ |
| FON1 | $f_1(x, y) = 1 - \exp(-(x-1)^2 - (y+1)^2)$<br>$f_2(x, y) = 1 - \exp(-(x+1)^2 - (y-1)^2)$ | $-4 \le x, y \le 4$ |
| FON2 | $f_1(x) = 1 - \exp\left[-\sum_{i=1}^{n}\left(x_i - \frac{1}{\sqrt{n}}\right)^2\right]$<br>$f_2(x) = 1 - \exp\left[-\sum_{i=1}^{n}\left(x_i + \frac{1}{\sqrt{n}}\right)^2\right]$ | $n = 3$<br>$-4 \le x_i \le 4$<br>$i = 1, 2, 3$ |

Subject to :

$$g_1(x) = \frac{27}{x_1 x_2^2 x_3} - 1 \le 0 \tag{D.9}$$

$$g_2(x) = \frac{397.5}{x_1 x_2^2 x_3^2} - 1 \le 0 \tag{D.10}$$

$$g_3(x) = \frac{1.93x_4^3}{x_2 x_3 x_6^4} - 1 \le 0 \tag{D.11}$$

$$g_4(x) = \frac{1.93x_5^3}{x_2 x_3 x_7^4} - 1 \le 0 \tag{D.12}$$

**Table A.2**
Mathematical formulations for test functions KUR, LAU, MUR, POL, SCH1, SCH2, VN1, VN2, and VN3.

| Problem | Definition | Constraints |
|---|---|---|
| KUR | $f_1(x) = \sum_{i=1}^{n-1}\left[-10\exp\left(-0.2\sqrt{x_i^2 + x_{i+1}^2}\right)\right]$ <br> $f_2(x) = \sum_{i=1}^{n}\left(\lvert x_i\rvert^{0.8} + 5\sin x_i^3\right)$ | $n = 3$ <br> $-5 \le x_i \le 5$ <br> $i = 1, 2, 3$ |
| LAU | $f_1(x, y) = x^2 + y^2$ <br> $f_2(x, y) = (x + 2)^2 + y^2$ | $-50 \le x, y \le 50$ |
| MUR | $f_1(x, y) = 2\sqrt{x}$ <br> $f_2(x, y) = x(1 - y) + 5$ | $1 \le x \le 4$ <br> $1 \le y \le 2$ |
| POL | $f_1(x, y) = 1 + (A_1 - B_1)^2 + (A_2 - B_2)^2$ <br> $f_2(x, y) = (x + 3)^2 + (y + 1)^2$ <br> $A_1 = 0.5\sin(1) - 2\cos(1) + \sin(2) - 1.5\cos(2)$ <br> $A_2 = 1.5\sin(1) - \cos(1) + 2\sin(2) - 0.5\cos(2)$ <br> $B_1 = 0.5\sin x - 2\cos x + \sin y - 1.5\cos y$ <br> $B_2 = 1.5\sin x - \cos x + 2\sin y - 0.5\cos y$ | $-\pi \le x, y \le \pi$ |
| SCH1 | $f_1(x) = x^2$ <br> $f_2(x) = (x - 2)^2$ | $-1000 \le x \le 1000$ |
| SCH2 | $f_1(x) = \begin{cases} -x, & \text{if } x \le 1, \\ -2 + x, & \text{if } 1 < x \le 3, \\ 4 - x, & \text{if } 3 < x \le 4, \\ -4 + x, & \text{if } x > 4, \end{cases}$ <br> $f_2(x) = (x - 5)^2$ | $-5 \le x \le 10$ |
| VN1 | $f_1(x, y) = x^2 + (y - 1)^2$ <br> $f_2(x, y) = x^2 + (y + 1)^2 + 1$ <br> $f_3(x, y) = (x - 1)^2 + y^2 + 2$ | $-2 \le x, y \le 2$ |
| VN2 | $f_1(x, y) = \frac{(x - 2)^2}{2} + \frac{(y + 1)^2}{13} + 3$ <br> $f_2(x, y) = \frac{(x + y - 3)^2}{36} + \frac{(-x + y + 2)^2}{8} - 17$ <br> $f_3(x, y) = \frac{(x + 2y - 1)^2}{175} + \frac{(2y - x)^2}{17} - 13$ | $-4 \le x, y \le 4$ |
| VN3 | $f_1(x, y) = 0.5(x^2 + y^2) + \sin(x^2 + y^2)$ <br> $f_2(x, y) = \frac{(3x - 2y + 4)^2}{8} + \frac{(x - y + 1)^2}{27} + 15$ <br> $f_3(x, y) = \frac{1}{(x^2 + y^2 + 1)} - 1.1e^{(-x^2 - y^2)}$ | $-3 \le x, y \le 3$ |

$$g_5(x) = \frac{1}{110x_6^3}\sqrt{\left(\frac{745x_4}{x_2x_3}\right)^2 + 16.9 \times 10^6} - 1 \le 0 \tag{D.13}$$

$$g_6(x) = \frac{1}{85x_7^3}\sqrt{\left(\frac{745x_5}{x_2x_3}\right)^2 + 157.5 \times 10^6} - 1 \le 0 \tag{D.14}$$

$$g_7(x) = \frac{x_2x_3}{40} - 1 \le 0 \tag{D.15}$$

$$g_8(x) = \frac{5x_2}{x_1} - 1 \le 0 \tag{D.16}$$

$$g_9(x) = \frac{x_1}{12x_2} - 1 \le 0 \tag{D.17}$$

$$g_{10}(x) = \frac{1.5x_6 + 1.9}{x_4} - 1 \le 0 \tag{D.18}$$

$$g_{11}(x) = \frac{1.1x_7 + 1.9}{x_5} - 1 \le 0 \tag{D.19}$$

Variable range :

$$2.6 \le x_1 \le 3.6 \tag{D.20}$$

$$0.7 \le x_2 \le 0.8 \tag{D.21}$$

$$17 \le x_3 \le 28 \tag{D.22}$$

$$7.3 \le x_4, x_5 \le 8.3 \tag{D.23}$$

$$2.9 \le x_6 \le 3.9 \tag{D.24}$$

$$5 \le x_7 \le 5.5 \tag{D.25}$$

**Table A.3**
Mathematical formulations for test functions BEL, BINH2, CONSTR, KITA, SRN, and TNK.

| Problem | Definition | Constraints |
|---|---|---|
| BEL | $f_1(x,y) = -2x + y$<br>$f_2(x,y) = 2x + y$ | $0 \le x \le 5$<br>$0 \le y \le 3$<br>$-x + y - 1 \le 0$<br>$x + y - 7 \le 0$ |
| BINH2 | $f_1(x,y) = 4x^2 + 4y^2$<br>$f_2(x,y) = (x-5)^2 + (y-5)^2$ | $0 \le x \le 5$<br>$0 \le y \le 3$<br>$(x-5)^2 + y^2 - 25 \le 0$<br>$-(x-8)^2 - (y+3)^2 + 7.7 \le 0$ |
| CONSTR | $f_1(x,y) = x$<br>$f_2(x,y) = \frac{1+y}{x}$ | $0.1 \le x \le 1$<br>$0 \le y \le 5$<br>$-9x - y + 6 \le 0$<br>$-9x + y + 1 \le 0$ |
| KITA | $\max\ f_1(x,y) = -x^2 + y$<br>$\max\ f_2(x,y) = \frac{1}{2}x + y + 1$ | $0 \le x, y \le 7$<br>$\frac{1}{6}x + y - \frac{13}{2} \le 0$<br>$\frac{1}{2}x + y - \frac{15}{2} \le 0$<br>$5x + y - 30 \le 0$ |
| SRN | $f_1(x,y) = (x-2)^2 + (y-1)^2 + 2$<br>$f_2(x,y) = 9x - (y-1)^2$ | $-20 \le x, y \le 20$<br>$x^2 + y^2 - 225 \le 0$<br>$x - 3y + 10 \le 0$ |
| TNK | $f_1(x,y) = x$<br>$f_2(x,y) = y$ | $0 \le x, y \le \pi$<br>$-x^2 - y^2 + 1 + 0.1\cos\left[16\arctan\left(\frac{x}{y}\right)\right] \le 0$<br>$\left(x - \frac{1}{2}\right)^2 + \left(y - \frac{1}{2}\right)^2 - \frac{1}{2} \le 0$ |

Among these seven design variables, the third variable ($x_3$) is a discrete integer variable, and the remaining variables are continuous.

**Disk Brake Design Problem**

This optimization problem was originally suggested by Ray and Liew [67], in which the stopping time and the mass of a brake are optimized based on five inequality constraints. The problem is expressed in the following equations:

$$\text{Minimize}: \begin{cases} f_1(x) = 4.9 \times 10^{-5}(x_2^2 - x_1^2)(x_4 - 1) \\ f_2(x) = \dfrac{9.82 \times 10^6 (x_2^2 - x_1^2)}{x_3 x_4 (x_2^3 - x_1^3)} \end{cases} \tag{D.26}$$

$$\text{Subject to}: \quad g_1(x) = 20 + x_1 - x_2 \le 0 \tag{D.27}$$

$$g_2(x) = 2.5(x_4 + 1) - 30 \le 0 \tag{D.28}$$

$$g_3(x) = \frac{x_3}{3.14(x_2^2 - x_1^2)^2} - 0.4 \le 0 \tag{D.29}$$

$$g_4(x) = \frac{2.22 \times 10^{-3} x_3 (x_2^3 - x_1^3)}{(x_2^2 - x_1^2)^2} - 1 \le 0 \tag{D.30}$$

$$g_5(x) = 900 - \frac{2.66 \times 10^{-2} x_3 x_4 (x_2^3 - x_1^3)}{(x_2^2 - x_1^2)} \le 0 \tag{D.31}$$

$$\text{Variable range}: 55 \le x_1 \le 80 \tag{D.32}$$

$$75 \le x_2 \le 110 \tag{D.33}$$

$$1000 \le x_3 \le 3000 \tag{D.34}$$

$$2 \le x_4 \le 20 \tag{D.35}$$

where $x_1$, $x_2$, $x_3$, and $x_4$ denote the inner radius of the disk, the outer radius of the disk, the engaging force (actuating force), and the number of friction surfaces, respectively. In particular, the fourth variable ($x_4$) is discrete.

**Welded Beam Design Problem**

This problem has four design variables with five imposed constraints related to shear stress, bending stress, and buckling load [68]. The fabrication cost and the end deflection of the beam, are expected to be optimized. Hence, the mathematical formulas of this problem are defined as follows [67]:

$$\text{Minimize}: \begin{cases} f_1(x) = 1.10471 x_1^2 x_2 + 0.04811 x_3 x_4 (14 + x_2) \\ f_2\,(x) = \dfrac{4PL^3}{E x_3^3 x_4} \end{cases} \tag{D.36}$$

$$\text{Subject to}: \quad g_1(x) = \tau(x) - \tau_{\max} \le 0 \tag{D.37}$$

$$g_2(x) = \sigma(x) - \sigma_{\max} \le 0 \tag{D.38}$$

$$g_3(x) = x_1 - x_4 \le 0 \tag{D.39}$$

$$g_4(x) = P - P_c(x) \le 0 \tag{D.40}$$

$$\text{where}: \quad \tau(x) = \sqrt{(\tau')^2 + 2\tau'\tau''\frac{x_2}{2R} + (\tau'')^2} \tag{D.41}$$

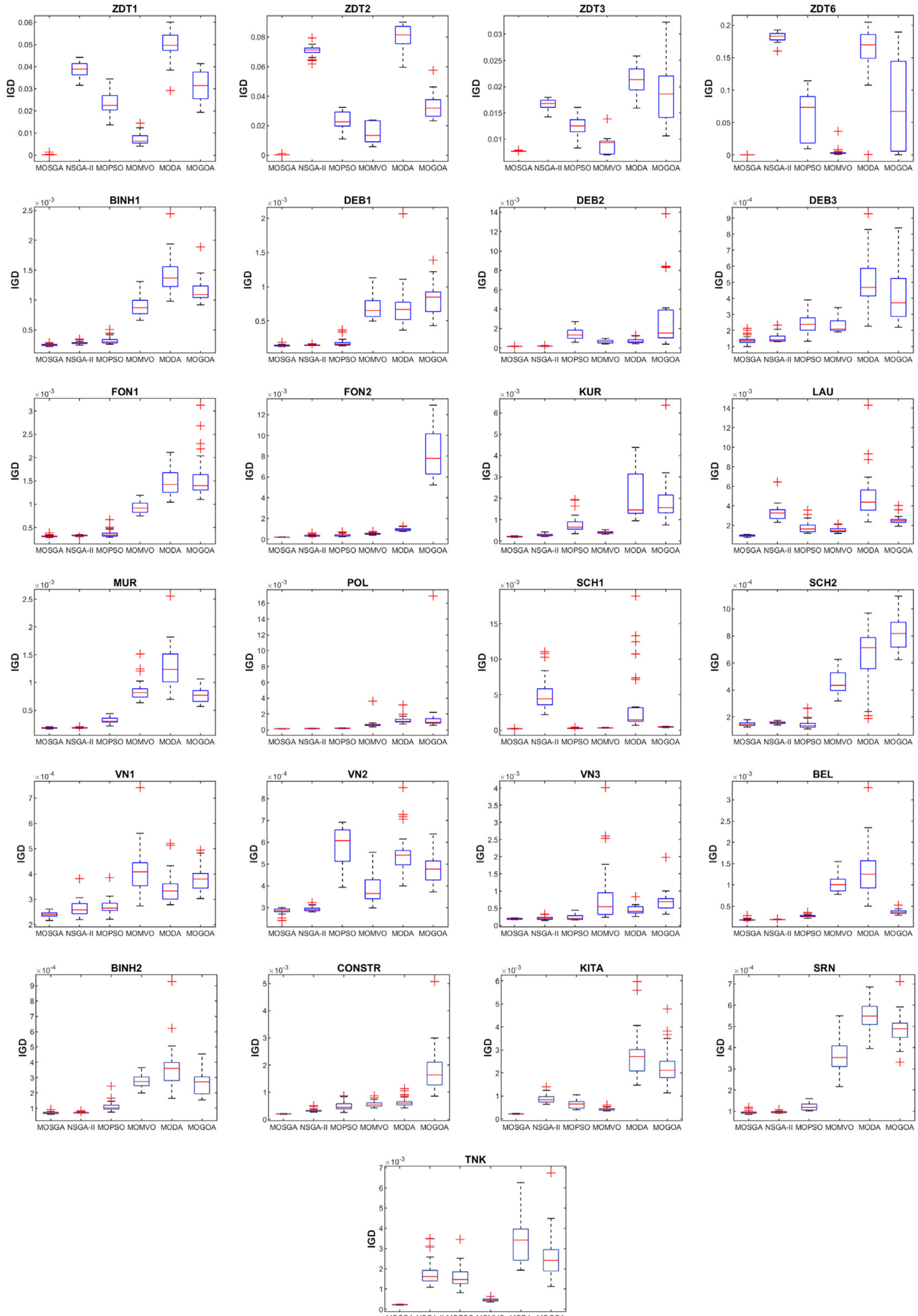



**Fig. B.1.** Statistical analysis of IGD metric for all test functions.

$$\tau' = \frac{P}{\sqrt{2}x_1x_2}, \quad \tau'' = \frac{MR}{J}, \quad M = P\left(L + \frac{x_2}{2}\right) \tag{D.42}$$

$$R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2} \tag{D.43}$$

$$J = 2\left\{\sqrt{2}x_1x_2\left[\frac{x_2^2}{12} + \left(\frac{x_1 + x_3}{2}\right)^2\right]\right\} \tag{D.44}$$

$$\sigma(x) = \frac{6PL}{x_4x_3^2}, \delta(x) = \frac{4PL^3}{Ex_4x_3^3} \tag{D.45}$$

$$P_c(x) = \frac{4.013E\sqrt{\frac{x_3^2x_4^6}{36}}}{L^2}\left(1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}}\right) \tag{D.46}$$

$P = 6000$ lb, $\quad L = 14$ in, $\quad E = 30 \times 10^6$ psi,

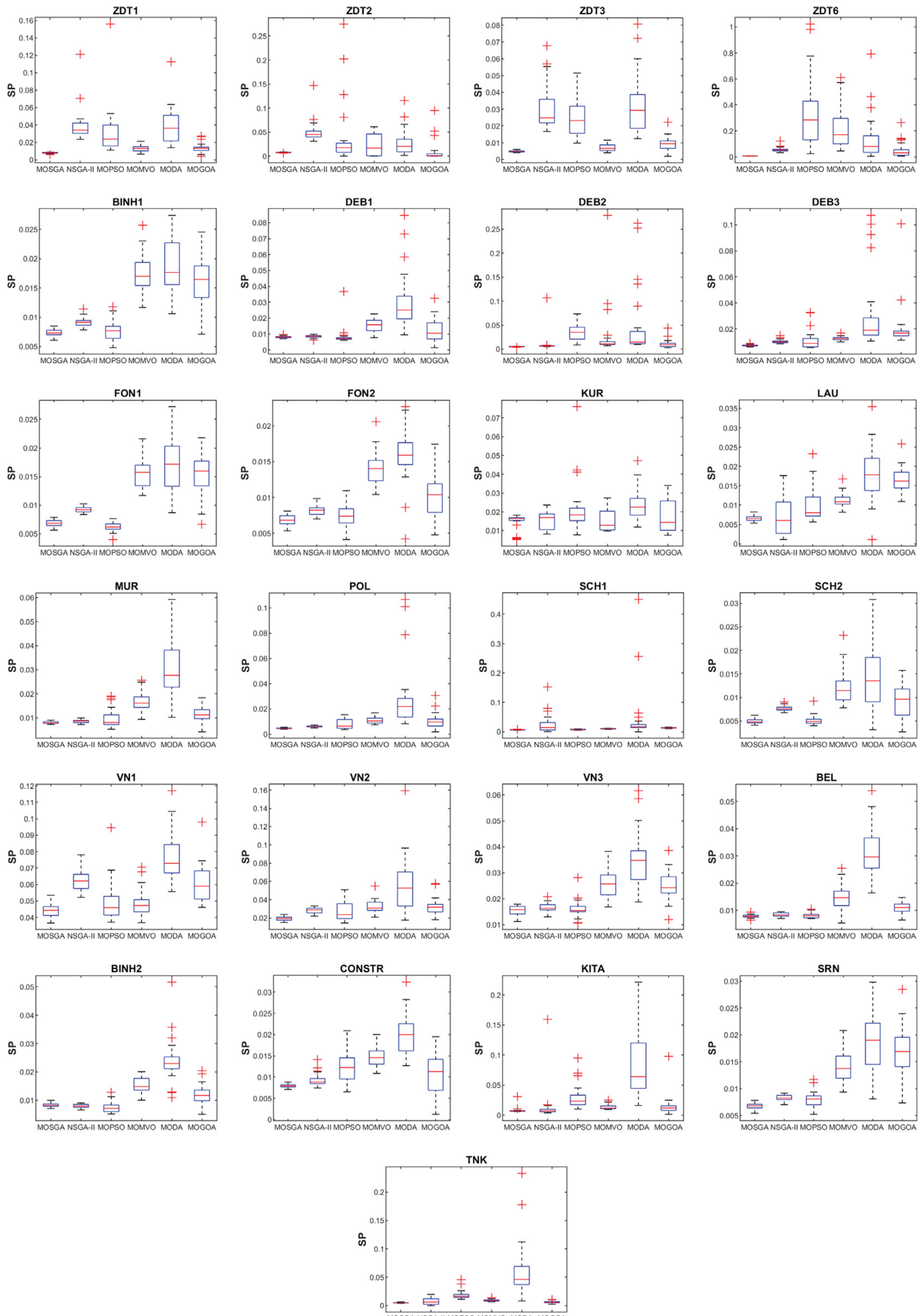



**Fig. B.2.** Statistical analysis of SP metric for all test functions.

$$G = 12 \times 10^6 \text{ psi} \tag{D.47}$$

$$\tau_{\max} = 13\,600 \text{ psi}, \quad \sigma_{\max} = 30\,000 \text{ psi},$$

$$\delta_{\max} = 0.25 \text{ in} \tag{D.48}$$

$$\text{Variable range}: 0.125 \le x_1, x_4 \le 5 \tag{D.49}$$

$$0.1 \le x_2, x_3 \le 10 \tag{D.50}$$

where $x_1$, $x_2$, $x_3$, and $x_4$ denote the thickness of the weld, the length of the clamped bar, the height of the bar, and the thickness of the bar, respectively.

**Spring Design Problem**

The target of this problem is to optimize the volume and stress of spring. The design variables, in this case, involved the number of spring coils ($x_1$), wire diameter ($x_2$), and spring diameter ($x_3$).

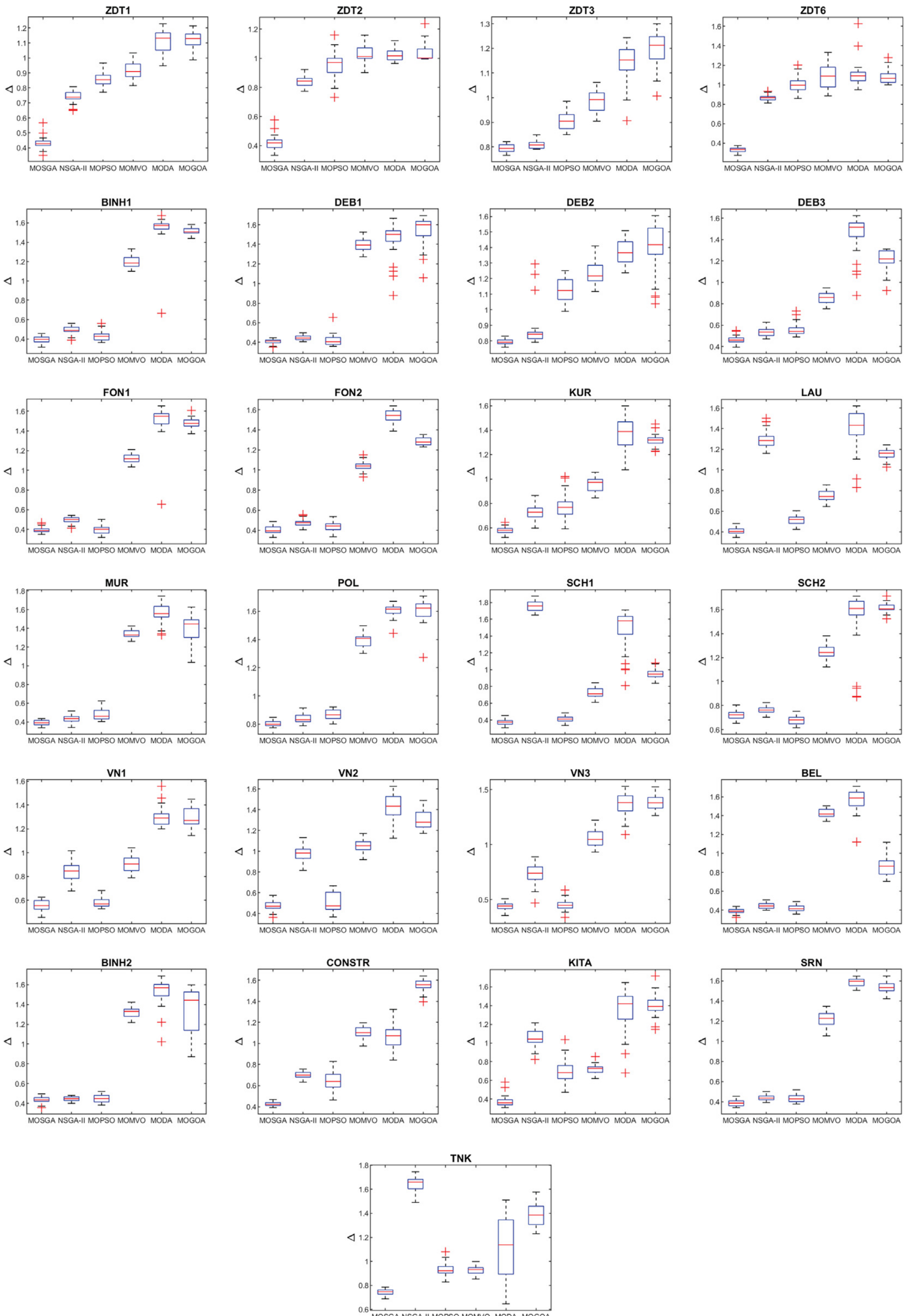



**Fig. B.3.** Statistical analysis of $\Delta$ metric for all test functions.

This is a mixed-integer-discrete optimization problem since $x_1$ is an integer variable, whereas $x_2$ is a discrete variable, and $x_3$ is a continuous variable. The constraints are imposed on minimum deflection, shear stress, surge frequency, and limits on the exterior diameter. The problem is formulated based on the following formulas [68]:

$$\text{Minimize}: \quad \begin{cases} f_1(x) = 0.25\pi^2 x_2^2 x_3(x_1+2) \\ f_2(x) = \dfrac{8KP_{\max}x_3}{\pi x_2^3} \end{cases} \tag{D.51}$$

$$\text{Subject to}: \quad g_1(x) = 1.05x_2(x_1+2) + \frac{P_{\max}}{k} - l_{\max} \le 0 \tag{D.52}$$

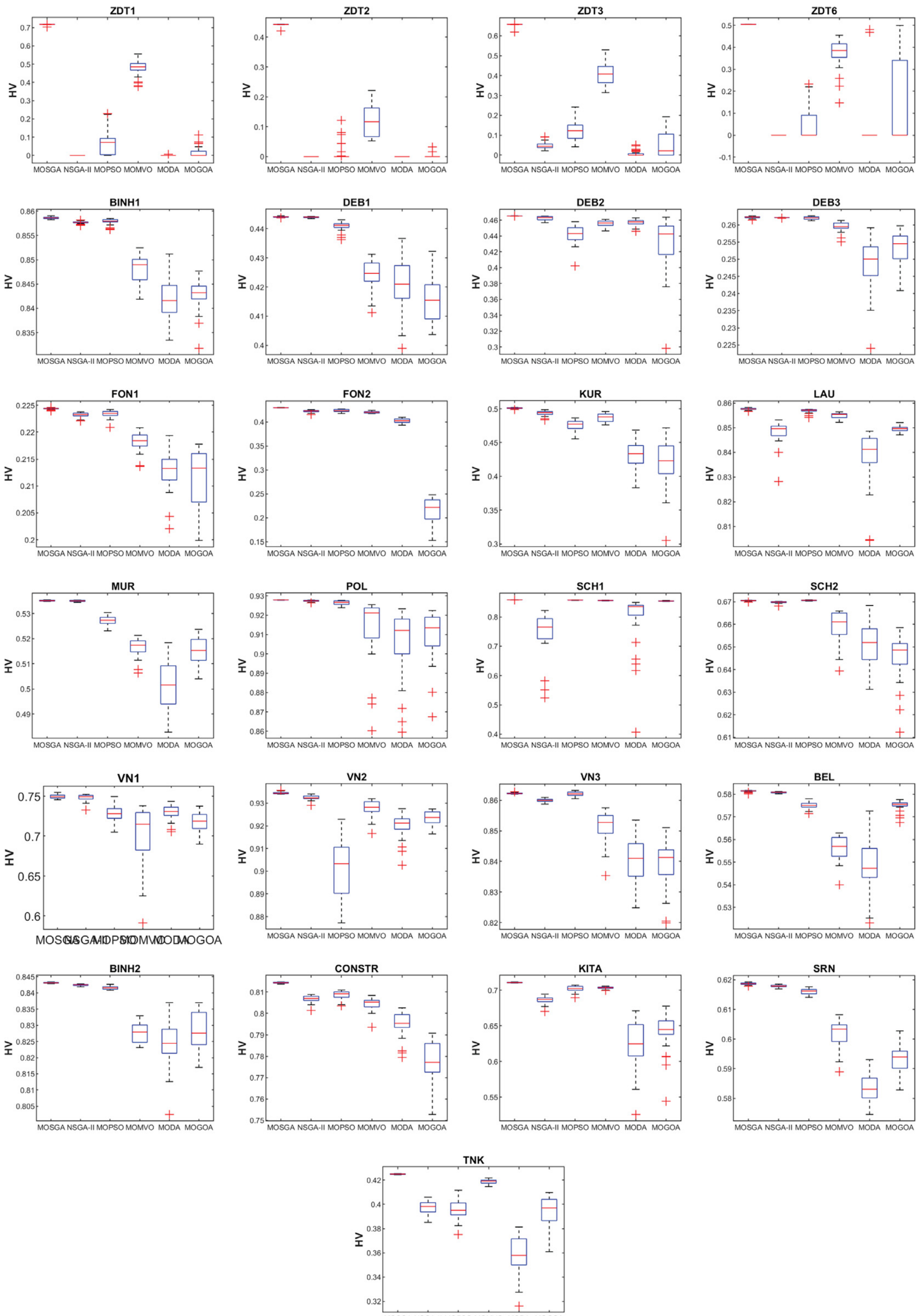



**Fig. B.4.** Statistical analysis of HV metric for all test functions.

$$g_2(x) = d_{\min} - x_2 \leq 0 \tag{D.53}$$

$$g_3(x) = x_2 + x_3 - D_{\max} \leq 0 \tag{D.54}$$

$$g_4(x) = 3 - C \leq 0 \tag{D.55}$$

$$g_5(x) = \delta_p - \delta_{pm} \leq 0 \tag{D.56}$$

$$g_6(x) = \delta_w - \frac{P_{\max} - P}{k} \leq 0 \tag{D.57}$$

$$g_7(x) = \frac{8KP_{\max}x_3}{\pi x_2^3} - S \leq 0 \tag{D.58}$$

$$g_8(x) = [0.25\pi^2 x_2^2 x_3(x_1 + 2)] - V_{\max} \leq 0 \tag{D.59}$$

$$\text{where} : K = \frac{4C - 1}{4C - 4} + \frac{0.615x_2}{x_3}, \quad P = 300 \text{ lb},$$

$$P_{\max} = 1000 \text{ lb}, \quad L_{\max} = 14 \text{ in} \tag{D.60}$$

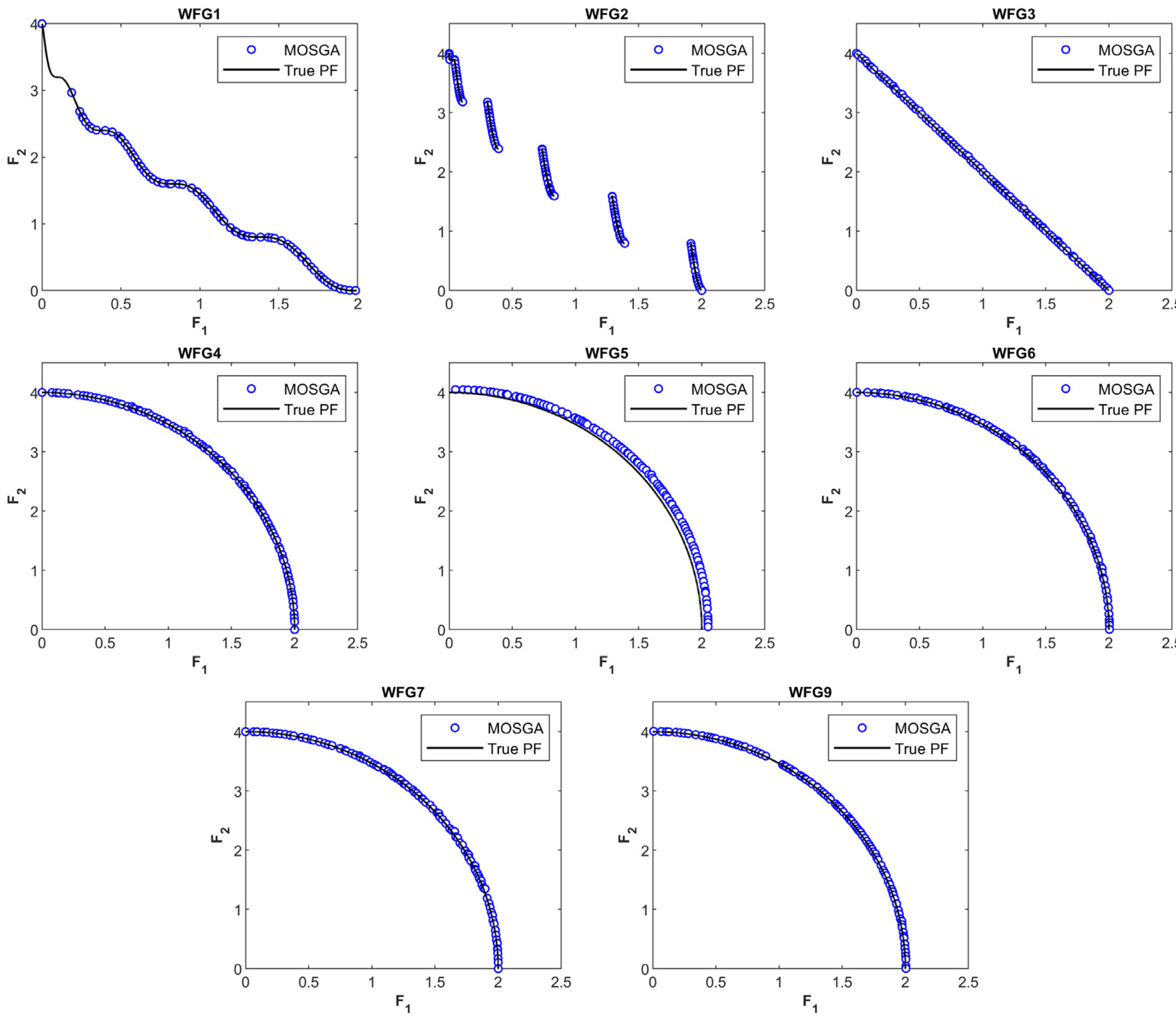



**Fig. C.1.** Pareto optimal fronts generated by MOSGA for test functions WFG test suite.

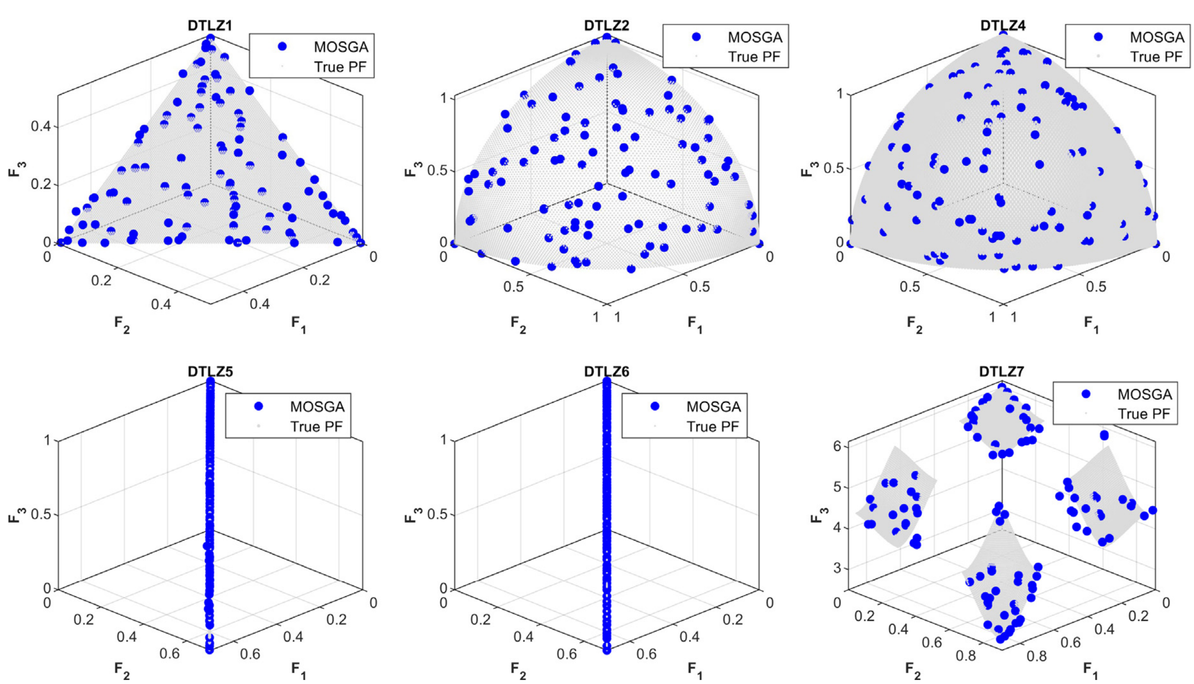



**Fig. C.2.** Pareto optimal fronts generated by MOSGA for test functions DTLZ test suite.

$$k = \frac{Gx_2^4}{8x_1x_3^3}, \quad d_{\min} = 0.2 \text{ in}, \quad D_{\max} = 3 \text{ in},$$
$$\delta_{pm} = 6 \text{ in}, \quad \delta_w = 1.25 \text{ in} \tag{D.61}$$
$$\delta_p = \frac{P}{k}, \quad C = \frac{x_3}{x_2}, \quad S = 189{,}000 \text{ psi},$$
$$G = 11{,}500{,}000 \text{ lb/in}^2, \quad V_{\max} = 30 \text{ in}^3 \tag{D.62}$$

$$\text{Variable range}: \quad 1 \le x_1 \le 32 \tag{D.63}$$
$$x_2 \in I \tag{D.64}$$
$$1 \le x_3 \le 30 \tag{D.65}$$

where $I = [0.009, 0.0095, 0.0104, 0.0118, 0.0128, 0.0132, 0.014, 0.015, 0.0162, 0.0173, 0.018, 0.020, 0.023, 0.025, 0.028, 0.032, 0.035, 0.041, 0.047, 0.054, 0.063, 0.072, 0.080, 0.092, 0.105, 0.120, 0.135, 0.148, 0.162, 0.177, 0.192, 0.207, 0.225, 0.244, 0.263, 0.283, 0.307, 0.331, 0.362, 0.394, 0.4375, 0.5]$.

## References


[1] A. Menchaca-Mendez, C.A. Coello Coello, GD-MOEA: A new multi-objective evolutionary algorithm based on the generational distance indicator, in: A. Gaspar-Cunha, C. Henggeler Antunes, C.C. Coello (Eds.), Evolutionary Multi-Criterion Optimization, Springer International Publishing, Cham, 2015, pp. 156–170, http://dx.doi.org/10.1007/978-3-319-15934-8_11.

[2] K. Deb, Multi-objective genetic algorithms: Problem difficulties and construction of test problems, Evol. Comput. 7 (1999) 205–230, http://dx.doi.org/10.1162/evco.1999.7.3.205.

[3] K. Deb, M. Abouhawwash, J. Dutta, An optimality theory based proximity measure for evolutionary multi-objective and many-objective optimization, in: A. Gaspar-Cunha, C. Henggeler Antunes, C.C. Coello (Eds.), Evolutionary Multi-Criterion Optimization, Springer International Publishing, Cham, 2015, pp. 18–33, http://dx.doi.org/10.1007/978-3-319-15892-1_2.

[4] C. Gil, A. Márquez, R. Baños, M.G. Montoya, J. Gómez, A hybrid method for solving multi-objective global optimization problems, J. Global Optim. 38 (2007) 265–281, http://dx.doi.org/10.1007/s10898-006-9105-1.

[5] K. Deb, Multi-Objective Optimization using Evolutionary Algorithms, John Wiley & Sons, Inc., USA, 2001.

[6] R.T. Marler, J.S. Arora, The weighted sum method for multi-objective optimization: new insights, Struct. Multidiscip. Optim. 41 (2010) 853–862, http://dx.doi.org/10.1007/s00158-009-0460-7.

[7] C.C. Coello, G.B. Lamont, D.A. van Veldhuizen, Evolutionary Algorithms for Solving Multi-Objective Problems, second ed., Springer US, 2007, http://dx.doi.org/10.1007/978-0-387-36797-2.

[8] D. Meignan, S. Knust, J.-M. Frayret, G. Pesant, N. Gaud, A review and taxonomy of interactive optimization methods in operations research, ACM Trans. Interact. Intell. Syst. 5 (2015) 17:1–17:43, http://dx.doi.org/10.1145/2808234.

[9] C.A. Coello Coello, Evolutionary multi-objective optimization: a historical view of the field, IEEE Comput. Intell. Mag. 1 (2006) 28–36, http://dx.doi.org/10.1109/MCI.2006.1597059.

[10] J.D. Schaffer, Multiple objective optimization with vector evaluated genetic algorithms, in: Proceedings of the 1st International Conference on Genetic Algorithms, L. Erlbaum Associates Inc., USA, 1985, pp. 93–100.

[11] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2002) 182–197, http://dx.doi.org/10.1109/4235.996017.

[12] C.A.C. Coello, G.T. Pulido, M.S. Lechuga, Handling multiple objectives with particle swarm optimization, IEEE Trans. Evol. Comput. 8 (2004) 256–279, http://dx.doi.org/10.1109/TEVC.2004.826067.

[13] M. Varadarajan, K.S. Swarup, Solving multi-objective optimal power flow using differential evolution, Transm. Distrib. IET Gener. 2 (2008) 720–730, http://dx.doi.org/10.1049/iet-gtd:20070457.

[14] S. Mirjalili, S. Saremi, S.M. Mirjalili, L. dos S. Coelho, Multi-objective grey wolf optimizer: A novel algorithm for multi-criterion optimization, Expert Syst. Appl. 47 (2016) 106–119, http://dx.doi.org/10.1016/j.eswa.2015.10.039.

[15] A. Sadollah, H. Eskandar, J.H. Kim, Water cycle algorithm for solving constrained multi-objective optimization problems, Appl. Soft Comput. 27 (2015) 279–298, http://dx.doi.org/10.1016/j.asoc.2014.10.042.

[16] A. Sadollah, H. Eskandar, A. Bahreininejad, J.H. Kim, Water cycle algorithm for solving multi-objective optimization problems, Soft Comput. 19 (2015) 2587–2603, http://dx.doi.org/10.1007/s00500-014-1424-4.

[17] S. Mirjalili, Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems, Neural Comput. Appl. 27 (2016) 1053–1073, http://dx.doi.org/10.1007/s00521-015-1920-1.

[18] S. Mirjalili, P. Jangir, S. Saremi, Multi-objective ant lion optimizer: a multi-objective optimization algorithm for solving engineering problems, Appl. Intell. 46 (2017) 79–95, http://dx.doi.org/10.1007/s10489-016-0825-8.

[19] S.Z. Mirjalili, S. Mirjalili, S. Saremi, H. Faris, I. Aljarah, Grasshopper optimization algorithm for multi-objective optimization problems, Appl. Intell. 48 (2018) 805–820, http://dx.doi.org/10.1007/s10489-017-1019-8.

[20] S. Mirjalili, P. Jangir, S.Z. Mirjalili, S. Saremi, I.N. Trivedi, Optimization of problems with multiple objectives using the multi-verse optimization algorithm, Knowl.-Based Syst. 134 (2017) 50–71, http://dx.doi.org/10.1016/j.knosys.2017.07.018.

[21] A. Panda, S. Pani, A symbiotic organisms search algorithm with adaptive penalty function to solve multi-objective constrained optimization problems, Appl. Soft Comput. 46 (2016) 344–360, http://dx.doi.org/10.1016/j.asoc.2016.04.030.

[22] D.H. Wolpert, W.G. Macready, No free lunch theorems for optimization, IEEE Trans. Evol. Comput. 1 (1997) 67–82, http://dx.doi.org/10.1109/4235.585893.

[23] M.S. Gonçalves, R.H. Lopez, L.F.F. Miguel, Search group algorithm: A new metaheuristic method for the optimization of truss structures, Comput. Struct. 153 (2015) 165–184, http://dx.doi.org/10.1016/j.compstruc.2015.03.003.

[24] R.L. Pedro, J. Demarche, L.F.F. Miguel, R.H. Lopez, An efficient approach for the optimization of simply supported steel-concrete composite I-girder bridges, Adv. Eng. Softw. 112 (2017) 31–45, http://dx.doi.org/10.1016/j.advengsoft.2017.06.009.

[25] D. Khamari, R.K. Sahu, S. Panda, Application of search group algorithm for automatic generation control of interconnected power system, in: H.S. Behera, J. Nayak, B. Naik, A. Abraham (Eds.), Computational Intelligence in Data Mining, Springer, Singapore, 2019, pp. 557–568, http://dx.doi.org/10.1007/978-981-10-8055-5_50.

[26] S.F.H. Noorbin, A. Alfi, Adaptive parameter control of search group algorithm using fuzzy logic applied to networked control systems, Soft Comput. 22 (2018) 7939–7960, http://dx.doi.org/10.1007/s00500-017-2742-0.

[27] F. Carraro, R.H. Lopez, L.F.F. Miguel, Optimum design of planar steel frames using the search group algorithm, J. Braz. Soc. Mech. Sci. Eng. 39 (2017) 1405–1418, http://dx.doi.org/10.1007/s40430-016-0628-1.

[28] G. Acampora, D. Caruso, A. Vaccaro, A. Vitiello, A search group algorithm for optimal voltage regulation in power systems, in: 2016 IEEE Congress on Evolutionary Computation (CEC), 2016, pp. 3662–3669, http://dx.doi.org/10.1109/CEC.2016.7744253.

[29] T.H.B. Huy, P. Nallagowden, R. Kannan, V.N. Dieu, Energetic optimization of solar water heating system with flat plate collector using search group algorithm, J. Adv. Res. Fluid Mech. Therm. Sci. 61 (2019) 306–322.

[30] B.-H. Truong, P. Nallagownden, K.H. Truong, R. Kannan, D.N. Vo, N. Ho, Multi-objective search group algorithm for thermo-economic optimization of flat-plate solar collector, Neural Comput. Appl. 33 (2021) 12661–12687, http://dx.doi.org/10.1007/s00521-021-05915-w.

[31] D.E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, first ed., Addison-Wesley Longman Publishing Co., Inc., USA, 1989.

[32] E. Zitzler, K. Deb, L. Thiele, Comparison of multiobjective evolutionary algorithms: Empirical results, Evol. Comput. 8 (2000) 173–195, http://dx.doi.org/10.1162/106365600568202.

[33] T. Thanh Binh, U. Korn, Multicriteria control system design using an intelligent evolution strategy, IFAC Proc. Vol. 30 (1997) 765–770, http://dx.doi.org/10.1016/S1474-6670(17)43457-4.

[34] C.M. Fonseca, P.J. Fleming, An overview of evolutionary algorithms in multiobjective optimization, Evol. Comput. 3 (1995) 1–16, http://dx.doi.org/10.1162/evco.1995.3.1.1.

[35] C.M. Fonseca, P.J. Fleming, Multiobjective genetic algorithms made easy: selection sharing and mating restriction, in: First International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications, 1995, pp. 45–52, http://dx.doi.org/10.1049/cp:19951023.

[36] F. Kursawe, A variant of evolution strategies for vector optimization, in: H.-P. Schwefel, R. Männer (Eds.), Parallel Problem Solving from Nature, Springer, Berlin, Heidelberg, 1991, pp. 193–197, http://dx.doi.org/10.1007/BFb0029752.

[37] M. Laumanns, G. Rudolph, H.-P. Schwefel, A spatial predator–prey approach to multi-objective optimization: A preliminary study, in: A.E. Eiben, T. Bäck, M. Schoenauer, H.-P. Schwefel (Eds.), Parallel Problem Solving from Nature — PPSN V, Springer, Berlin, Heidelberg, 1998, pp. 241–249, http://dx.doi.org/10.1007/BFb0056867.

[38] T. Murata, H. Ishibuchi, MOGA: multi-objective genetic algorithms, in: Proceedings of 1995 IEEE International Conference on Evolutionary Computation, 1995, p. 289, http://dx.doi.org/10.1109/ICEC.1995.489161.

[39] C. Poloni, A. Giurgevich, L. Onesti, V. Pediroda, Hybridization of a multi-objective genetic algorithm, a neural network and a classical optimizer for a complex design problem in fluid dynamics, Comput. Methods Appl. Mech. Engrg. 186 (2000) 403–420, http://dx.doi.org/10.1016/S0045-7825(99)00394-1.
[40] J.D. Schaffer, Some Experiments in Machine Learning using Vector Evaluated Genetic Algorithms (Artificial Intelligence, Optimization, Adaptation, Pattern Recognition) (phd), Vanderbilt University, 1984.
[41] R. Vlennet, C. Fonteix, I. Marc, Multicriteria optimization using a genetic algorithm for determining a Pareto set, Internat. J. Systems Sci. 27 (1996) 255–260, http://dx.doi.org/10.1080/00207729608929211.
[42] N. Srinivas, K. Deb, Muiltiobjective optimization using nondominated sorting in genetic algorithms, Evol. Comput. 2 (1994) 221–248, http://dx.doi.org/10.1162/evco.1994.2.3.221.
[43] A. Belegundu, E. Constans, R. Salagame, D. Murthy, Multi-objective optimization of laminated ceramic composites using genetic algorithms, in: 5th Symposium on Multidisciplinary Analysis and Optimization, American Institute of Aeronautics and Astronautics, 1994, http://dx.doi.org/10.2514/6.1994-4363.
[44] T.T. Binh, U. Korn, MOBES: A multiobjective evolution strategy for constrained optimization problems, in: Proceedings of the Third International Conference on Genetic Algorithms (Mendel97, 1997, pp. 176–182.
[45] H. Kita, Y. Yabumoto, N. Mori, Y. Nishikawa, Multi-objective optimization by means of the thermodynamical genetic algorithm, in: H.-M. Voigt, W. Ebeling, I. Rechenberg, H.-P. Schwefel (Eds.), Parallel Problem Solving from Nature — PPSN IV, Springer, Berlin, Heidelberg, 1996, pp. 504–512, http://dx.doi.org/10.1007/3-540-61723-X_1014.
[46] M. Tanaka, H. Watanabe, Y. Furukawa, T. Tanino, GA-based decision support system for multicriteria optimization, in: 1995 IEEE International Conference on Systems, Man and Cybernetics. Intelligent Systems for the 21st Century, Vol. 2, 1995, pp. 1556–1561, http://dx.doi.org/10.1109/ICSMC.1995.537993.
[47] V. Martínez-Cagigal, Non sorting genetic algorithm II (NSGA-II), 2020, https://www.mathworks.com/matlabcentral/fileexchange/65494-non-sorting-genetic-algorithm-ii-nsga-ii (accessed August 9, 2020).
[48] V. Martínez-Cagigal, Multi-objective particle swarm optimization (MOPSO), 2020, https://www.mathworks.com/matlabcentral/fileexchange/62074-multi-objective-particle-swarm-optimization-mopso (accessed August 9, 2020).
[49] S. Mirjalili, Multi-objective multi-verse optimization (MOMVO) algorithm, 2020, https://www.mathworks.com/matlabcentral/fileexchange/63796-multi-objective-multi-verse-optimization-momvo-algorithm (accessed August 9, 2020).
[50] S. Mirjalili, MODA: Multi-objective dragonfly algorithm, 2020, https://www.mathworks.com/matlabcentral/fileexchange/51033-moda-multi-objective-dragonfly-algorithm (accessed August 13, 2020).
[51] S. Mirjalili, MOGOA: Multi-objective grasshopper optimization algorithm, 2020, https://www.mathworks.com/matlabcentral/fileexchange/63786-mogoa-multi-objective-grasshopper-optimization-algorithm (accessed August 13, 2020).
[52] D.A.V. Veldhuizen, G.B. Lamont, Evolutionary computation and convergence to a Pareto front, in: Late Breaking Papers at the Genetic Programming 1998 Conference, Stanford University Bookstore, Stanford, CA, USA, 1998, pp. 221–228.
[53] M.R. Sierra, C.A. Coello Coello, Improving PSO-based multi-objective optimization using crowding, mutation and $\in$-dominance, in: C.A. Coello Coello, A. Hernández Aguirre, E. Zitzler (Eds.), Evolutionary Multi-Criterion Optimization, Springer, Berlin, Heidelberg, 2005, pp. 505–519, http://dx.doi.org/10.1007/978-3-540-31880-4_35.
[54] J.R. Schott, Fault Tolerant Design using Single and Multicriteria Genetic Algorithm Optimization (Thesis), Massachusetts Institute of Technology, 1995, https://dspace.mit.edu/handle/1721.1/11582 (accessed August 9, 2020).
[55] Y.-N. Wang, L.-H. Wu, X.-F. Yuan, Multi-objective self-adaptive differential evolution with elitist archive and crowding entropy-based diversity measure, Soft Comput. 14 (2009) 193, http://dx.doi.org/10.1007/s00500-008-0394-9.
[56] M. Karakoyun, A. Ozkis, H. Kodaz, A new algorithm based on gray wolf optimizer and shuffled frog leaping algorithm to solve the multi-objective optimization problems, Appl. Soft Comput. 96 (2020) 106560, http://dx.doi.org/10.1016/j.asoc.2020.106560.
[57] A. Özkış, A. Babalık, A novel metaheuristic for multi-objective optimization problems: The multi-objective vortex search algorithm, Inform. Sci. 402 (2017) 124–148, http://dx.doi.org/10.1016/j.ins.2017.03.026.
[58] A. Babalik, A. Ozkis, S.A. Uymaz, M.S. Kiran, A multi-objective artificial algae algorithm, Appl. Soft Comput. 68 (2018) 377–395, http://dx.doi.org/10.1016/j.asoc.2018.04.009.
[59] S. Huband, P. Hingston, L. Barone, L. While, A review of multiobjective test problems and a scalable test problem toolkit, IEEE Trans. Evol. Comput. 10 (2006) 477–506, http://dx.doi.org/10.1109/TEVC.2005.861417.
[60] K. Deb, L. Thiele, M. Laumanns, E. Zitzler, Scalable test problems for evolutionary multiobjective optimization, in: A. Abraham, L. Jain, R. Goldberg (Eds.), Evolutionary Multiobjective Optimization: Theoretical Advances and Applications, Springer, London, 2005, pp. 105–145, http://dx.doi.org/10.1007/1-84628-137-7_6.
[61] H. Li, Q. Zhang, Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II, IEEE Trans. Evol. Comput. 13 (2009) 284–302, http://dx.doi.org/10.1109/TEVC.2008.925798.
[62] T. Okabe, Y. Jin, M. Olhofer, B. Sendhoff, On test functions for evolutionary multi-objective optimization, in: X. Yao, E.K. Burke, J.A. Lozano, J. Smith, J.J. Merelo-Guervós, J.A. Bullinaria, J.E. Rowe, P. Tiňo, A. Kabán, H.-P. Schwefel (Eds.), Parallel Problem Solving from Nature - PPSN VIII, Springer, Berlin, Heidelberg, 2004, pp. 792–802, http://dx.doi.org/10.1007/978-3-540-30217-9_80.
[63] P. Jangir, N. Jangir, A new non-dominated sorting grey wolf optimizer (NS-GWO) algorithm: Development and application to solve engineering designs and economic constrained emission dispatch problem with integration of wind power, Eng. Appl. Artif. Intell. 72 (2018) 449–467, http://dx.doi.org/10.1016/j.engappai.2018.04.018.
[64] C.A. Coello Coello, G.T. Pulido, Multiobjective structural optimization using a microgenetic algorithm, Struct. Multidiscip. Optim. 30 (2005) 388–403, http://dx.doi.org/10.1007/s00158-005-0527-z.
[65] F.Y. Cheng, X.S. Li, Generalized center method for multiobjective engineering optimization, Eng. Optim. 31 (1999) 641–661, http://dx.doi.org/10.1080/03052159908941390.
[66] S. Gunawan, S. Azarm, J. Wu, A. Boyars, Quality-assisted multi-objective multidisciplinary genetic algorithms, AIAA J. 41 (2003) 1752–1762, http://dx.doi.org/10.2514/2.7293.
[67] T. Ray, K.M. Liew, A swarm metaphor for multiobjective design optimization, Eng. Optim. 34 (2002) 141–153, http://dx.doi.org/10.1080/03052150210915.
[68] K. Deb, A. Pratap, S. Moitra, Mechanical component design for multiple ojectives using elitist non-dominated sorting GA, in: M. Schoenauer, K. Deb, G. Rudolph, X. Yao, E. Lutton, J.J. Merelo, H.-P. Schwefel (Eds.), Parallel Problem Solving from Nature PPSN VI, Springer, Berlin, Heidelberg, 2000, pp. 859–868, http://dx.doi.org/10.1007/3-540-45356-3_84.