2023

# A Multi–Objective Gaining–Sharing Knowledge-Based Optimization Algorithm for Solving Engineering Problems

Nour Elhouda Chalabi
*University Mohamed Boudiaf of Msila, Algeria*

Abdelouahab Attia
*Mohamed El Bachir El Ibrahimi University of Bordj Bou Arreridj, Algeria*

Khalid Abdulaziz Alnowibet
*King Saud University, Saudi Arabia*

*See next page for additional authors*



## Recommended Citation


Funder: The research is funded by the Researchers Supporting Program at King Saud University, (RSP2023R305).

## Authors
Nour Elhouda Chalabi, Abdelouahab Attia, Khalid Abdulaziz Alnowibet, Hossam Zawbaa, Hatem Masri, and Ali Wagdy Mohamed

*mathematics*

*Article*

# A Multi–Objective Gaining–Sharing Knowledge-Based Optimization Algorithm for Solving Engineering Problems

**Nour Elhouda Chalabi [1], Abdelouahab Attia [2,3], Khalid Abdulaziz Alnowibet [4], Hossam M. Zawbaa [5], Hatem Masri [6] and Ali Wagdy Mohamed [7,8,*]**

1 Computer Science Department, University Mohamed Boudiaf of Msila, Msila 28000, Algeria
2 LMSE Laboratory, Mohamed El Bachir El Ibrahimi University of Bordj Bou Arreridj, Bordj Bou Arreridj 34000, Algeria
3 Computer Science Department, University Mohamed El Bachir El Ibrahimi of Bordj Bou Arreridj, Bordj Bou Arreridj 34000, Algeria
4 Statistics and Operations Research Department, College of Science, King Saud University, P.O. Box 2455, Riyadh 11451, Saudi Arabia; knowibet@ksu.edu.sa
5 CeADAR Ireland's Center for Applied AI, Technological University Dublin, D7 EWV4 Dublin, Ireland
6 Applied Science University, Sakhir 32038, Bahrain; hatem.masri@asu.edu.bh
7 Operations Research Department, Faculty of Graduate Studies for Statistical Research, Cairo University, Giza 12613, Egypt
8 Applied Science Research Center, Applied Science Private University, Amman 11937, Jordan
* Correspondence: aliwagdy@staff.cu.edu.org

**Abstract:** Metaheuristics in recent years has proven its effectiveness; however, robust algorithms that can solve real-world problems are always needed. In this paper, we suggest the first extended version of the recently introduced gaining–sharing knowledge optimization (GSK) algorithm, named multiobjective gaining–sharing knowledge optimization (MOGSK), to deal with multiobjective optimization problems (MOPs). MOGSK employs an external archive population to store the nondominated solutions generated thus far, with the aim of guiding the solutions during the exploration process. Furthermore, fast nondominated sorting with crowding distance was incorporated to sustain the diversity of the solutions and ensure the convergence towards the Pareto optimal set, while the $\epsilon$-dominance relation was used to update the archive population solutions. $\epsilon$-dominance helps provide a good boost to diversity, coverage, and convergence overall. The validation of the proposed MOGSK was conducted using five biobjective (ZDT) and seven three-objective test functions (DTLZ) problems, along with the recently introduced CEC 2021, with fifty-five test problems in total, including power electronics, process design and synthesis, mechanical design, chemical engineering, and power system optimization. The proposed MOGSK was compared with seven existing optimization algorithms, including MOEAD, eMOEA, MOPSO, NSGAII, SPEA2, KnEA, and GrEA. The experimental findings show the good behavior of our proposed MOGSK against the comparative algorithms in particular real-world optimization problems.

**Keywords:** multiobjective optimization; gaining–sharing knowledge optimization; crowding distance; Pareto optimal set; $\epsilon$ dominance relation

**MSC:** 68T01; 68T05; 68T07; 68T09; 68T20; 68T30

check for updates

**Citation:** Chalabi, N.E.; Attia, A.; Alnowibet, K.A.; Zawbaa, H.M.; Masri, H.; Mohamed, A.W. A Multi–Objective Gaining–Sharing Knowledge-Based Optimization Algorithm for Solving Engineering Problems. *Mathematics* **2023**, *11*, 3092. https://doi.org/10.3390/math11143092

Academic Editors: Mohammad Shokouhifar, Frank Werner, Laith Abualigah, Seyedali Mirjalili and Andreas C. Georgiou

Received: 17 April 2023
Revised: 7 July 2023
Accepted: 10 July 2023
Published: 13 July 2023



## 1. Introduction

In recent years, multiobjective optimization problems (MOPs) have been given significant attention by researchers in solving real-world optimization problems [1], where they deal with multiple contradicting objectives. Due to their robustness; MOP methods are widely used in various fields [2]. Two major approaches, a priori and a posteriori, have been planned to solve MOP problems [3,4]. When using an a priori approach, the MOP is turned into a single-objective problem (SOP) using a weight vector that describes the implication of each objective. Such approaches produce one Pareto solution set [5]. In general, a priori approaches are not feasible, due to the need to have the decision makers provide a weight for each objective. Consequently, the a posteriori approach selects the effectively distributed set of solutions, also known as nondominated solutions. Then, the decision makers can select a fitting solution. Techniques and algorithms that can deal with MOP have been studied used heavily over the years. Those algorithms are generally metaheuristic algorithms; they are based on two concepts—exploration and exploitation—and most of the time, a metaheuristic algorithm looks for a balance between those two. Now, metaheuristic algorithms are categorized based on the origin of their inspiration, such as evolution, swarm intelligence, physics-based origins, and human-related origins.

Evolutionary-based techniques [6] are well known and widely used; the most representative algorithm of this category is the genetic algorithm (GA) [7]. GA is founded on biological evolution; it has also an extended version that solves MOP, named the nondominating sorting genetic algorithm (NSGA) [8], where the aspect of the nondominated sort was first introduced. The nondominating sorting genetic algorithm (NSGAII) [9] is another version of NSGA; in this version, the fast nondominated sorting approach (FNS) and crowding distance (CD) are presented. In the Pareto archive evolutionary strategy PAES [10], the implication of external archives is introduced; numerous multiobjective-based evolutionary algorithms have been designed and studied, including the multiobjective evolutionary algorithm based on decomposition, also recognized as MOEA/D [11], eMOEA [9,12], SPEA2 [13], KnEA [14], GrEA [15], and many others. The remarkable successes motivated researchers to investigate and plan other multiobjective evolutionary algorithms, such as the harmony search algorithm [16], water cycle algorithm [17], ant lion optimizer [2], and discrete cooperative swarm intelligence algorithm [18]. Kumawat et al. [19] proposed a multiobjective whale optimization algorithm (MOWOA). Moreover, Mohamed Abdel-Basset and Mirjalili [20] introduced an extension of the whale optimization algorithm for solving MOP. Also, Mohamed Abdel-Basset et al. [21] enhanced the equilibrium algorithm for resolving multiobjective problems based on the archive approach; to maintain diversity among the nondominated solutions, Pareto optimal solutions and the crowding distance metric were also used. Wang et al. [22] introduced a multiobjective evolutionary method incorporated by a uniformly evolving system to locate nondominated solutions that are homogeneously distributed on the true Pareto optimal curve to provide flexibility of the decision makers while choosing the suitable solutions. Swarm intelligence-based metaheuristic algorithms are based generally on the homogeneous movement of an agent. It should be noted that there has been massive work in this category. Over the last three decades, these categories in particular have assumed the lead and continue to do so, where a huge number of those algorithms have been introduced and applied [23]. The most known one is particle swarm optimization (PSO) [24]; several versions of this technique, in particular multiobjective particle swarm optimization (MOPSO) [25], have been employed to solve MOP, where the concept of dominance is used. Another well-known algorithm is ant colony optimization (ACO) [26] as well as its extended version for handling MOP, named multiobjective ant colony optimization (MOACO) [27]. There is also the ant lion optimizer (ALO) [28] with MOALO as a multiobjective extended version [2]. Another recently introduced algorithm, the whale optimization algorithm (WOA) [29], is an algorithm that imitates humpback whales' social behavior. A guided marine predator optimization for multiobjective issues, known as GMOMPA, was introduced in a recent article [30]; it is based on mono-objective marine predator optimization (MPA), and this work incorporates an external archive to keep the best solutions found so far; in addition, it uses the epsilon dominance relation to update the archive solutions. Fast nondominated sorting (FNS) and crowding distance (CD) were used to keep a balance between exploration and exploitation.

Human-related algorithms are numbered [31], since we as humans have a limited understating of the human brain. A human is considered a highly intelligent being since he or she holds several critical abilities, such as understanding, reasoning, identifying, communicating, solving a problem, and many more. Therefore, inspiration from such a creature to develop an algorithm sounds reasonable, and might help in solving critical real-life issues. One of the oldest known human-related algorithms is the cultural algorithm [32], inspired from cultural evolution and the mechanism of inheritance of that culture; another known and widely used algorithm is the harmony search algorithm [33], where the improvisation of music players is the base of its inspiration. The league championship-inspired algorithm [34] is another one that is based on the dynamics of the competition of sports teams in a sports league, such as the league schedule, pair play, losses and wins, and team formation. The teaching–learning-based optimization [35] algorithm is based on the effect of the influence of a teacher on learners. Proposed by Yuhui, brain storm optimization [36] is an algorithm based on the brainstorming process. Cohort intelligence [37] is an optimization algorithm inspired by the social and natural urge for people to learn from one another. The soccer league competition algorithm [38] was developed based on competitions between clubs and players in soccer leagues. The ideology algorithm [39] draws inspiration from the self-centered and competitive behavior of political party members who are driven to raise their standing. The competitiveness between volleyball teams is what inspired the volleyball premier league algorithm [40] algorithm. Te life-choice-based optimizer [41] is a recent optimizer based on how people often make decisions to achieve their goals and pick up knowledge from others. The future search algorithm [42] simulates the person's life when a person looks for a better life than the one he or she has. The forensic-based investigation optimizer [43] is inspired by police officers' methods for locating, pursuing, and investigating suspects. The dynastic optimization algorithm [44] is based on the social behavior of human dynasties. Finally, there is anticoronavirus optimization [45]. All these human-related algorithms have been used to solve different optimization problems.

Recently, a new optimization algorithm was introduced, named gaining–sharing knowledge optimization (GSK) [31]. GSK is an optimization algorithm with a nature-inspired background that is based on the acquiring process of information and knowledge, as well as sharing it during a human's life span; this algorithm distinguishes two phases: the first phase is the junior (child) gaining–sharing knowledge phase, while the second phase is the senior (adult) gaining–sharing knowledge phase, where it follows how a junior shares and gains knowledge and the change of that process when moving to adulthood. GSK as an optimization algorithm showed great potential where several binary GSK versions had been proposed and applied, such as the gaining–sharing knowledge-based S- and V-shaped feature selection algorithm [46] and the binary GSK for the location of the fault in distribution networks via mutation [47]; in this work, a new mutation-based enhanced binary gaining–sharing algorithm (IBGSK) is introduced and applied to the converted binary fault section location (FSL). In another work, Agrawal et al. [48] proposed a binary GSK to solve the known knapsack problems. In addition to the various binary GSKs and their application, numerous works have applied GSK, such as Li et al. [49]'s recent work, where GSK is applied to optimize the parameters in the proposed structure of fault section diagnosis (FSD) based on the Takagi–Sugeno fuzzy neural networks; this structure is designed to deal effectively with the issues related to uncertainties of protective relays and circuit breakers existing within power system faults. In a different work proposed by Ortega-Sánchez et al. [50] that tackles the issue of the identification of apple diseases via digital images, here, when segmenting apple images with the disease, GSK is used to minimize cross-entropy thresholding. Hassan et al. [51] propose and use a binary version of GSK to address scheduling issues of the technical counseling process for using the electricity generated by solar energy power; in this work, a new application problem is introduced, named the traveling counseling problem (TCP). Xiong et al. [52] engaged GSK in feature extraction; this work handles the solar photovoltaic (PV) system, and it is crucial to precisely create an equivalent model of the PV cell and derive the relevant unidentified model parameters for it to function effectively. In this case, GSK is employed to achieve this. Lastly, it is safe to say that GSK is a powerful tool for optimization, not to mention that GSK has another version with adaptive parameters [53,54]; this version tackles the issue of the appropriate parameters that should be selected and set for GSK to give the best results. Moreover, it is motivated by the ability of GSK and the no free lunch (NFL) theorem [55], which states that there is no optimization algorithm capable of solving all sorts of optimization problems with this fact applied to both single and multiobjective optimization. In addition, the fact is that human-related algorithms that can solve mono-objective optimization problems are limited [56], let alone the ones that can solve multiobjective problems. Furthermore, real-world optimization problems are increasing every day, and a tool that can help solve them is much needed. Therefore, this study presents the first-ever extended version of the recently introduced gaining–sharing knowledge optimization (GSK) algorithm to solve MOPs named MOGSK. To pass from a single-objective optimization to a multiobjective optimization, several strategies were adapted, which can be summarized as follows:

- We proposed an MOGSK to solve multiobjective optimization problems.
- The external archive was incorporated to maintain the nondominant solutions discovered so far and guide the particles toward the optimal Pareto set later in the exploration process.
- The $\epsilon$-dominance relation was used to update the archive solutions. Additionally, it promoted exploitation and exploration while helping to increase diversity.
- In aim to preserve a good exploitation, diversity, and an effective solution distribution, the crowding distance and fast nondominated sorting were used.
- ZDT, DTLZ series test functions, and CEC 2021 RWMOPs (real-world constrained multiobjective optimization problems) were the test benchmarks to be utilized to validate the proposed MOGSK algorithm.
- In order to further evaluate the proposed MOGSK, a comparison was conducted against different algorithms, such as MOEAD, eMOEA, MOPSO, NSGAII, SPEA2, KnEA, and GrEA.

The rest of the paper is arranged according to Figure 1 as follows: Section 2 explains the basics of multiobjective optimization problems and the gaining–sharing knowledge optimization algorithm (GSK). Section 3 introduces the proposed MOGSK algorithm. The experimental results, the comparisons, and the discussion are presented in Section 4. Finally, Section 5 presents the conclusions and suggestions for future work directions.

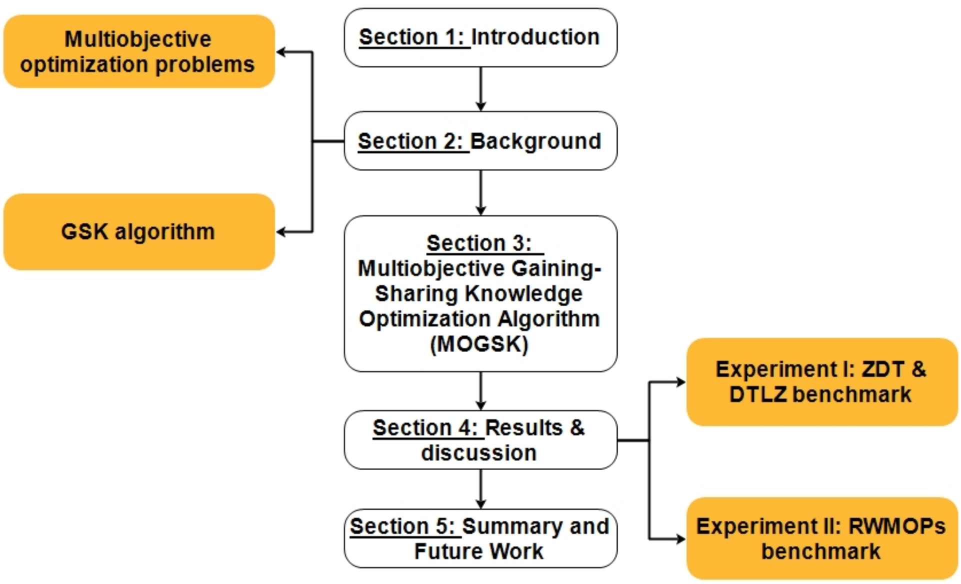


**Figure 1.** Paper organization.

## 2. Background

This section presents some useful information regarding multiobjective optimization problems (MOPs), including the definition of some concepts such as Pareto dominance. In addition, this section describes the standard single-objective-based gaining–sharing knowledge optimization algorithm (GSK).

### 2.1. Multiobjective Optimization Problems

Multiobjective optimization is a process where conflicting objective functions are optimized simultaneously. Depending on the problem treated, multiobjective optimization can be either minimizing or maximizing. In case of minimization, an MOP is formulated [57], as follows:

$$Mminimize : f_m(x),\ (m = 1, 2, ..., M), \tag{1}$$

$$subject\ to\ g_j(x) = 0,\ j = (1, 2, 3, ..., J), \tag{2}$$

$$h_k(x) \le 0,\ k = (1, 2, 3, ..., K) \tag{3}$$

$$l_i \le x_i \le u_i,\ i = (1, 2, 3, ..., n) \tag{4}$$

where $x$ in $f_m(x)$ represents the solution for $n$ decision variables $x = (x_1, x_2, ..., x_n)$, while satisfying the $J$ of $g_j(x)$ inequality and $K$ of $h_k(x)$ equality constraints. $M$ refers to the number of objective functions. The lower and upper boundaries of the decision variables are represented by $l_i$ and $u_i$, respectively. In an MOP, comparing the generated solutions with relational arithmetic operators is challenging. Therefore, the Pareto optimal dominance concept offered an easy approach to compare solutions, where there is a set of solutions instead of one single solution.

### 2.2. Pareto Dominance

The core concept behind the Pareto dominance relation comprised :

**Definition 1** (Pareto dominance)**.** *A solution u is said to dominate another solution v (as $u \succ v$) iff:*

$$\begin{aligned} &\forall i \in \{1, 2, 3, ..., M\} : f_i(u) \le f_i(v)\ and \\ &\exists j \in \{1, 2, 3, ..., M\} : f_j(u) < f_j(v) \end{aligned} \tag{5}$$

*$M$ represents the number of the objective functions. A solution u is generally said to weakly dominate another solution v (noted as: $u \succcurlyeq v$) iff:*

$$\forall i \in \{1, 2, 3, ..., M\} : f_i(u) \le f_i(v) \tag{6}$$

**Definition 2** (A nondominated set)**.** *The solutions that are not dominated by any other solution are said to be the nondominated set. Let A be a set of solutions; the nondominated solution is included in set $A' \subseteq A$ is nondominated by any other solution in set A.*

**Definition 3** (Pareto optimal set)**.** *This is the set of all of the nondominated solutions in the research space. The Pareto front refers to the Pareto optimal set illustration in the objective space.*

### 2.3. Gaining–Sharing Knowledge Optimization Algorithm (GSK)

New optimization approaches are developed and introduced each year to solve real-world problems. Therefore, a new optimization algorithm was proposed recently, titled the gaining–sharing Knowledge optimization algorithm (GSK) [31]. GSK is a human-based algorithm that simulates knowledge gaining and sharing in the course of a human lifetime. GSK's main mechanisms depend on two important stages: first, junior gaining–sharing knowledge; and second, senior gaining–sharing knowledge.

- Junior gaining and sharing knowledge: in this stage, the individual tries to gain information from their small circle of people, such as family, relatives, and neighbors, since they cannot interact on a large scale such as social media. Even with a lack of experience, juniors still have the will to share their knowledge with the people they know or not; in addition, they do not have the ability yet to categorize people as bad or good, so they share due to curiosity and exploration.
- Senior gaining and sharing knowledge: in this stage, an individual is more experienced and has a wider circle of people to interact with, such as social networks, friends, and colleagues. Therefore, they gain their knowledge from their entourage. In addition, in this phase, they have an advanced ability to categorize people into classes such as best, better, and worst. Therefore, they share knowledge with the most suitable individuals and improve their skills.

The mathematical formulation of the above-mentioned GSK process follows several steps:

- Initialization of the necessary factors, such as N—the population size, which corresponds to the number of people. Initialization of the starting population is random while respecting the boundary constraints, where $x_i$ $(i = 1, 2, 3, ..., N)$ represent the individuals; each $x_i$ corresponds to $x_{ih}$ with $x_{ih}$ $(x_{i1}, x_{i2}, x_{i3}, ..., x_{id})$, deferring to the possible number of fields of disciplines. To rephrase, it can be seen as branch of knowledge allocated to an individual. The fitness evaluation of the population noted as $f_j(j = 1, 2, 3, ..., N)$ is also conducted.
- Now, the dimension between junior and senior is decided through the following nonlinear equation:

$$d(junior) = Problemsize * \left(1 - \frac{G}{Gen}\right)^k \tag{7}$$

$$d(senior) = Problemsize - d(junior) \tag{8}$$

  $d(junior)$ and $d(senior)$ are the dimensions of junior and senior phases. $k$ refers to the knowledge rate $(k > 0)$. $G$ is the number of generations, while $Gen$ is the maximum number of generations.
- In this step, the junior gaining–sharing knowledge stage begins. In this stage each individual tries to gain knowledge from their small network; at the same time, they try to share their knowledge. The people that they interact with can be from their network or not, since in this phase, they are driven by curiosity.
- Now the update of the individuals in the current stage is conducted following the junior scheme:
  - Based on the values of the objective function, the individuals are sorted in ascendant order.
  - For each individual, the closest best and worst are selected to gain knowledge. In addition, a random individual is selected to share knowledge.

  This step process is shown in Algorithm 1. $K_f$ is the knowledge factor, where $K_f > 0$; this parameter controls the amount of knowledge (gained/shared) that is going to be added to the actual individual. $k_r$ is the knowledge ratio, where $k_r \in [0, 1]$; this parameter controls the amount of knowledge (gained/shared) that is going to be transferred to another individual.

**Algorithm 1** Phase 1: junior gaining and sharing knowledge [31].

**for** $i = 1$: $(N)$ **do**
  **for** $h = 1$: $(d)$ **do**
    **if** $rand \leq k_r(knowledge\ ratio)$ **then**
      **if** $f(x_i) > f(x_{ran})(x_{ran}$ *is a randomly selected individual)* **then**
        $x_{ih}^{new} = x_h * K_f[(x_{(i-1)} - x_{(i+1)}) + (x_{ran} - x_i)]$
      **else**
        $x_{ih}^{new} = x_h * K_f[(x_{(i-1)} - x_{(i+1)}) + (x_i - x_{ran})]$
      **end**
    **else**
      $x_{ih}^{new} = x_{ih}^{old}$
    **end**
  **end**
**end**

- This step is the senior gaining–sharing knowledge phase. This stage takes into account a person's capacity for classification (such as good and bad). And the scheme in this stage is as follows:
  - First, the values of the objective function are used to sort the individuals in ascendant order.
  - Then, those individuals are split into three groups: worst, middle, and best, for example: $Best = 100p\%(x_{best}), Middle = N - (2 * 100p\%)(x_{middle})$, and $Worst = 100p\%(x_{worst})$.
  - Now, two vectors are chosen from the best and worst for gaining ($100p\%$), while a third vector from the middle is chosen for sharing $N - (2 * 100p\%)$. $p$ here indicates the percentage of best and worst individuals, where $p \in [0, 1]$. This step process is shown in Algorithm 2.

**Algorithm 2** Phase 2: senior gaining and sharing knowledge [31].

**for** $i = 1$: $(N)$ **do**
  **for** $h = 1$: $(d)$ **do**
    **if** $rand \leq k_r$ **then**
      **if** $f(x_i) > f(x_{ran})$ **then**
        $x_{ih}^{new} = x_h * K_f[(x_{best} - x_{worst}) + (x_{middle} - x_i)]$
      **else**
        $x_{ih}^{new} = x_h * K_f[(x_{best} - x_{worst}) + (x_i - x_{middle})]$
      **end**
    **else**
      $x_{ih}^{new} = x_{ih}^{old}$
    **end**
  **end**
**end**

## 3. Multiobjective Gaining–Sharing Knowledge Optimization Algorithm (MOGSK)

This section describes the proposed MOGSK and its mathematical formulation. In order to pass from single-objective optimization to multiobjective optimization, several components are introduced, which include the implementation of a separate repository to store the nondominant solutions uncovered thus far. Those nondominated solutions are obtained using the Pareto dominance relation in addition to fast nondominated sorting and the crowding distance, which help with diversity and improve exploitation and exploration, while the $\epsilon$-dominance relation is incorporated to update the archive (repository) solutions. In order to update the archive, the solutions of the current population and previous archive solutions are used in the process. At last, the archive is used to guide the population toward the Pareto optimal. To sum up, the techniques used in the proposed MOGSK are:

- Fast nondominated sorting (FNS), in order to obtain the nondominated solutions.
- Crowding distance, to insure the distribution and convergence of the solutions as well as to improve the diversity.
- The archive, to preserve the best solutions so far and to act as a guide to the individual towards the Pareto optimal set.
- The epsilon dominance relation, which is employed each iteration to update the archive's solutions.

The introduced techniques have tremendous advantages that help MOGSK to be a good optimization algorithm. First, the archive acts as a guide for the solution toward the Pareto optimal, while preserving diversity and helping maintain the balance between exploration and exploitation. Also, The flexibility and diversity provided by the $\epsilon$-dominance relation help to include a variety of solutions, in addition to crowding distance and fast nondominated sorting (FNS), which boost coverage and accelerate convergence toward the Pareto optimal. And lastly, the new population, which is a combination of the current population and previous archive, contributes as much to the exploitation and exploration.

After the initialization of the necessary parameters, which are shown in Table 1, comes the initialization of the first population, followed by the assessment of the fitness value for each individual. The elitist fast nondominated sorting (FNS) [9] is used on the first population. FNS can obtain the nondominated solution and sort the solution according to different fronts. Following that, there is the application of the crowding distance (CD) [9]. Once finished, the archive is initialized, with the nondominated solution obtained. The MOGSK algorithm will compute, for a predetermined number of iterations, a series of key steps, as follows: updating the population (gaining or sharing) and updating the archive.

**Table 1.** MOGSK parameters.

| **Algorithm Parameters** | |
|---|---|
| $N$ | Population size (number of individuals) = 100 |
| $k$ | Knowledge rate $(k > 0)$ = 10 |
| $k_r$ | Knowledge ratio $(k_r \in [0,1])$ = 0.1 |
| $K_f$ | Knowledge factor $(K_f > 0)$ = 0.9 |
| $Run_no$ | Number of runs = 30 independent runs |
| $Max_fe$ | Maximum number of function evaluation = 60,000 |

### *3.1. Update Population (Gaining/Sharing)*

The population's gaining/sharing must be updated at each iteration to move towards the Pareto optimal. In addition, this population plays a huge role, as it is used in the process of updating the archive in a later step. However, the update is different than the one used in the GSK algorithm, where it selects an individual to gain knowledge from and another to share with by arranging the individuals according to their objective value. In our case, in order to obtain the set of solutions that will be used in the gaining and sharing process, we first combine the current population's solutions and the previous archive solutions to preserve the diversity, followed by the application of FNS and crowding distance. These two techniques were considered to help boost exploitation and exploration, and a new set of solutions was founded ($New_{sol}$). This set is now used in the process of gaining and sharing. To summarize, in this phase, three key points are used: $New_{sol}$ solutions, fast nondominated sorting (FNS), and crowding distance.

#### 3.1.1. $New_{sol}$ Solutions

As stated previously, in order to update the population (gaining/sharing), a new set of solutions is used. This set is obtained by combining the current population's solutions $Population^{iteration}$ and the previous archive solutions $Archive^{iteration-1}$; by carrying this out, we can ensure diversity and convergence, as well as maintaining good exploration and exploitation.

$$New_{sol} = Population^{iteration}, A^{iteration-1} \tag{9}$$

#### 3.1.2. Fast Nondominated Sorting (FNS)

FNS [9] is employed on the $New_sol$; now, this technique is employed since a simple comparison (using the comparison operator) to find the best solution among the obtained ones is not possible, due to the contracting objectives. FNS picks each solution from the population and evaluates its dominance over the remaining solutions. This procedure generates a first front; in order to generate the next front, the first front solutions are excluded from the population, and the procedure recurs until all the solutions are ranked and sorted according to their respected front, as illustrated in Figure 2.

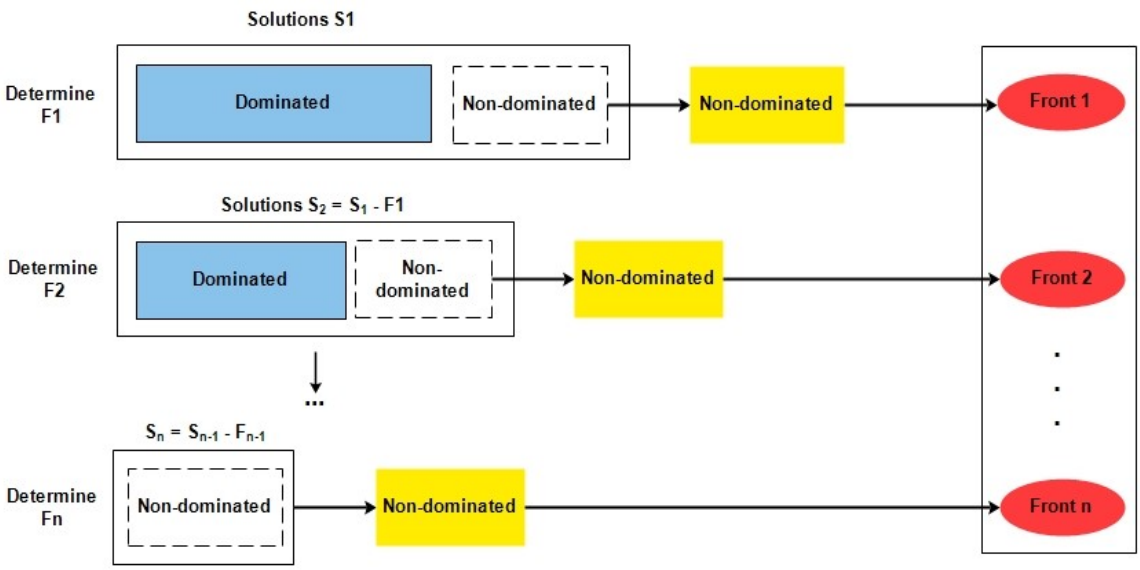


**Figure 2.** Nondominated sorting illustration.

#### 3.1.3. Crowding Distance

Once the solutions are sorted using FNS, the crowding distance (CD) [9] is employed. Mainly, CD is used to keep up the distribution and diversity of the solutions. CD estimates the density around a particular solution, which means the CD is formulated by calculating the average distance between the two nearest solutions of a cuboid of a given solution, as shown in Figure 3. The mathematical formulation of CD is noted as:

$$CD^{i}_{fj} = \frac{f_j^{(i+1)} + f_j^{(i-1)}}{f_j^{(max)} + f_j^{(min)}} \tag{10}$$

where $f_j^{(i+1)}$ and $f_j^{(i-1)}$ are the objective values of the neighborhood solutions of the solution $i$; the objective function's maximum and minimum values are $f_j^{max}$ and $f_j^{min}$, where $j$ is the objective function. The CD of all the solutions for all the objective functions is calculated, then the solutions are arranged in an ascending order following the values of the CD obtained.

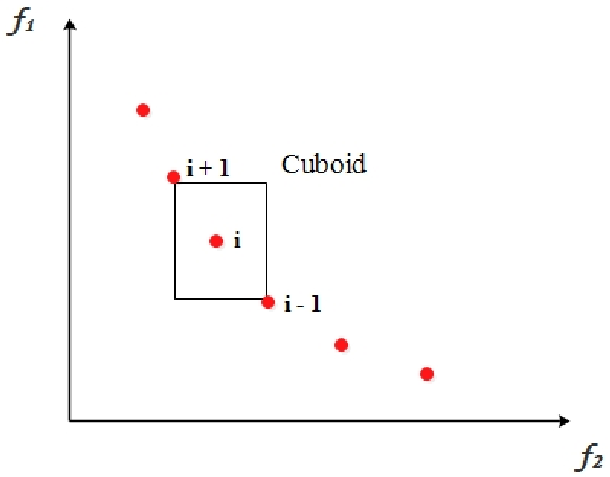


**Figure 3.** Crowding distance [9].

## 3.2. Update Archive

Once the solutions are sorted, then comes the third component of MOGSK: the archive. The archive's main roles are as follows: first, to keep the best solutions found so far; and second, to help in the update process of the new population. Here, we have two cases, as shown in the flowchart in Figure 4. The first case is when it is the first generation: the archive is initialized using the solutions of the first front found (of the first population) by FNS, since the first front has the best solutions. The second case is when it is not the first generation: here, the current solutions and the archive of the previous generation are combined ($New_sol$); once completed, the solutions of the archive are updated using this new solution combination by applying the epsilon dominance relation. The main point in the combination of those two solutions is to make sure to preserve the best solution of the previous archive and include the best solution for the new population, since in the case where only the current solution is used, there is a possibility of losing good solutions from the previous generation. In addition, the archive solution participates in orienting the solutions around the Pareto optimal. The size of the archive is managed where only the first $N$ solutions are kept.

### $\epsilon$-Dominance

The epsilon dominance relation ($\epsilon$-dominance) is a known and widely used relaxed dominance relation to improve multiobjective algorithms' efficiency. Let $\epsilon$ be a relaxation vector, where $\epsilon \in R^m$, with $m$ as the number of objective function and $\epsilon_i > 0$. For a solution $a$ to be said to be $\epsilon$-dominant, another solution $b$ is noted as $a \prec_\epsilon b$, when the $f_i(a) - \epsilon \leq f_i(b)$ condition is satisfied for all the objective functions $(f_i)$. The mechanism of this concept is basically box-level dominance in addition to regular dominance. First, the space is divided into hyperboxes (hypercubes). Each box is identified by a unique vector $B = (B_1, B_2, ..., B_M)$ assigned for each solution $x$, where $M$ represents the number of objectives. The vector $B$ can be identified as follows:

$$B_i(f) = \lfloor \frac{log(f_i)}{log(\epsilon + 1)} \rfloor \tag{11}$$

where $\lfloor . \rfloor$ indicates the absolute value, $f_i$ the objective value of the $ith$ solution, and $\epsilon$ is the permissible error. Figure 5 describes a presentation of $\epsilon$-dominance for the $x$ solution.

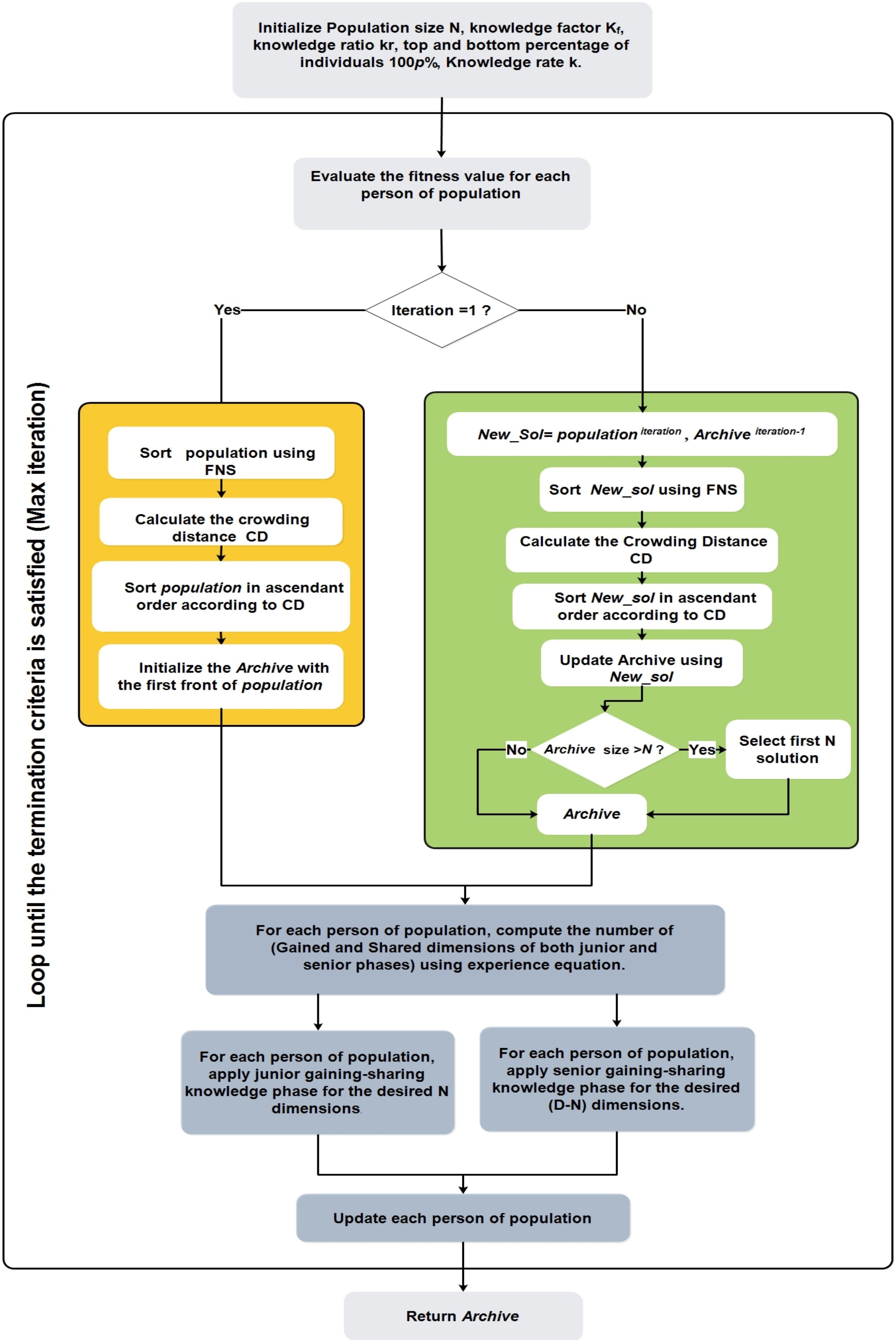


**Figure 4.** MOGSK flowchart.

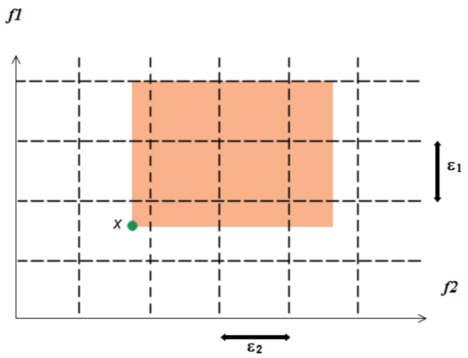


**Figure 5.** An illustration of $\epsilon$-dominance for a solution $x$, with $f_1$ and $f_2$ objective functions and $\epsilon_i$ as the tolerance level.

$\epsilon$-dominance is a practical technique that helps maintain diversity and convergence toward the Pareto optimal set. Also, the application of this dominance relation is simple, and can help the decision maker, who has control over the set of the achieved solutions. As shown in Algorithm 3, the mechanism of $\epsilon$-dominance is simple. The update procedure is conducted with one solution from the $New_{sol}$ population against all the archive solutions. First, the $B$ vector values are computed for one solution $s$ from $New_{sol}$ and all the solutions of the archive, then a test is conducted to determine whether the solution $s$ will be part of the archive or not. Here, two cases are distinguished: first, if the identification vector $B_s$ of $s$, dominates the identification vector $B_x$, any solution of the archive denoted as $x$ then $s$ would be stored in the archive, while the solution $x$ would be removed, and if $B_s$ does not dominate $B_x$, then $s$ will not be added to the archive. Secondly, if $B_s$ does not dominate and is not dominated, then a regular dominance mechanism is used, where $s$ dominates $x$, then $s$ will be added to the archive.

**Algorithm 3** Updating archive solutions using $\epsilon$-dominance.

```
Input: Archive(t), iteration number t, New_s ols solution.
Calculate vector B_s and B_A for all archive population solutions Archive(t),
if ∃ x ∈ Archive(t)|B_x ⋡ B_s then
  | s is rejected
end
if ∃ x ∈ Archive(t)|B_s ⋡ B_x then
  | sreplacex ∈ Archive(t)
end
if both above cases does not occurs then then
  | if ∃ x ∈ Archive(t)|B_s ∼ B_x then
  |   | if s ∼ x then
  |   |   | Keep the solution with the smallest distance to vector B
  |   | else
  |   |   | Retain the solution dominating all other solutions
  |   | end
  | else
  |   | Add the solution s to the archive Archive(t)
  | end
end
```

Finally, MOGSK updates the senior/junior population status similarly to GSK, as shown in Algorithm 4; these procedures are repeated until the end criterion is met, which in our case is the max iteration. The whole process of the proposed MOGSK is shown in the flowchart of Figure 4. The complexity MOGSK is that of NSGAII, since the main point of it uses fast nondominated sorting and crowding distance; therefore, the complexity equals $O(MN^2)$, with $N$ population size, and $M$ the set of objectives.

**Algorithm 4** Multiobjective gaining–sharing knowledge optimization algorithm (MOGSK).

Initialize parameters $G = 0, N, k_f, k_r, K, P$
Create randomly first population $Pop^0$;
Calculate fitness value for $Pop^0$; **while** $Iteration < Max_{iteration}$ **do**
  **if** $iteration = 1$ **then**
    Sort $Pop^0$ using FNS
    Compute crowding distance
    Initialize $Archive$ with the first front of $Pop^0$
  **else**
    $New_{sol}$ = Combine $[Pop^{iteration}, Archive^{iteration-1}]$
    Sort $New_{sol}$ using FNS
    Compute crowding distance
    $New_{sol}$ = first $N$ solutions
    Update $Archive$ using $New_{sol}$ **if** $Archive > N$ **then**
      Sort $Archive$ using FNS
      Select the first $N$ solutions
    **end**
    Compute the number of (gained and shared dimensions of both phases)
    Junior gaining–sharing knowledge phase
    Senior gaining–sharing knowledge phase
    $iteration = itearation + 1$
  **end**
**end**
**Return**:$Archive$

## 4. Results and Discussion

### 4.1. Experiments Setup

To validate the designed MOGSK performance, a series of different experiments were conducted. The first experiment was carried out using the ZDT [57] and DTLZ [58] test functions, which include 12 distinct test benchmarks. In the second experiment and to further assess the quality of the proposed MOGSK, the recently introduced CEC 2021 real-world constrained multiobjective optimization problems (RWMOPs) [59] were employed. A comparison was conducted between MOGSK, MOEAD [11], eMOEA [12], MOPSO [25], NSGAII [9], SPEA2 [13], KnEA [14], and GrEA [15] using the statistical findings reached from the given test functions. The experiments conducted are listed below:

- Experiment I: ZDT, DTLZ test functions for MOPs.
- Experiment II: CEC 2021 test problems.

The stated number of runs was set to 30 independent runs, and there were 6000 function evaluations. As for the metrics used to compare MOGSK with other algorithms, the inverted generational distance (IGD) [4] and hyper volume indicator (HV) [60] were used. IGD is a metric used for assessing the quality of approximations towards the Pareto front achieved by a multiobjective optimization algorithm. IGD is formulated as:

$$IGD = \frac{\sqrt{\sum_{i=1}^{n} d_i^2}}{n} \quad (12)$$

where $n$ is the number of true Pareto solutions set, and $d_i^2$ is the Euclidean distance between the true Pareto front and the closest obtained Pareto solution. $HV$ assesses the outcome of an optimization algorithm by simultaneously taking into consideration the proximity of the points to the Pareto front, diversity, and spread. $HV$ is also known as the $S$ measure.

$HV$ refers to the volume space in the objective space dominated by the Pareto front $S$ and the $r \in R^m$ reference point as a bound, for all $z \in S, z \prec r$. The $HV$ is noted as:

$$HV(S, r) = \lambda_m \left( \bigcup_{z \in S} [z; r] \right) \tag{13}$$

where $\lambda_m$ refers to the $m$-dimensional Lebesgue measure.

Note that the platform PlatEMO [61] is used in Experiment II.

## 4.2. Experiment I

ZDT and DTLZ benchmark characteristics are listed in Table 2.

**Table 2.** Multiobjective test function characteristics.

| Biobjective Test Functions | |
|---|---|
| **Function** | **Description** |
| ZDT1 | Has a convex front |
| ZDT2 | Nonconvex front |
| ZDT3 | Has a discontinuous front |
| ZDT4 | Has 221 local Pareto optimal fronts, as results highly multimodal |
| ZDT6 | Has a nonuniform search space |
| **Three-Objective Test Functions** | |
| **Function** | **Description** |
| DTLZ1 | Has a linear Pareto optimal front (POF) |
| DTLZ2 | Has a spherical POF |
| DTLZ3 | Has many POFs |
| DTLZ4 | The POF has a dense set of solutions to exist near the $f_M - f_1$ |
| DTLZ5 | This problem will verify the ability to converge to a degenerated curve. |
| DTLZ6 | 2M-1 disconnected Pareto optimal front. |
| DTLZ7 | Has a POF that combines straight line and a hyperplane. |

### 4.2.1. ZDT Test Results

Table 3 describes the statistical results for the IGD metric. Table 4 reports the statistical results for the HV metric. Figure 6 illustrates the obtained results. Table 3 displays the results for the best, worst, average, median, and std for MOGSK, MOEAD, eMOEA, MOPSO, NSGAII, SPEA2, KnEA, and GrEA of the IGD metric. MOGSK was able to give the best results for the test function ZDT1, ZDT2, and ZDT6, where it was successful in surpassing all the comparative algorithms in performance. As for ZDT3, the best result was obtained by SPEA2, followed by NSGAII then KnEA, after which MOGSK came in fourth, followed by the rest of the comparative methods. The ZDT4 results show that the best IGD results were obtained by SPEA2 then NSGAII, followed by MOEAD, while MOGSK came in fourth place. Table 4, on the other hand, displays the outcomes attained regarding the best, worst, average, median, and std results of the HV metric, for all the test functions $ZDT_i(i, i = 1 \ldots 4)$, and ZDT6 MOGSK was able to surpass all the comparative performances of the algorithms, including MOEAD, eMOEA, MOPSO, NSGAII, SPEA2, KnEA, and GrEA. In addition, Figure 6 supports the statistical quantitative and qualitative results presented previously. MOGSK shows good converge and distribution for ZDT1, ZDT2, and ZDT6; as for ZDT3, MOGSK was able to converge towards three fronts of ZDT3 out of five discontinuous fronts. In ZDT4, MOGSK was stuck in a local optimum. Overall, MOGSK showed good behavior in this experiment using the ZDT test function, which shows the ability of MOGSK to be a useful optimization tool.

**Table 3.** IGD results on ZDT.

| Algorithm | MOGSK | MOEAD | eMOEA | MOPSO | NSGAII | SPEA2 | KnEA | GrEA |
|---|---|---|---|---|---|---|---|---|
| ZDT1 | | | | | | | | |
| Best | **1.88E-04** | 4.16E-03 | 2.48E-02 | 3.50E-01 | 4.42E-03 | 3.81E-03 | 3.07E-02 | 6.34E-03 |
| Worst | **2.59E-04** | 6.27E-03 | 3.34E-02 | 1.25E+00 | 5.41E-03 | 4.15E-03 | 3.74E-01 | 1.36E-02 |
| Average | **2.13E-04** | 4.81E-03 | 2.92E-02 | 7.42E-01 | 4.76E-03 | 3.96E-03 | 1.77E-01 | 7.36E-03 |
| median | **2.12E-04** | 4.70E-03 | 2.94E-02 | 7.30E-01 | 4.71E-03 | 3.95E-03 | 1.67E-01 | 7.14E-03 |
| Std | **1.56E-05** | 4.67E-04 | 2.17E-03 | 2.23E-01 | 2.17E-04 | 7.50E-05 | 9.36E-02 | 1.29E-03 |
| ZDT2 | | | | | | | | |
| Best | **1.83E-04** | 4.53E-03 | 2.68E-02 | 2.29E-02 | 4.55E-03 | 3.84E-03 | 5.87E-02 | 7.90E-03 |
| Worst | **1.24E-03** | 7.57E-03 | 3.73E-02 | 2.48E+00 | 5.34E-03 | 4.06E-03 | 1.26E-01 | 8.05E-03 |
| Average | **2.38E-04** | 5.42E-03 | 3.16E-02 | 1.68E+00 | 4.82E-03 | 3.94E-03 | 9.62E-02 | 8.00E-03 |
| median | **2.05E-04** | 5.22E-03 | 3.10E-02 | 1.77E+00 | 4.78E-03 | 3.94E-03 | 9.79E-02 | 8.01E-03 |
| Std | 1.89E-04 | 7.45E-04 | 3.15E-03 | 5.66E-01 | 1.82E-04 | 5.10E-05 | 1.83E-02 | **3.56E-05** |
| ZDT3 | | | | | | | | |
| Best | 8.01E-03 | 1.22E-02 | 4.03E-02 | 2.19E-01 | 5.11E-03 | **4.70E-03** | 7.22E-03 | 1.15E-02 |
| Worst | 9.33E-03 | 4.31E-02 | 9.07E-02 | 1.00E+00 | 6.47E-03 | **5.07E-03** | 3.92E-02 | 1.60E-02 |
| Average | 8.97E-03 | 1.97E-02 | 6.62E-02 | 6.47E-01 | 5.47E-03 | **4.91E-03** | 1.09E-02 | 1.42E-02 |
| median | 9.06E-03 | 1.37E-02 | 6.50E-02 | 6.29E-01 | 5.41E-03 | **4.92E-03** | 1.00E-02 | 1.41E-02 |
| Std | 2.30E-04 | 1.14E-02 | 9.94E-03 | 1.95E-01 | 2.71E-04 | **8.91E-05** | 5.50E-03 | 1.17E-03 |
| ZDT4 | | | | | | | | |
| Best | 6.57E-02 | 4.69E-03 | 2.62E-02 | 7.88E+00 | 4.39E-03 | **3.82E-03** | 1.32E-01 | 7.23E-02 |
| Worst | 2.53E-01 | 1.20E-02 | 3.62E-02 | 3.47E+01 | **4.93E-03** | 5.10E-03 | 3.74E-01 | 5.24E-01 |
| Average | 1.53E-01 | 7.76E-03 | 3.02E-02 | 1.57E+01 | 4.64E-03 | **4.06E-03** | 2.59E-01 | 3.11E-01 |
| median | 1.55E-01 | 7.20E-03 | 3.02E-02 | 1.55E+01 | 4.62E-03 | **3.94E-03** | 2.65E-01 | 3.08E-01 |
| Std | 5.57E-02 | 1.90E-03 | 2.28E-03 | 6.70E+00 | **1.53E-04** | 2.96E-04 | 5.94E-02 | 1.31E-01 |
| ZDT6 | | | | | | | | |
| Best | **1.53E-04** | 3.36E-03 | 2.49E-02 | 5.20E-03 | 3.49E-03 | 3.05E-03 | 5.02E-03 | 5.67E-03 |
| Worst | **1.13E-03** | 5.81E-03 | 3.18E-02 | 5.51E+00 | 3.90E-03 | 3.14E-03 | 1.41E-02 | 6.18E-03 |
| Average | **2.60E-04** | 4.68E-03 | 2.90E-02 | 1.90E-01 | 3.68E-03 | 3.09E-03 | 7.22E-03 | 6.02E-03 |
| median | **1.82E-04** | 4.70E-03 | 2.93E-02 | 6.64E-03 | 3.65E-03 | 3.09E-03 | 6.41E-03 | 6.03E-03 |
| Std | 1.96E-04 | 5.93E-04 | 1.86E-03 | 1.00E+00 | 1.07E-04 | **2.22E-05** | 2.15E-03 | 1.29E-04 |

**Table 4.** HV results on ZDT.

| Algorithm | MOGSK | MOEAD | eMOEA | MOPSO | NSGAII | SPEA2 | KnEA | GrEA |
|---|---|---|---|---|---|---|---|---|
| ZDT1 | | | | | | | | |
| Best | **4.44E+00** | 7.19E-01 | 0.00E+00 | 3.22E-01 | 7.20E-01 | 7.20E-01 | 7.00E-01 | 7.17E-01 |
| Worst | **1.93E-01** | 7.17E-01 | 3.22E-01 | 0.00E+00 | 7.18E-01 | 7.20E-01 | 4.95E-01 | 7.09E-01 |
| Average | **5.56E-01** | 7.18E-01 | 8.75E-02 | 8.75E-02 | 7.19E-01 | 7.20E-01 | 6.16E-01 | 7.16E-01 |
| median | **2.55E-01** | 7.18E-01 | 5.92E-02 | 5.92E-02 | 7.19E-01 | 7.20E-01 | 6.24E-01 | 7.16E-01 |
| Std | **1.04E+00** | 5.85E-04 | 9.49E-02 | 9.49E-02 | 2.98E-04 | 1.28E-04 | 5.52E-02 | 1.45E-03 |
| ZDT2 | | | | | | | | |
| Best | **4.61E+01** | 4.43E-01 | 0.00E+00 | 4.10E-01 | 4.44E-01 | 4.45E-01 | 3.90E-01 | 4.42E-01 |
| Worst | **2.60E-01** | 4.36E-01 | 4.10E-01 | 0.00E+00 | 4.44E-01 | 4.45E-01 | 3.31E-01 | 4.41E-01 |
| Average | **9.91E+00** | 4.41E-01 | 1.40E-02 | 1.40E-02 | 4.44E-01 | 4.45E-01 | 3.56E-01 | 4.41E-01 |
| median | **5.23E+00** | 4.42E-01 | 0.00E+00 | 0.00E+00 | 4.44E-01 | 4.45E-01 | 3.54E-01 | 4.41E-01 |
| Std | **1.00E+01** | 1.68E-03 | 7.49E-02 | 7.49E-02 | 2.07E-04 | 7.92E-05 | 1.59E-02 | 4.11E-05 |
| ZDT3 | | | | | | | | |
| Best | **1.77E+00** | 6.89E-01 | 1.33E-02 | 4.77E-01 | 6.00E-01 | 6.00E-01 | 6.87E-01 | 5.98E-01 |
| Worst | **1.39E-01** | 5.83E-01 | 4.77E-01 | 1.33E-02 | 5.99E-01 | 5.99E-01 | 5.97E-01 | 5.96E-01 |
| Average | **2.38E-01** | 6.15E-01 | 1.60E-01 | 1.60E-01 | 5.99E-01 | 6.00E-01 | 6.01E-01 | 5.97E-01 |
| median | **1.76E-01** | 5.98E-01 | 1.44E-01 | 1.44E-01 | 5.99E-01 | 6.00E-01 | 5.98E-01 | 5.97E-01 |
| Std | **2.93E-01** | 3.58E-02 | 1.16E-01 | 1.16E-01 | 1.16E-04 | 5.90E-05 | 1.63E-02 | 4.36E-04 |

**Table 4.** *Cont.*

| Algorithm | MOGSK | MOEAD | eMOEA | MOPSO | NSGAII | SPEA2 | KnEA | GrEA |
|---|---|---|---|---|---|---|---|---|
| ZDT4 | | | | | | | | |
| Best | **7.46E-01** | 7.17E-01 | 0.00E+00 | 0.00E+00 | 7.20E-01 | 7.20E-01 | 6.43E-01 | 6.56E-01 |
| Worst | **5.34E-01** | 7.07E-01 | 0.00E+00 | 0.00E+00 | 7.18E-01 | 7.17E-01 | 4.93E-01 | 3.85E-01 |
| Average | **6.51E-01** | 7.12E-01 | 0.00E+00 | 0.00E+00 | 7.19E-01 | 7.20E-01 | 5.67E-01 | 5.29E-01 |
| median | **6.38E-01** | 7.13E-01 | 0.00E+00 | 0.00E+00 | 7.19E-01 | 7.20E-01 | 5.65E-01 | 5.34E-01 |
| Std | **4.80E-02** | 2.83E-03 | 0.00E+00 | 0.00E+00 | 4.36E-04 | 8.59E-04 | 3.64E-02 | 8.34E-02 |
| ZDT6 | | | | | | | | |
| Best | **4.30E+00** | 3.88E-01 | 0.00E+00 | 3.86E-01 | 3.89E-01 | 3.89E-01 | 3.87E-01 | 3.86E-01 |
| Worst | **4.20E-01** | 3.84E-01 | 3.86E-01 | 0.00E+00 | 3.88E-01 | 3.89E-01 | 3.78E-01 | 3.86E-01 |
| Average | **2.89E+00** | 3.85E-01 | 3.69E-01 | 3.69E-01 | 3.88E-01 | 3.89E-01 | 3.85E-01 | 3.86E-01 |
| median | **3.26E+00** | 3.85E-01 | 3.82E-01 | 3.82E-01 | 3.88E-01 | 3.89E-01 | 3.86E-01 | 3.86E-01 |
| **Std** | **1.36E+00** | 1.01E-03 | 6.98E-02 | 6.98E-02 | 9.79E-05 | 2.52E-05 | 2.11E-03 | 1.33E-04 |

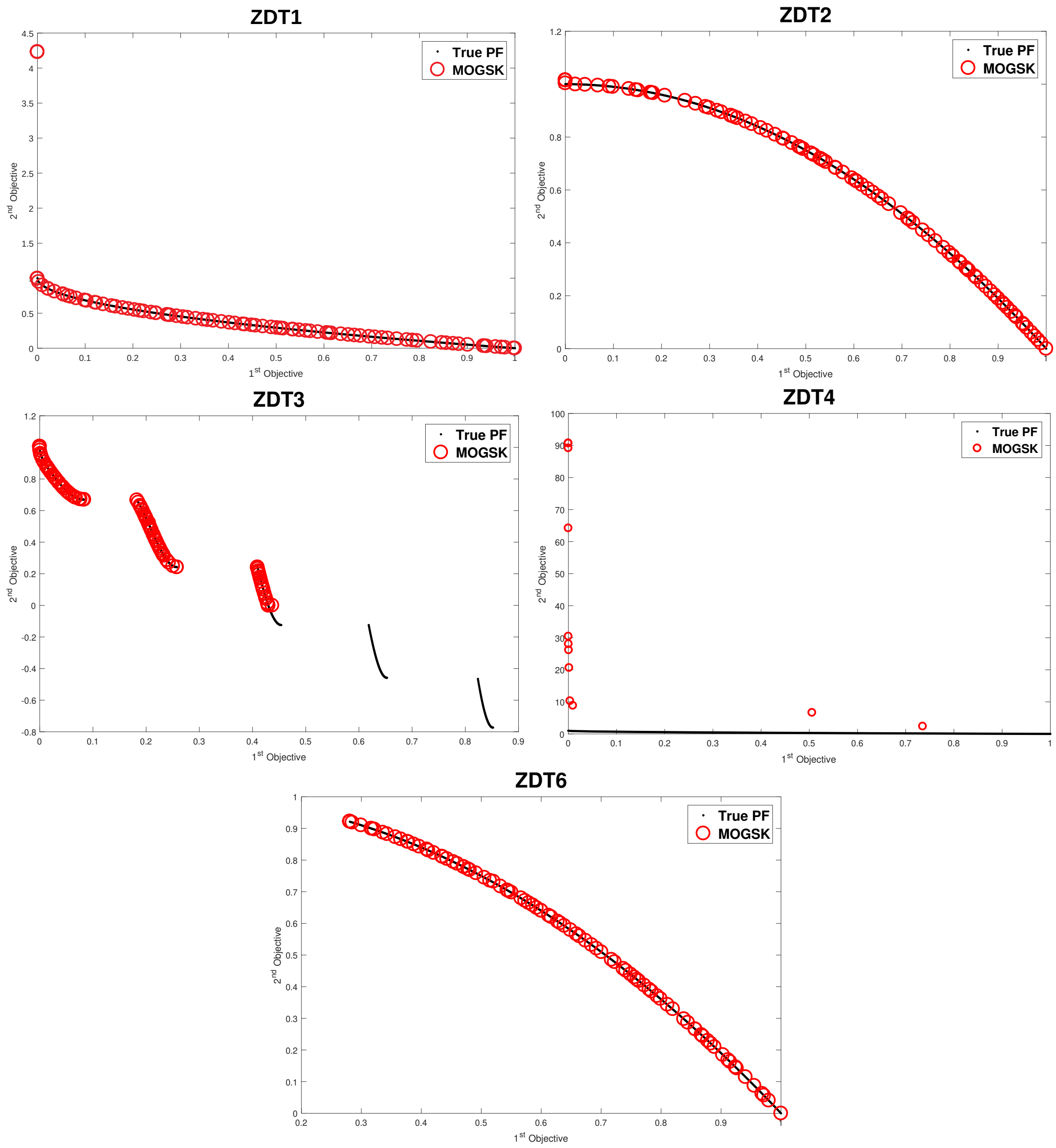


**Figure 6.** Pareto front obtained by MOGSK of ZDT test functions.

#### 4.2.2. DTLZ Test Results

Table 5 reports the statistical results for the IGD metric. Table 6 reports the statistical outcomes achieved for the HV metric. Figure 7 illustrates the obtained the results. Table 5 displays the results for five measures, which are the best, worst, average, median, and std for MOGSK, MOEAD, eMOEA, MOPSO, NSGAII, SPEA2, KnEA, and GrEA of the IGD metric. MOGSK in this test function series showed good results, where it was capable of topping the comparative algorithms in six out of seven test functions, which are DTLZ1, DZLZ2,DTLZ4, DTLZ5, and DTLZ6, with a gap particularly in DTLZ5 and DTLZ6; as for DTLZ3, the best results belong to SPEA2, whereas MOGSK came in last. Table 6 presents the acquired measures (best, worst, average, median, and std) results of the HV metric for the DTLZ test function. MOGSK showed good performance as well, where it was capable of surpassing the comparative algorithms in DTLZ2, DTLZ4, DTLZ5, DTLZ6 and DTLZ7; as for DTLZ1, the best result was reported to SPEA2, followed by MOEAD, NSGAII, KnEA, GrEA, and eMOEA, then MOGSK, then MOPSO. The DTLZ3 test function's best results were yielded by SPEA2 then GrEA, MOEA, eMOEA, NSGAII, and KnEA, then MOGSK, followed by MOPSO. These findings are supported by Figure 7; it is apparent to observe that MOGSK has good solution distribution and convergence for five of the seven test problems. Overall, MOGSK achieved excellent performance.

**Table 5.** IGD results on DTLZ.

| **Algorithm** | **MOGSK** | **MOEAD** | **eMOEA** | **MOPSO** | **NSGAII** | **SPEA2** | **KnEA** | **GrEA** |
|---|---|---|---|---|---|---|---|---|
| DTLZ1 | | | | | | | | |
| Best | **8.77E-03** | 2.06E-02 | 3.39E-02 | 7.33E-01 | 2.57E-02 | 1.99E-02 | 2.37E-02 | 2.36E-02 |
| Worst | 4.82E-02 | **2.09E-02** | 4.24E-02 | 1.17E+01 | 2.99E-02 | 2.08E-02 | 1.44E-01 | 3.36E-01 |
| Average | **2.03E-02** | 2.06E-02 | 3.67E-02 | 5.96E+00 | 2.73E-02 | 2.02E-02 | 5.58E-02 | 8.59E-02 |
| median | **1.60E-02** | 2.06E-02 | 3.66E-02 | 5.82E+00 | 2.75E-02 | 2.02E-02 | 4.30E-02 | 7.19E-02 |
| Std | 1.09E-02 | 8.00E-05 | 1.52E-03 | 2.52E+00 | 9.78E-04 | **1.63E-04** | 3.18E-02 | 6.84E-02 |
| DTLZ2 | | | | | | | | |
| Best | **1.00E-03** | 5.45E-02 | 6.07E-02 | 1.21E-01 | 6.48E-02 | 5.26E-02 | 6.31E-02 | 6.27E-02 |
| Worst | **1.16E-03** | 5.45E-02 | 6.72E-02 | 2.40E-01 | 7.46E-02 | 5.57E-02 | 7.43E-02 | 6.65E-02 |
| Average | **1.09E-03** | 5.45E-02 | 6.46E-02 | 1.69E-01 | 6.94E-02 | 5.43E-02 | 6.67E-02 | 6.39E-02 |
| median | **1.09E-03** | 5.45E-02 | 6.49E-02 | 1.67E-01 | 6.95E-02 | 5.42E-02 | 6.59E-02 | 6.38E-02 |
| Std | **4.00E-05** | 3.72E-07 | 1.42E-03 | 2.66E-02 | 2.03E-03 | 6.47E-04 | 2.88E-03 | 7.78E-04 |
| DTLZ3 | | | | | | | | |
| Best | 2.77E-01 | 5.47E-02 | 6.87E-02 | 1.63E+00 | 6.45E-02 | **5.31E-02** | 6.60E-02 | 6.38E-02 |
| Worst | 6.31E-01 | 1.06E-01 | 8.22E-01 | 1.72E+02 | 7.68E-02 | **6.60E-02** | 2.03E-01 | 5.34E-01 |
| Average | 4.30E-01 | 6.22E-02 | 1.09E-01 | 6.74E+01 | 7.13E-02 | **5.60E-02** | 1.01E-01 | 1.17E-01 |
| median | 4.38E-01 | 5.92E-02 | 7.87E-02 | 5.93E+01 | 7.15E-02 | **5.49E-02** | 8.81E-02 | 6.80E-02 |
| Std | 9.42E-02 | 9.96E-03 | 1.36E-01 | 4.73E+01 | 3.17E-03 | **2.98E-03** | 3.34E-02 | 1.06E-01 |
| DTLZ4 | | | | | | | | |
| Best | **2.36E-03** | 5.45E-02 | 6.51E-02 | 1.20E-01 | 6.35E-02 | 5.39E-02 | 6.06E-02 | 6.42E-02 |
| Worst | **5.44E-03** | 9.46E-01 | 5.53E-01 | 9.50E-01 | 7.12E-02 | 9.46E-01 | 9.46E-01 | 9.46E-01 |
| Average | **3.92E-03** | 2.57E-01 | 1.96E-01 | 3.14E-01 | 6.71E-02 | 2.46E-01 | 1.24E-01 | 2.36E-01 |
| median | **4.10E-03** | 5.45E-02 | 6.74E-02 | 2.71E-01 | 6.71E-02 | 5.51E-02 | 6.49E-02 | 6.73E-02 |
| Std | **8.39E-04** | 3.11E-01 | 2.18E-01 | 1.89E-01 | 2.02E-03 | 2.66E-01 | 2.23E-01 | 2.80E-01 |
| DTLZ5 | | | | | | | | |
| Best | **8.80E-05** | 3.38E-02 | 5.30E-02 | 8.41E-03 | 5.30E-03 | 4.17E-03 | 7.61E-03 | 2.00E-02 |
| Worst | **2.04E-04** | 3.39E-02 | 7.20E-02 | 2.20E-02 | 7.20E-03 | 4.67E-03 | 1.41E-02 | 2.44E-02 |
| Average | **1.23E-04** | 3.39E-02 | 6.70E-02 | 1.21E-02 | 5.87E-03 | 4.41E-03 | 9.41E-03 | 2.15E-02 |
| median | **1.14E-04** | 3.39E-02 | 6.80E-02 | 1.16E-02 | 5.84E-03 | 4.41E-03 | 9.12E-03 | 2.13E-02 |
| Std | **2.73E-05** | 2.72E-05 | 4.55E-03 | 2.74E-03 | 3.79E-04 | 1.28E-04 | 1.32E-03 | 9.55E-04 |
| DTLZ6 | | | | | | | | |
| Best | **2.24E-03** | 3.39E-02 | 5.85E-02 | 6.18E-01 | 5.48E-03 | 4.03E-03 | 4.38E-03 | 2.19E-02 |
| Worst | 3.38E-02 | 3.39E-02 | 6.59E-02 | 4.40E+00 | 6.78E-03 | **4.19E-03** | 5.95E-03 | 2.23E-02 |

**Table 5.** *Cont.*

| Algorithm | MOGSK | MOEAD | eMOEA | MOPSO | NSGAII | SPEA2 | KnEA | GrEA |
|---|---|---|---|---|---|---|---|---|
| Average | 5.66E-03 | 3.39E-02 | 6.27E-02 | 2.48E+00 | 5.92E-03 | **4.09E-03** | 4.89E-03 | 2.23E-02 |
| median | **3.78E-03** | 3.39E-02 | 6.29E-02 | 2.27E+00 | 5.87E-03 | 4.09E-03 | 4.86E-03 | 2.23E-02 |
| Std | 6.09E-03 | **1.23E-05** | 1.86E-03 | 1.16E+00 | 2.74E-04 | 4.00E-05 | 3.41E-04 | 9.20E-05 |
| DTLZ7 | | | | | | | | |
| Best | **8.53E-04** | 1.50E-01 | 5.76E-02 | 5.37E-01 | 7.00E-02 | 5.77E-02 | 5.79E-02 | 7.65E-02 |
| Worst | **1.30E-03** | 8.03E-01 | 8.14E-01 | 5.38E+00 | 8.76E-02 | 3.46E-01 | 3.52E-01 | 3.73E-01 |
| Average | **1.07E-03** | 1.77E-01 | 1.48E-01 | 2.85E+00 | 7.77E-02 | 7.90E-02 | 7.54E-02 | 9.36E-02 |
| median | **1.10E-03** | 1.55E-01 | 6.21E-02 | 2.81E+00 | 7.84E-02 | 6.01E-02 | 6.60E-02 | 8.42E-02 |
| Std | **1.11E-04** | 1.18E-01 | 1.77E-01 | 1.28E+00 | 4.38E-03 | 7.24E-02 | 5.23E-02 | 5.29E-02 |

**Table 6.** HV results on DTLZ.

| Algorithm | MOGSK | MOEAD | eMOEA | MOPSO | NSGAII | SPEA2 | KnEA | GrEA |
|---|---|---|---|---|---|---|---|---|
| DTLZ1 | | | | | | | | |
| Best | 5.67E-01 | 8.42E-01 | 7.77E-01 | 0.00E+00 | 8.28E-01 | **8.43E-01** | 8.21E-01 | 8.13E-01 |
| Worst | 3.20E-01 | 8.38E-01 | 6.93E-01 | 0.00E+00 | 8.16E-01 | **8.38E-01** | 5.65E-01 | 2.41E-01 |
| Average | 4.76E-01 | 8.41E-01 | 7.27E-01 | 0.00E+00 | 8.23E-01 | **8.41E-01** | 7.39E-01 | 6.79E-01 |
| median | 4.97E-01 | 8.41E-01 | 7.27E-01 | 0.00E+00 | 8.23E-01 | **8.42E-01** | 7.54E-01 | 6.97E-01 |
| Std | 7.12E-02 | 7.62E-04 | 1.71E-02 | 0.00E+00 | 3.13E-03 | **1.25E-03** | 5.90E-02 | 1.27E-01 |
| DTLZ2 | | | | | | | | |
| Best | **9.74E+00** | 5.60E-01 | 5.50E-01 | 4.12E-01 | 5.38E-01 | 5.57E-01 | 5.48E-01 | 5.60E-01 |
| Worst | **2.72E-01** | 5.60E-01 | 5.42E-01 | 2.85E-01 | 5.26E-01 | 5.53E-01 | 5.32E-01 | 5.57E-01 |
| Average | **1.12E+00** | 5.60E-01 | 5.46E-01 | 3.50E-01 | 5.32E-01 | 5.56E-01 | 5.44E-01 | 5.58E-01 |
| median | **3.24E-01** | 5.60E-01 | 5.46E-01 | 3.48E-01 | 5.33E-01 | 5.56E-01 | 5.45E-01 | 5.58E-01 |
| Std | **2.16E+00** | 5.63E-06 | 2.00E-03 | 3.13E-02 | 3.34E-03 | 1.10E-03 | 3.39E-03 | 7.19E-04 |
| DTLZ3 | | | | | | | | |
| Best | 2.54E-01 | 5.56E-01 | 5.44E-01 | 0.00E+00 | 5.37E-01 | **5.61E-01** | 5.36E-01 | 5.58E-01 |
| Worst | 2.52E-01 | 4.52E-01 | 1.48E-03 | 0.00E+00 | 4.88E-01 | **5.15E-01** | 3.97E-01 | 3.24E-01 |
| Average | 2.54E-01 | 5.30E-01 | 4.90E-01 | 0.00E+00 | 5.20E-01 | **5.46E-01** | 5.00E-01 | 5.10E-01 |
| median | 2.54E-01 | 5.36E-01 | 5.07E-01 | 0.00E+00 | 5.22E-01 | **5.49E-01** | 5.13E-01 | 5.49E-01 |
| Std | 2.93E-04 | 2.19E-02 | 9.43E-02 | 0.00E+00 | 1.23E-02 | **1.03E-02** | 3.62E-02 | 7.46E-02 |
| DTLZ4 | | | | | | | | |
| Best | **9.83E-01** | 5.60E-01 | 5.54E-01 | 4.81E-01 | 5.40E-01 | 5.57E-01 | 5.51E-01 | 5.60E-01 |
| Worst | **8.25E-01** | 9.09E-02 | 3.07E-01 | 8.36E-02 | 5.20E-01 | 9.09E-02 | 9.09E-02 | 9.09E-02 |
| Average | **9.31E-01** | 4.62E-01 | 4.86E-01 | 3.86E-01 | 5.34E-01 | 4.70E-01 | 5.15E-01 | 4.74E-01 |
| median | **9.42E-01** | 5.60E-01 | 5.49E-01 | 4.20E-01 | 5.35E-01 | 5.54E-01 | 5.45E-01 | 5.59E-01 |
| Std | **3.88E-02** | 1.56E-01 | 1.08E-01 | 1.05E-01 | 4.53E-03 | 1.22E-01 | 1.15E-01 | 1.42E-01 |
| DTLZ5 | | | | | | | | |
| Best | **5.42E+04** | 1.82E-01 | 1.71E-01 | 1.95E-01 | 2.00E-01 | 2.00E-01 | 1.96E-01 | 1.89E-01 |
| Worst | **5.40E+03** | 1.82E-01 | 1.63E-01 | 1.74E-01 | 1.98E-01 | 1.99E-01 | 1.89E-01 | 1.88E-01 |
| Average | **2.07E+04** | 1.82E-01 | 1.69E-01 | 1.90E-01 | 1.99E-01 | 2.00E-01 | 1.94E-01 | 1.88E-01 |
| median | **1.46E+04** | 1.82E-01 | 1.69E-01 | 1.91E-01 | 1.99E-01 | 2.00E-01 | 1.94E-01 | 1.88E-01 |
| Std | **1.29E+04** | 1.49E-05 | 1.78E-03 | 4.54E-03 | 2.30E-04 | 1.76E-04 | 1.52E-03 | 3.50E-04 |
| DTLZ6 | | | | | | | | |
| Best | **5.86E-01** | 1.82E-01 | 1.78E-01 | 0.00E+00 | 2.00E-01 | 2.00E-01 | 2.00E-01 | 1.88E-01 |
| Worst | **4.61E-01** | 1.82E-01 | 1.76E-01 | 0.00E+00 | 1.99E-01 | 2.00E-01 | 1.98E-01 | 1.88E-01 |
| Average | **5.19E-01** | 1.82E-01 | 1.77E-01 | 0.00E+00 | 1.99E-01 | 2.00E-01 | 2.00E-01 | 1.88E-01 |
| median | **5.21E-01** | 1.82E-01 | 1.77E-01 | 0.00E+00 | 1.99E-01 | 2.00E-01 | 2.00E-01 | 1.88E-01 |
| Std | **3.91E-02** | 6.44E-06 | 6.29E-04 | 0.00E+00 | 1.17E-04 | 4.09E-05 | 3.04E-04 | 1.68E-05 |

**Table 6.** *Cont.*

| Algorithm | MOGSK | MOEAD | eMOEA | MOPSO | NSGAII | SPEA2 | KnEA | GrEA |
|---|---|---|---|---|---|---|---|---|
| DTLZ7 | | | | | | | | |
| Best | **4.85E+03** | 2.58E-01 | 2.70E-01 | 1.86E-01 | 2.73E-01 | 2.78E-01 | 2.79E-01 | 2.75E-01 |
| Worst | **1.82E+02** | 2.02E-01 | 1.93E-01 | 0.00E+00 | 2.65E-01 | 2.43E-01 | 2.41E-01 | 2.31E-01 |
| Average | **1.80E+03** | 2.54E-01 | 2.56E-01 | 1.18E-02 | 2.68E-01 | 2.75E-01 | 2.76E-01 | 2.70E-01 |
| median | **1.87E+03** | 2.56E-01 | 2.65E-01 | 0.00E+00 | 2.68E-01 | 2.77E-01 | 2.78E-01 | 2.69E-01 |
| Std | **1.35E+03** | 9.85E-03 | 1.86E-02 | 4.05E-02 | 1.88E-03 | 8.51E-03 | 6.77E-03 | 7.95E-03 |

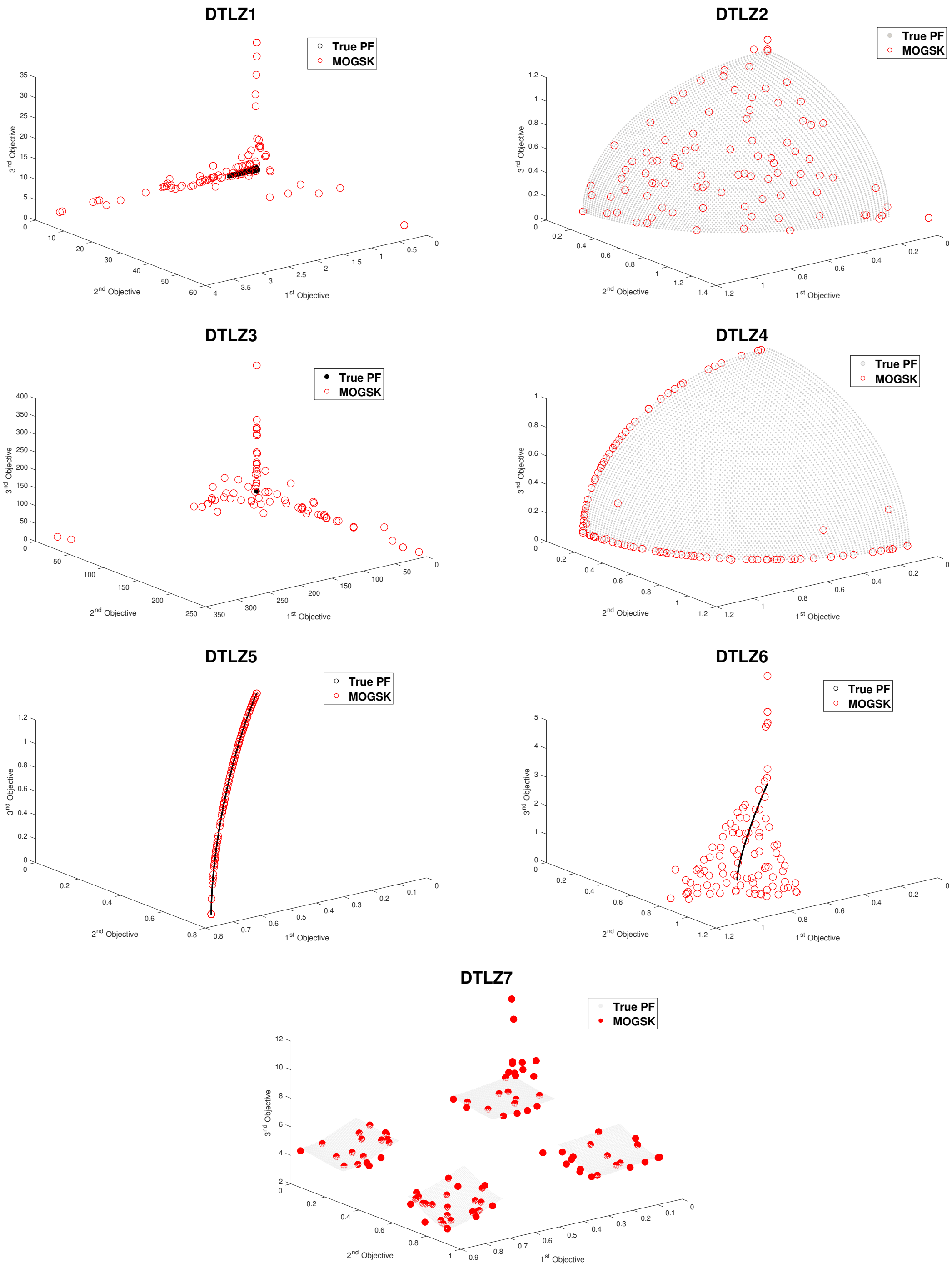


**Figure 7.** Pareto front generated by MOGSK of DTLZ test problems.

### 4.3. Experiment II

During this experiment, the CEC 2021 RWMOPs test problems were used. The RWMOPs has fifty different problems, including mechanical design problems (from RWMOP1 to RWMOP21); chemical engineering problems (from RWMOP22 to RWMOP24); process, design, and synthesis problems (from RWMOP25 to RWMOP29); power electronics problems (from RWMOP30 to RWMOP35); and power system optimization problems (from RWMOP36 to RWMOP50). Table 7 displays the fifty different problems.

**Table 7.** Real-world constrained multiobjective optimization problems.

| Name | Problem |
|---|---|
| **Mechanical Design Problems** | |
| RWMOP1 | Design of Pressure Vessels [62] |
| RWMOP2 | Design of Vibrating Platform [63] |
| RWMOP3 | Design of Two-Bar Truss [64] |
| RWMOP4 | Design of Welded Beam [65] |
| RWMOP5 | Disc Brake Design [66] |
| RWMOP6 | Speed Reducer Design [67] |
| RWMOP7 | Gear Train Design [68] |
| RWMOP8 | Car Side Impact Design [69] |
| RWMOP9 | Four-Bar Plane Truss [70] |
| RWMOP10 | Two-Bar Plane Truss |
| RWMOP11 | Water Resource Management |
| RWMOP12 | Simply Supported I-beam Design [71] |
| RWMOP13 | Gear Box Design |
| RWMOP14 | Multiple-Disk Clutch Brake Design [72] |
| RWMOP15 | Spring Design [62] |
| RWMOP16 | Cantilever Beam Design [73] |
| RWMOP17 | Bulk Carrier Design [74] |
| RWMOP18 | Front Rail Design [75] |
| RWMOP19 | Multiproduct Batch Plant [76] |
| RWMOP20 | Hydrostatic Thrust Bearing Design [77] |
| RWMOP21 | Crash Energy Management for High-Speed Train Problem [78] |
| **Chemical Engineering Problems** | |
| RWMOP22 | Problem of Haverly's Pooling Test [79] |
| RWMOP23 | Reactor Network Design [80] |
| RWMOP24 | Heat Exchanger Network Design [81] |
| **Process, Design, and Synthesis Problems** | |
| RWMOP25 | Process Synthesis Problem [82] |
| RWMOP26 | Process Synthesis, and Design Problem [83] |
| RWMOP27 | Process Flow Sheeting Problem [84] |
| RWMOP28 | Two-Reactor Problem [82] |
| RWMOP29 | Process Synthesis Problem [82] |

**Table 7.** *Cont.*

| Name | Problem |
|---|---|
| **Power Electronics Problems** | |
| RWMOP30 | The problem of Synchronous Optimal Pulse-Width Modulation of 3-Level Inverters [85] |
| RWMOP31 | The problem of Synchronous Optimal Pulse-Width Modulation of 5-Level Inverters [86] |
| RWMOP32 | The problem of Synchronous Optimal Pulse-Width Modulation of 7-Level Inverters [87] |
| RWMOP33 | The problem of Synchronous Optimal Pulse-Width Modulation of 9-Level Inverters [88] |
| RWMOP34 | The problem of Synchronous Optimal Pulse-Width Modulation of 11-Level inverters [89] |
| RWMOP35 | Synchronous Optimal Pulse-Width Modulation of 13-Level Inverters [89] |
| **Power System Optimization Problems** | |
| RWMOP36 | The Problem of Optimal Sizing of Single-Phase Distribution Generation with Reactive Power Support for Phase Balancing at Main Transformer/Grid and Reducing Active Power Loss [90] |
| RWMOP37 | The Problem of Optimal Sizing of Single-Phase Distribution Generation with Reactive Power Support for Phase Balancing at Main Transformer/Grid and Reducing Reactive Power Loss [90] |
| RWMOP38 | The Problem of Optimal Sizing of Single-Phase Distribution Generation with Reactive Power Support for Reducing Active and Reactive Power Loss [90] |
| RWMOP39 | The Problem of Optimal Sizing of Single-Phase Distribution Generation with Reactive Power Support for Phase Balancing at Main Transformer/Grid and Reducing Active and Reactive Power Loss [90] |
| RWMOP40 | The Problem of Optimal Power Flow for Reducing Active and Reactive Power Loss [91] |
| RWMOP41 | The Problem of Optimal Power Flow for Reducing Voltage Deviation, Active and Reactive Power Loss [92] |
| RWMOP42 | The Problem of Optimal Power Flow for Reducing Voltage Deviation and Active Power Loss [93] |
| RWMOP43 | The Problem of Optimal Power Flow for Reducing Fuel Cost and Active Power Loss [94] |
| RWMOP44 | Optimal Power Flow for reducing Fuel Cost, Active and Reactive Power Loss [95] |
| RWMOP45 | Optimal Power Flow for Reducing Fuel Cost, Voltage Deviation, and Active Power Loss [91] |
| RWMOP46 | Optimal Power Flow for Minimizing Fuel Cost, Voltage Deviation, Active and Reactive Power Loss [91] |
| RWMOP47 | The Problem of Optimal Droop Setting for Reducing Active and Reactive Power Loss [96] |
| RWMOP48 | The Problem of Optimal Droop Setting for Reducing Voltage Deviation and Active Power Loss [97] |
| RWMOP49 | The Problem of Optimal Droop Setting for Reducing Voltage Deviation, Active and Reactive Power Loss [98] |
| RWMOP50 | Power Distribution System Planning [99] |

For this experiment, the results for the HV metric are reported in Tables 8–12, respectively, as well as in Figures 8–12.

Table 8 displays the results for the best, worst, average, median, and std for MOGSK, MOEAD, eMOEA, MOPSO, NSGAII, SPEA2, KnEA, and GrEA of the HV metric, using mechanical design problems (21 problems in total), while Figure 8 shows the HV curve. For this problem, series MOGSK showed good performance, where it was capable of topping the comparative algorithms' performance in eleven out of twenty-one problems, namely RWMOP1, RWMOP4, RWMOP5, RWMOP6, RWMOP8, RWMOP12, RWMOP13, RWMOP14, RWMOP15, RWMOP18, and RWMOP19. As for RWMOP2, the best result was reported for toMOEAD, eMOEA, MOPSO, SPEA2, and GrEA with the same results, while MOGSK came in second place by a close margin, followed by KnEA and NSGAII. For RWMOP3, the best result was yielded by NSGAII, then KnEA, while MOGSK came in third place, followed by the remainder of the comparative algorithms. In RWMOP7 and RWMOP10, the results for all the algorithms, including MOGSK, gave almost the same results overall. RWMOP9 showed a very close range of the best for MOGSK, MOPSO, NSGAII, SPEA2, and GrEA, while MOEAD, eMOEA, and KnEA did not perform well on this problem. In the RWMOP11 problem, the best results were given by eMOEA, followed by MOGSk, NSGAII, KnEA, and GrEA, with a not-so-big gap in their results, while MOEAD, MOPSO, and SPEA2 came in last. RWMOP16 showed almost the same behavior, where the best results yielded a quite close range between MOGSK, MOPSO, NSGAII, SPEA2 and KnEA, followed by MOEAD, eMOEA, and GrEA. Using RWMOP17, the best results by a large gap were reported for SPEA2, followed by MOPSO, eMOEA, and GrEA; therefore, MOGSK was in fifth place, where it was able to top MOEAD, NSGAII and KnEA. RWMOP20's best results were reported equally for MOEAD, eMOEA, MOPSO, SPEA2, and GrEA after which came MOGSK, followed by NSGAII and KnEA. For the last test function in the mechanical design problems, RWMOP21, overall, the results were close, where the best results were reported for SPEA2, NSGAII, MOPSO, GrEA, eMOEA, and MOGSK, outperforming MOEAD and KnEA. Supporting the obtained results, in Figure 8 where the HV curves for the different test problems are shown, overall, MOGSK performed well using the mechanical design problems, where it was able to give the best results for most test problems and close to the best for the others.

**Table 8.** HV results using the mechanical design problems.

| Algorithm Problem | MOGSK | MOEAD | eMOEA | MOPSO | NSGAII | SPEA2 | KnEA | GrEA |
|---|---|---|---|---|---|---|---|---|
| RWMOP1 | | | | | | | | |
| Best | **1.00E+00** | 9.89E-01 | 1.00E+00 | 9.97E-01 | 6.06E-01 | 9.99E-01 | 6.31E-01 | 9.89E-01 |
| Worst | **1.00E+00** | 9.89E-01 | 9.89E-01 | 9.89E-01 | 6.04E-01 | 9.89E-01 | 5.82E-01 | 9.89E-01 |
| Average | **1.00E+00** | 9.89E-01 | 9.95E-01 | 9.91E-01 | 6.05E-01 | 9.93E-01 | 5.93E-01 | 9.89E-01 |
| median | **1.00E+00** | 9.89E-01 | 9.95E-01 | 9.90E-01 | 6.05E-01 | 9.93E-01 | 5.90E-01 | 9.89E-01 |
| Std | **2.94E-06** | 1.13E-16 | 3.69E-03 | 2.54E-03 | 5.06E-04 | 3.49E-03 | 9.71E-03 | 1.13E-16 |
| RWMOP2 | | | | | | | | |
| Best | 9.84E-01 | **1.00E+00** | **1.00E+00** | **1.00E+00** | 3.93E-01 | **1.00E+00** | 3.94E-01 | **1.00E+00** |
| Worst | 3.90E-01 | **1.00E+00** | **1.00E+00** | **1.00E+00** | 0.00E+00 | **1.00E+00** | 0.00E+00 | **1.00E+00** |
| Average | 6.96E-01 | **1.00E+00** | **1.00E+00** | **1.00E+00** | 2.70E-01 | **1.00E+00** | 2.62E-01 | **1.00E+00** |
| median | 6.48E-01 | **1.00E+00** | **1.00E+00** | **1.00E+00** | 3.03E-01 | **1.00E+00** | 2.74E-01 | **1.00E+00** |
| Std | 1.87E-01 | **0.00E+00** | **0.00E+00** | **0.00E+00** | 1.40E-01 | **1.00E+00** | 1.32E-01 | **0.00E+00** |
| RWMOP3 | | | | | | | | |
| Best | 6.62E-01 | 0.00E+00 | 4.73E-01 | 0.00E+00 | **9.02E-01** | 3.42E-01 | 8.99E-01 | 6.03E-01 |
| Worst | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | **9.02E-01** | 0.00E+00 | 8.57E-01 | 0.00E+00 |
| Average | 1.10E-01 | 0.00E+00 | 1.58E-02 | 0.00E+00 | **9.02E-01** | 4.55E-02 | 8.87E-01 | 1.98E-01 |
| median | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | **9.02E-01** | 0.00E+00 | 8.89E-01 | 0.00E+00 |
| Std | 2.51E-01 | 0.00E+00 | 8.64E-02 | 0.00E+00 | **1.55E-04** | 8.43E-02 | 9.96E-03 | 2.69E-01 |
| RWMOP4 | | | | | | | | |
| Best | **8.74E-01** | 0.00E+00 | 0.00E+00 | 7.90E-01 | 8.63E-01 | 1.54E-01 | 7.93E-01 | 6.32E-01 |
| Worst | **8.69E-01** | 0.00E+00 | 0.00E+00 | 0.00E+00 | 8.57E-01 | 0.00E+00 | 6.89E-01 | 0.00E+00 |
| Average | **8.72E-01** | 0.00E+00 | 0.00E+00 | 3.67E-01 | 8.61E-01 | 5.15E-03 | 7.37E-01 | 2.11E-02 |
| median | **8.72E-01** | 0.00E+00 | 0.00E+00 | 4.20E-01 | 8.62E-01 | 0.00E+00 | 7.36E-01 | 0.00E+00 |
| Std | **1.14E-03** | 0.00E+00 | 0.00E+00 | 2.98E-01 | 1.59E-03 | 2.82E-02 | 2.34E-02 | 1.15E-01 |
| RWMOP5 | | | | | | | | |
| Best | **6.30E-01** | 5.77E-01 | 2.51E-01 | 2.76E-01 | 4.35E-01 | 2.77E-01 | 4.14E-01 | 2.77E-01 |
| Worst | **6.29E-01** | 5.77E-01 | 2.48E-01 | 2.56E-01 | 4.27E-01 | 2.67E-01 | 3.18E-01 | 2.61E-01 |

**Table 8.** *Cont.*

| Algorithm Problem | MOGSK | MOEAD | eMOEA | MOPSO | NSGAII | SPEA2 | KnEA | GrEA |
|---|---|---|---|---|---|---|---|---|
| Average | **6.30E-01** | 5.77E-01 | 2.49E-01 | 2.67E-01 | 4.33E-01 | 2.73E-01 | 3.96E-01 | 2.74E-01 |
| median | **6.30E-01** | 5.77E-01 | 2.50E-01 | 2.68E-01 | 4.34E-01 | 2.73E-01 | 4.02E-01 | 2.74E-01 |
| Std | **1.27E-04** | 2.59E-08 | 7.58E-04 | 6.32E-03 | 1.75E-03 | 2.60E-03 | 2.20E-02 | 2.70E-03 |
| RWMOP6 | | | | | | | | |
| Best | **3.20E-01** | 6.87E-02 | 3.07E-01 | 3.07E-01 | 2.77E-01 | 3.00E-01 | 2.72E-01 | 2.45E-01 |
| Worst | **3.19E-01** | 6.87E-02 | 1.11E-01 | 0.00E+00 | 2.77E-01 | 0.00E+00 | 1.84E-01 | 1.99E-02 |
| Average | **3.19E-01** | 6.87E-02 | 2.17E-01 | 1.15E-01 | 2.77E-01 | 1.69E-01 | 2.16E-01 | 1.47E-01 |
| median | **3.19E-01** | 6.87E-02 | 2.26E-01 | 9.27E-02 | 2.77E-01 | 1.53E-01 | 2.02E-01 | 1.53E-01 |
| Std | **5.63E-05** | 1.34E-05 | 5.89E-02 | 1.04E-01 | 3.27E-05 | 9.98E-02 | 2.90E-02 | 5.54E-02 |
| RWMOP7 | | | | | | | | |
| Best | 4.83E-01 | 4.81E-01 | 4.84E-01 | 4.82E-01 | **4.84E-01** | 4.83E-01 | 4.84E-01 | 4.82E-01 |
| Worst | 4.70E-01 | 4.80E-01 | 4.83E-01 | 4.28E-01 | **4.84E-01** | 4.82E-01 | 4.80E-01 | 4.81E-01 |
| Average | 4.81E-01 | 4.81E-01 | 4.83E-01 | 4.69E-01 | **4.84E-01** | 4.83E-01 | 4.82E-01 | 4.82E-01 |
| median | 4.82E-01 | 4.81E-01 | 4.84E-01 | 4.73E-01 | **4.84E-01** | 4.83E-01 | 4.82E-01 | 4.82E-01 |
| Std | 2.38E-03 | 4.38E-04 | 2.39E-04 | 1.39E-02 | **7.48E-05** | 2.38E-04 | 1.01E-03 | 3.09E-04 |
| RWMOP8 | | | | | | | | |
| Best | **2.64E-02** | 0.00E+00 | 2.28E-02 | 2.34E-02 | 2.60E-02 | 2.43E-02 | 2.57E-02 | 2.36E-02 |
| Worst | **2.49E-02** | 0.00E+00 | 1.07E-02 | 1.79E-02 | 2.57E-02 | 2.24E-02 | 2.39E-02 | 2.22E-02 |
| Average | **2.58E-02** | 0.00E+00 | 2.09E-02 | 2.18E-02 | 2.59E-02 | 2.35E-02 | 2.52E-02 | 2.27E-02 |
| Median | **2.59E-02** | 0.00E+00 | 2.23E-02 | 2.20E-02 | 2.59E-02 | 2.35E-02 | 2.53E-02 | 2.26E-02 |
| Std | **4.12E-04** | 0.00E+00 | 3.32E-03 | 1.46E-03 | 8.49E-05 | 4.53E-04 | 4.38E-04 | 2.94E-04 |
| RWMOP9 | | | | | | | | |
| Best | 4.08E-01 | 5.32E-02 | 3.39E-01 | 4.08E-01 | 4.09E-01 | **4.10E-01** | 3.84E-01 | 4.05E-01 |
| Worst | 4.02E-01 | 5.30E-02 | 1.45E-01 | 4.04E-01 | 4.09E-01 | **4.09E-01** | 3.44E-01 | 3.95E-01 |
| Average | 4.07E-01 | 5.31E-02 | 2.56E-01 | 4.07E-01 | 4.09E-01 | **4.09E-01** | 3.67E-01 | 4.01E-01 |
| median | 4.07E-01 | 5.31E-02 | 2.59E-01 | 4.07E-01 | 4.09E-01 | **4.09E-01** | 3.67E-01 | 4.02E-01 |
| Std | 1.44E-03 | 5.82E-05 | 5.06E-02 | 9.20E-04 | 1.73E-04 | **1.38E-04** | 7.50E-03 | 2.61E-03 |
| RWMOP10 | | | | | | | | |
| Best | 8.46E-01 | 8.01E-02 | 8.19E-01 | 8.47E-01 | **8.48E-01** | 8.44E-01 | 8.47E-01 | 8.42E-01 |
| Worst | 8.42E-01 | 7.84E-02 | 1.86E-01 | 8.45E-01 | **8.47E-01** | 8.35E-01 | 8.06E-01 | 8.14E-01 |
| Average | 8.45E-01 | 7.97E-02 | 6.25E-01 | 8.47E-01 | **8.47E-01** | 8.41E-01 | 8.27E-01 | 8.36E-01 |
| median | 8.45E-01 | 8.00E-02 | 7.54E-01 | 8.47E-01 | **8.47E-01** | 8.41E-01 | 8.23E-01 | 8.38E-01 |
| Std | 1.14E-03 | 5.76E-04 | 2.25E-01 | 3.80E-04 | **1.54E-04** | 2.41E-03 | 1.30E-02 | 7.01E-03 |
| RWMOP11 | | | | | | | | |
| Best | 9.58E-02 | 5.80E-02 | **1.08E-01** | 6.45E-02 | 9.69E-02 | 7.53E-02 | 9.96E-02 | 9.08E-02 |
| Worst | 9.03E-02 | 5.62E-02 | **1.07E-01** | 0.00E+00 | 9.16E-02 | 3.52E-02 | 9.47E-02 | 7.37E-02 |
| Average | 9.36E-02 | 5.74E-02 | **1.08E-01** | 1.23E-02 | 9.47E-02 | 6.03E-02 | 9.77E-02 | 8.32E-02 |
| median | 9.41E-02 | 5.76E-02 | **1.08E-01** | 0.00E+00 | 9.49E-02 | 6.42E-02 | 9.81E-02 | 8.28E-02 |
| Std | 1.60E-03 | 5.69E-04 | **2.54E-04** | 1.90E-02 | 1.26E-03 | 1.16E-02 | 1.35E-03 | 4.24E-03 |
| RWMOP12 | | | | | | | | |
| Best | **5.70E-01** | 0.00E+00 | 0.00E+00 | 5.47E-01 | 5.61E-01 | 5.55E-01 | 5.45E-01 | 0.00E+00 |
| Worst | **5.62E-01** | 0.00E+00 | 0.00E+00 | 4.67E-01 | 5.59E-01 | 5.22E-01 | 5.07E-01 | 0.00E+00 |
| Average | **5.69E-01** | 0.00E+00 | 0.00E+00 | 5.19E-01 | 5.60E-01 | 5.35E-01 | 5.32E-01 | 0.00E+00 |
| median | **5.69E-01** | 0.00E+00 | 0.00E+00 | 5.31E-01 | 5.60E-01 | 5.34E-01 | 5.30E-01 | 0.00E+00 |
| Std | **1.64E-03** | 0.00E+00 | 0.00E+00 | 2.71E-02 | 3.38E-04 | 8.61E-03 | 7.53E-03 | 0.00E+00 |
| RWMOP13 | | | | | | | | |
| Best | **9.86E-02** | 2.47E-02 | 8.09E-02 | 7.69E-02 | 9.00E-02 | 8.55E-02 | 9.00E-02 | 9.11E-02 |
| Worst | **9.82E-02** | 2.46E-02 | 4.38E-03 | 0.00E+00 | 8.91E-02 | 0.00E+00 | 7.20E-02 | 5.45E-02 |

**Table 8.** *Cont.*

| Algorithm Problem | MOGSK | MOEAD | eMOEA | MOPSO | NSGAII | SPEA2 | KnEA | GrEA |
|---|---|---|---|---|---|---|---|---|
| Average | **9.84E-02** | 2.46E-02 | 5.52E-02 | 2.23E-02 | 8.96E-02 | 3.57E-02 | 8.92E-02 | 7.94E-02 |
| median | **9.83E-02** | 2.46E-02 | 5.78E-02 | 1.75E-02 | 8.96E-02 | 3.49E-02 | 8.98E-02 | 8.18E-02 |
| Std | **1.29E-04** | 8.62E-06 | 1.80E-02 | 2.16E-02 | 2.18E-04 | 1.95E-02 | 3.25E-03 | 1.05E-02 |
| RWMOP14 | | | | | | | | |
| Best | **7.16E-01** | 1.29E-01 | 3.29E-01 | 5.93E-01 | 6.19E-01 | 3.52E-01 | 6.06E-01 | 5.88E-01 |
| Worst | **7.15E-01** | 1.29E-01 | 7.00E-02 | 0.00E+00 | 6.14E-01 | 3.42E-01 | 5.80E-01 | 3.37E-01 |
| Average | **7.16E-01** | 1.29E-01 | 1.44E-01 | 3.26E-01 | 6.18E-01 | 3.49E-01 | 5.98E-01 | 4.86E-01 |
| Median | **7.16E-01** | 1.29E-01 | 1.16E-01 | 3.36E-01 | 6.18E-01 | 3.49E-01 | 6.00E-01 | 4.90E-01 |
| Std | **2.69E-04** | 1.79E-07 | 8.06E-02 | 1.12E-01 | 9.97E-04 | 2.38E-03 | 6.11E-03 | 6.02E-02 |
| RWMOP15 | | | | | | | | |
| Best | **7.85E-01** | 7.53E-01 | 7.55E-01 | 7.55E-01 | 5.43E-01 | 7.50E-01 | 5.28E-01 | 0.00E+00 |
| Worst | **7.84E-01** | 7.53E-01 | 0.00E+00 | 0.00E+00 | 5.42E-01 | 6.02E-01 | 3.57E-01 | 0.00E+00 |
| Average | **7.84E-01** | 7.53E-01 | 2.66E-01 | 3.32E-01 | 5.43E-01 | 7.01E-01 | 4.54E-01 | 0.00E+00 |
| Median | **7.84E-01** | 7.53E-01 | 3.57E-01 | 3.01E-01 | 5.43E-01 | 7.03E-01 | 4.53E-01 | 0.00E+00 |
| Std | **2.05E-04** | 5.69E-09 | 2.10E-01 | 3.21E-01 | 2.47E-04 | 3.31E-02 | 4.47E-02 | 0.00E+00 |
| RWMOP16 | | | | | | | | |
| Best | 7.54E-01 | 0.00E+00 | 6.95E-01 | 7.56E-01 | 7.64E-01 | 7.63E-01 | **7.64E-01** | 5.75E-01 |
| Worst | 7.43E-01 | 0.00E+00 | 8.94E-02 | 7.46E-01 | 7.63E-01 | 7.61E-01 | **7.54E-01** | 0.00E+00 |
| Average | 7.49E-01 | 0.00E+00 | 4.46E-01 | 7.52E-01 | 7.64E-01 | 7.62E-01 | **7.61E-01** | 3.70E-01 |
| median | 7.48E-01 | 0.00E+00 | 4.75E-01 | 7.52E-01 | 7.64E-01 | 7.62E-01 | **7.62E-01** | 4.15E-01 |
| Std | 2.78E-03 | 0.00E+00 | 1.68E-01 | 2.63E-03 | 1.60E-04 | 3.57E-04 | **2.00E-03** | 1.50E-01 |
| RWMOP17 | | | | | | | | |
| Best | 5.48E-01 | 4.37E-01 | 3.37E+00 | 7.97E+07 | 2.73E-01 | **1.50E+12** | 2.88E-01 | 3.24E+00 |
| Worst | 1.20E-01 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 2.26E-01 | **5.48E-01** | 1.27E-01 | 0.00E+00 |
| Average | 3.92E-01 | 7.24E-02 | 5.38E-01 | 4.11E+06 | 2.64E-01 | **1.87E+11** | 2.07E-01 | 6.23E-01 |
| median | 4.61E-01 | 3.19E-04 | 4.21E-01 | 5.45E-01 | 2.68E-01 | **5.48E-01** | 2.09E-01 | 5.48E-01 |
| Std | 1.38E-01 | 1.20E-01 | 5.77E-01 | 1.62E+07 | 1.04E-02 | **3.50E+11** | 3.28E-02 | 5.20E-01 |
| RWMOP18 | | | | | | | | |
| Best | **4.14E-02** | 4.03E-02 | 3.30E-02 | 4.04E-02 | 4.05E-02 | 4.05E-02 | 3.99E-02 | 4.04E-02 |
| Worst | **4.12E-02** | 4.02E-02 | 2.36E-02 | 4.00E-02 | 4.05E-02 | 4.03E-02 | 3.68E-02 | 3.98E-02 |
| Average | **4.13E-02** | 4.03E-02 | 2.94E-02 | 4.03E-02 | 4.05E-02 | 4.04E-02 | 3.85E-02 | 4.02E-02 |
| median | **4.13E-02** | 4.03E-02 | 3.07E-02 | 4.03E-02 | 4.05E-02 | 4.04E-02 | 3.87E-02 | 4.02E-02 |
| Std | **5.47E-05** | 1.44E-05 | 3.00E-03 | 8.72E-05 | 5.06E-06 | 4.83E-05 | 8.83E-04 | 1.47E-04 |
| RWMOP19 | | | | | | | | |
| Best | **6.63E-01** | 6.63E-01 | 6.63E-01 | 6.63E-01 | 3.71E-01 | 6.63E-01 | 3.13E-01 | 6.63E-01 |
| Worst | **6.63E-01** | 6.63E-01 | 6.63E-01 | 5.89E-01 | 3.22E-01 | 6.63E-01 | 2.22E-01 | 6.63E-01 |
| Average | **6.63E-01** | 6.63E-01 | 6.63E-01 | 6.57E-01 | 3.43E-01 | 6.63E-01 | 2.60E-01 | 6.63E-01 |
| median | **6.63E-01** | 6.63E-01 | 6.63E-01 | 6.63E-01 | 3.45E-01 | 6.63E-01 | 2.59E-01 | 6.63E-01 |
| Std | **3.40E-15** | 0.00E+00 | 1.19E-06 | 1.59E-02 | 9.52E-03 | 0.00E+00 | 2.26E-02 | 0.00E+00 |
| RWMOP20 | | | | | | | | |
| Best | 1.74E-01 | **1.00E+00** | **1.00E+00** | **1.00E+00** | 0.00E+00 | **1.00E+00** | 0.00E+00 | **1.00E+00** |
| Worst | 1.71E-01 | **1.00E+00** | **1.00E+00** | **1.00E+00** | 0.00E+00 | **1.00E+00** | 0.00E+00 | **1.00E+00** |
| Average | 1.73E-01 | **1.00E+00** | **1.00E+00** | **1.00E+00** | 0.00E+00 | **1.00E+00** | 0.00E+00 | **1.00E+00** |
| median | 1.74E-01 | **1.00E+00** | **1.00E+00** | **1.00E+00** | 0.00E+00 | **1.00E+00** | 0.00E+00 | **1.00E+00** |
| Std | 4.41E-04 | **4.47E-15** | **3.28E-05** | **2.60E-05** | 0.00E+00 | **0.00E+00** | 0.00E+00 | **7.50E-07** |
| RWMOP21 | | | | | | | | |
| Best | 3.02E-02 | 2.93E-02 | 3.16E-02 | 3.17E-02 | 3.18E-02 | **3.18E-02** | 2.84E-02 | 3.17E-02 |
| Worst | 2.85E-02 | 2.93E-02 | 2.99E-02 | 2.04E-02 | 3.17E-02 | **3.18E-02** | 2.41E-02 | 3.15E-02 |
| Average | 2.95E-02 | 2.93E-02 | 3.10E-02 | 2.84E-02 | 3.18E-02 | **3.18E-02** | 2.48E-02 | 3.16E-02 |
| median | 2.95E-02 | 2.93E-02 | 3.12E-02 | 2.91E-02 | 3.18E-02 | **3.18E-02** | 2.47E-02 | 3.16E-02 |
| Std | 4.13E-04 | 2.94E-06 | 3.98E-04 | 2.35E-03 | 1.38E-06 | **8.07E-07** | 7.17E-04 | 5.90E-05 |

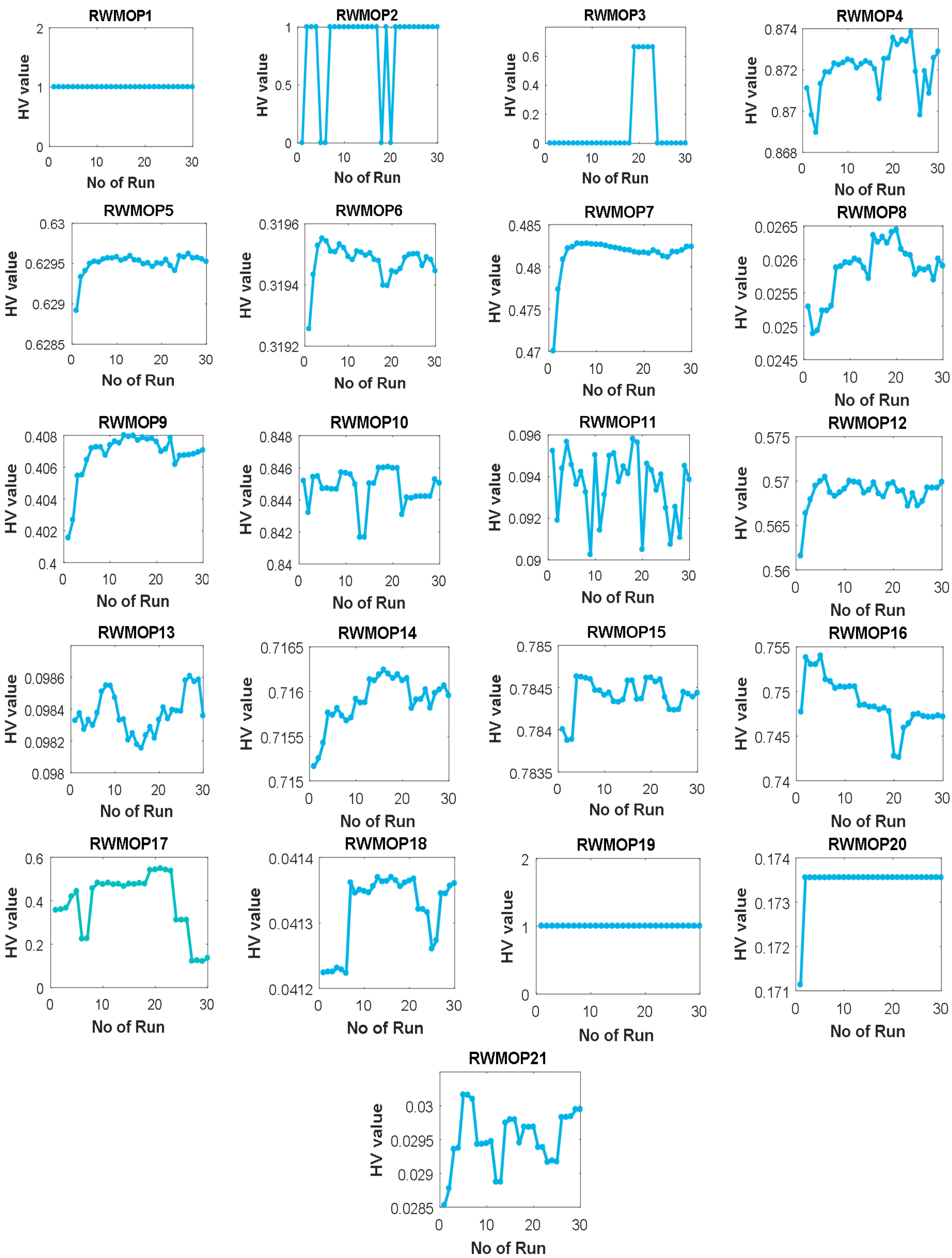


**Figure 8.** Mechanical design problems' HV value curves.

Table 9 displays the results for the best, worst, average, median, and std for MOGSK, MOEAD, eMOEA, MOPSO, NSGAII, SPEA2, KnEA, and GrEA of the HV metric, using chemical engineering problems (three problems in total), while Figure 9 shows the HV curve. For these test problems, MOGSK showed good behavior; in the RWMOP22 problem, the best results were given by KnEA, while all the remaining algorithms, including MOGSK, yielded the same results. A similar pattern was detected in the results of RWMOP 23, where the best results were given by MOGSK, MOEAD, eMOEA, MOPSO, SPEA2, and GrEA, while NSGAII and KnEA came in last. Similarly, in RWMOP24, the best results were yielded by NSGAII with a large gap, while the rest of the algorithms, including MOGSK, gave the same result. Overall, and using the chemical engineering problems, MOSGK gave good and constant results, as displayed in the HV curve in Figure 9.

**Table 9.** HV results using the chemical engineering problems.

| Algorithm Problem | MOGSK | MOEAD | eMOEA | MOPSO | NSGAII | SPEA2 | KnEA | GrEA |
|---|---|---|---|---|---|---|---|---|
| RWMOP22 | | | | | | | | |
| Best | 1.00E+00 | 1.00E+00 | 1.00E+00 | 1.00E+00 | 1.30E+00 | 1.00E+00 | **2.17E+00** | 1.00E+00 |
| Worst | 1.00E+00 | 1.00E+00 | 1.00E+00 | 1.00E+00 | 3.24E-01 | 1.00E+00 | **3.41E-01** | 1.00E+00 |
| Average | 1.00E+00 | 1.00E+00 | 1.00E+00 | 1.00E+00 | 7.15E-01 | 1.00E+00 | **7.85E-01** | 1.00E+00 |
| median | 1.00E+00 | 1.00E+00 | 1.00E+00 | 1.00E+00 | 7.07E-01 | 1.00E+00 | **7.34E-01** | 1.00E+00 |
| Std | 4.12E-17 | 0.00E+00 | 3.67E-06 | 0.00E+00 | 1.80E-01 | 0.00E+00 | **3.05E-01** | 0.00E+00 |
| RWMOP23 | | | | | | | | |
| Best | **9.99E-01** | **9.99E-01** | **9.99E-01** | **9.99E-01** | 7.19E-01 | **9.99E-01** | 8.25E-01 | **9.99E-01** |
| Worst | **9.99E-01** | **9.99E-01** | **9.86E-01** | **9.99E-01** | 9.73E-02 | **9.99E-01** | 9.09E-02 | **9.99E-01** |
| Average | **9.99E-01** | **9.99E-01** | **9.97E-01** | **9.99E-01** | 3.44E-01 | **9.99E-01** | 3.11E-01 | **9.99E-01** |
| median | **9.99E-01** | **9.99E-01** | **9.99E-01** | **9.99E-01** | 3.65E-01 | **9.99E-01** | 2.90E-01 | **9.99E-01** |
| Std | **5.46E-15** | **8.59E-15** | **3.42E-03** | **4.52E-16** | 1.58E-01 | **4.52E-16** | 1.78E-01 | **4.52E-16** |
| RWMOP24 | | | | | | | | |
| Best | 1.00E+00 | 1.00E+00 | 1.00E+00 | 1.00E+00 | **7.47E+05** | 1.00E+00 | 1.00E+00 | 1.00E+00 |
| Worst | 9.94E-01 | 1.00E+00 | 9.96E-01 | 1.00E+00 | **0.00E+00** | 1.00E+00 | 0.00E+00 | 1.00E+00 |
| Average | 9.99E-01 | 1.00E+00 | 1.00E+00 | 1.00E+00 | **2.49E+04** | 1.00E+00 | 3.00E-01 | 1.00E+00 |
| median | 1.00E+00 | 1.00E+00 | 1.00E+00 | 1.00E+00 | **0.00E+00** | 1.00E+00 | 0.00E+00 | 1.00E+00 |
| Std | 1.73E-03 | 2.93E-09 | 7.36E-04 | 0.00E+00 | **1.36E+05** | 0.00E+00 | 4.66E-01 | 0.00E+00 |

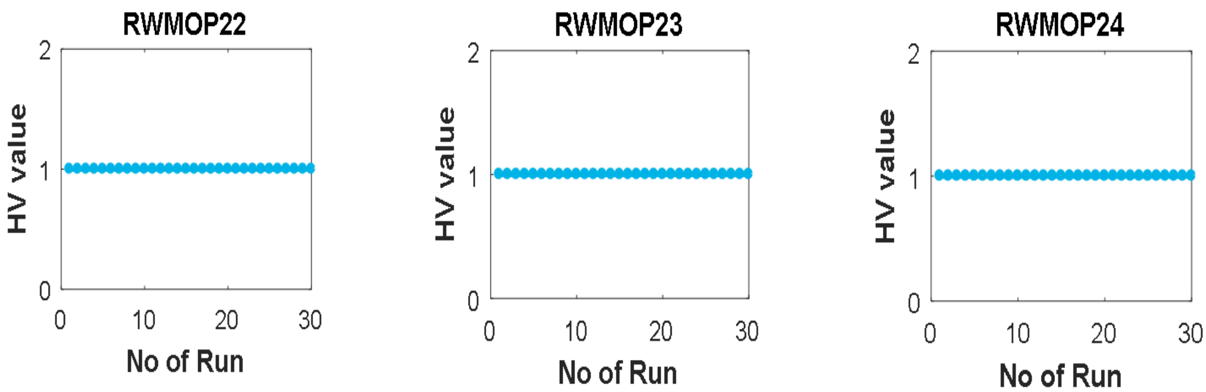


**Figure 9.** Chemical engineering problems' HV value curves.

Table 10 displays the results for the best, worst, average, median, and std for MOGSK, MOEAD, eMOEA, MOPSO, NSGAII, SPEA2, KnEA, and GrEA of the HV metric, using process, design, and synthesis problems (five problems in total), while Figure 10 shows the HV curve. In this test problem series, MOGSK showed a good performance, where it was capable of topping the comparative algorithms' performance in four out of the five test problems, namely RWMOP25, RWMOP26, RWMOP28, and RWMOP29; these results are supported by the HV curve in Figure 10, where steady values of HV were recorded. For RWMOP27, the best results were yielded by eMOEA with a considerable gap, followed by KnEA, NSGAII, SPEA2, MOPSO, MOEAD, GrEA, and then MOGSK.

**Table 10.** HV results using the process, design, and synthesis problems.

| Algorithm Problem | MOGSK | MOEAD | eMOEA | MOPSO | NSGAII | SPEA2 | KnEA | GrEA |
|---|---|---|---|---|---|---|---|---|
| RWMOP25 | | | | | | | | |
| Best | **1.00E+00** | 4.02E-01 | 2.33E-01 | 2.34E-01 | 2.41E-01 | 2.36E-01 | 2.41E-01 | 2.31E-01 |
| Worst | **9.99E-01** | 2.28E-01 | 2.26E-01 | 2.24E-01 | 2.41E-01 | 2.32E-01 | 2.40E-01 | 2.31E-01 |
| Average | **9.99E-01** | 2.56E-01 | 2.29E-01 | 2.29E-01 | 2.41E-01 | 2.35E-01 | 2.41E-01 | 2.31E-01 |
| median | **1.00E+00** | 2.33E-01 | 2.30E-01 | 2.29E-01 | 2.41E-01 | 2.35E-01 | 2.41E-01 | 2.31E-01 |
| Std | **4.04E-04** | 5.83E-02 | 1.68E-03 | 2.39E-03 | 8.76E-05 | 1.02E-03 | 2.96E-04 | 1.67E-06 |
| RWMOP26 | | | | | | | | |
| Best | **8.49E-01** | 8.45E-01 | 6.98E-01 | 6.59E-01 | 2.00E-01 | 7.28E-01 | 2.05E-01 | 5.65E-01 |
| Worst | **8.21E-01** | 8.45E-01 | 0.00E+00 | 0.00E+00 | 1.22E-01 | 0.00E+00 | 9.10E-02 | 0.00E+00 |
| Average | **8.39E-01** | 8.45E-01 | 1.53E-01 | 2.29E-01 | 1.54E-01 | 2.78E-01 | 1.51E-01 | 1.51E-01 |
| median | **8.40E-01** | 8.45E-01 | 7.54E-02 | 2.41E-01 | 1.49E-01 | 2.90E-01 | 1.50E-01 | 0.00E+00 |
| Std | **6.59E-03** | 9.76E-08 | 2.03E-01 | 1.85E-01 | 2.29E-02 | 2.57E-01 | 4.21E-02 | 2.54E-01 |
| RWMOP27 | | | | | | | | |
| Best | 1.90E+00 | 1.93E+02 | **9.33E+13** | 1.19E+04 | 7.88E+11 | 7.62E+10 | 8.97E+12 | 1.22E+02 |
| Worst | 1.43E+00 | 1.00E+00 | **1.00E+00** | 1.00E+00 | 1.99E+08 | 1.00E+00 | 6.02E+07 | 1.00E+00 |
| Average | 1.50E+00 | 2.36E+01 | **4.48E+12** | 4.59E+02 | 1.00E+11 | 3.55E+09 | 3.17E+11 | 3.27E+01 |
| median | 1.46E+00 | 1.86E+00 | **1.00E+00** | 1.45E+00 | 5.07E+09 | 8.96E+02 | 1.44E+09 | 1.12E+01 |
| Std | 1.05E-01 | 4.38E+01 | **1.83E+13** | 2.17E+03 | 2.05E+11 | 1.41E+10 | 1.64E+12 | 3.92E+01 |
| RWMOP28 | | | | | | | | |
| Best | **1.00E+00** | 1.00E+00 | 1.00E+00 | 1.00E+00 | 5.01E-02 | 1.00E+00 | 4.99E-02 | 1.00E+00 |
| Worst | **1.00E+00** | 1.00E+00 | 9.96E-01 | 0.00E+00 | 0.00E+00 | 1.00E+00 | 0.00E+00 | 1.00E+00 |
| Average | **1.00E+00** | 1.00E+00 | 9.99E-01 | 9.56E-01 | 4.70E-03 | 1.00E+00 | 2.59E-03 | 1.00E+00 |
| median | **1.00E+00** | 1.00E+00 | 1.00E+00 | 1.00E+00 | 0.00E+00 | 1.00E+00 | 0.00E+00 | 1.00E+00 |
| Std | **1.75E-15** | 2.06E-17 | 9.68E-04 | 1.82E-01 | 1.20E-02 | 0.00E+00 | 9.57E-03 | 0.00E+00 |
| RWMOP29 | | | | | | | | |
| Best | **1.00E+00** | 1.00E+00 | **1.00E+00** | 1.00E+00 | 7.87E-01 | 1.00E+00 | 7.56E-01 | 1.00E+00 |
| Worst | **9.93E-01** | 1.00E+00 | **1.00E+00** | 0.00E+00 | 5.59E-01 | 1.00E+00 | 7.89E-02 | 1.00E+00 |
| Average | **9.97E-01** | 1.00E+00 | **1.00E+00** | 9.37E-01 | 7.68E-01 | 1.00E+00 | 6.29E-01 | 1.00E+00 |
| median | **9.98E-01** | 1.00E+00 | **1.00E+00** | 1.00E+00 | 7.84E-01 | 1.00E+00 | 7.06E-01 | 1.00E+00 |
| Std | **1.75E-03** | 6.25E-07 | **1.82E-10** | 1.93E-01 | 4.26E-02 | 1.15E-07 | 1.78E-01 | 2.51E-07 |

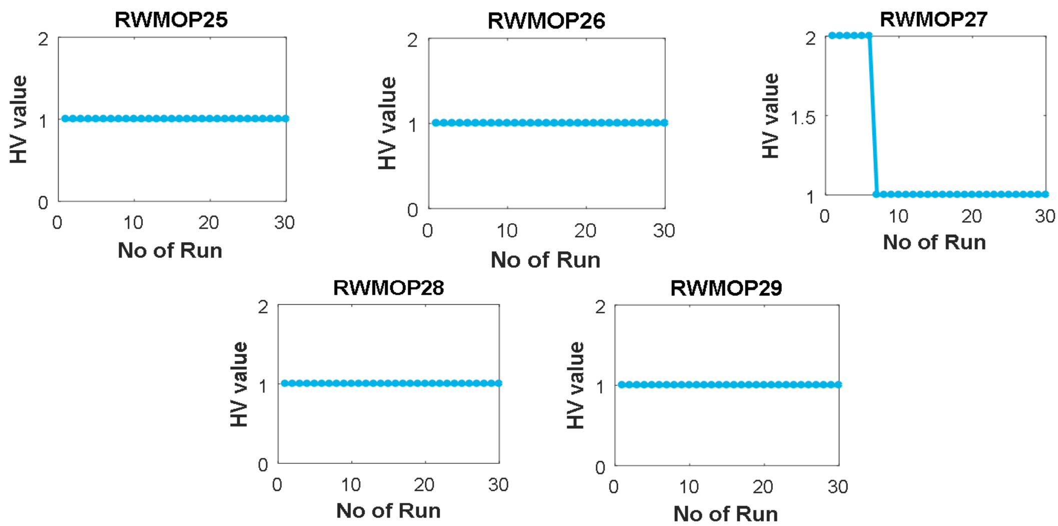


**Figure 10.** Process, design, and synthesis problems' HV value curves.

Table 11 displays the results for the best, worst, average, median, and std for MOGSK, MOEAD, eMOEA, MOPSO, NSGAII, SPEA2, KnEA, and GrEA of the HV metric, using the power electronics problems (six problems in total), while Figure 11 shows the HV curve for the respective test problems. For RWMOP30, the best result was reported for GrEA, followed by MOEAD, SPEA2, eMOEA, NSGAII, and KnEA, then MOGSK and MOPSO. For RWMOP31, the best result was given by MOEAD, followed by GrEA, eMOEA, SPEA2, and MOPSO, then MOGSK in fifth place, followed by KnEA and NSGAII. From RWMOP31 to RWMOP36, the best results were reported to MOEAD, while MOGSK overall was able to only outperform three or four comparative algorithms with close-ranged results. Supporting these findings, Figure 11 shows the HV curve for this problem series.

**Table 11.** HV results using power electronics problems.

| Algorithm Problem | MOGSK | MOEAD | eMOEA | MOPSO | NSGAII | SPEA2 | KnEA | GrEA |
|---|---|---|---|---|---|---|---|---|
| RWMOP30 | | | | | | | | |
| Best | 5.41E-01 | 7.81E-01 | 7.36E-01 | 4.78E-01 | 6.82E-01 | 7.43E-01 | 6.54E-01 | **8.00E-01** |
| Worst | 2.12E-01 | 2.83E-01 | 3.31E-01 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | **1.08E-01** |
| Average | 4.55E-01 | 6.99E-01 | 6.15E-01 | 2.21E-01 | 2.48E-01 | 4.53E-01 | 2.62E-01 | **6.43E-01** |
| median | 4.90E-01 | 7.19E-01 | 6.43E-01 | 2.49E-01 | 0.00E+00 | 5.51E-01 | 2.59E-01 | **6.65E-01** |
| Std | 9.89E-02 | 9.94E-02 | 9.80E-02 | 1.59E-01 | 2.77E-01 | 2.46E-01 | 2.66E-01 | **1.30E-01** |
| RWMOP31 | | | | | | | | |
| Best | 7.96E-01 | **9.18E-01** | 9.04E-01 | 8.69E-01 | 7.40E-01 | 8.94E-01 | 7.80E-01 | 9.08E-01 |
| Worst | 7.16E-01 | **3.13E-01** | 4.14E-01 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 7.63E-01 |
| Average | 7.58E-01 | **8.32E-01** | 8.32E-01 | 6.81E-01 | 1.33E-01 | 7.83E-01 | 2.22E-01 | 8.65E-01 |
| median | 7.59E-01 | **8.78E-01** | 8.58E-01 | 7.08E-01 | 0.00E+00 | 8.42E-01 | 5.27E-02 | 8.70E-01 |
| Std | 2.12E-02 | **1.27E-01** | 1.12E-01 | 1.73E-01 | 2.39E-01 | 1.84E-01 | 2.92E-01 | 3.29E-02 |
| RWMOP32 | | | | | | | | |
| Best | 7.26E-01 | **9.14E-01** | 8.95E-01 | 7.89E-01 | 8.45E-01 | 8.54E-01 | 7.89E-01 | 8.86E-01 |
| Worst | 6.40E-01 | **3.96E-01** | 5.12E-01 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 4.65E-01 |
| Average | 7.06E-01 | **8.50E-01** | 8.27E-01 | 5.28E-01 | 3.89E-01 | 7.05E-01 | 1.81E-01 | 8.12E-01 |
| median | 7.15E-01 | **8.73E-01** | 8.40E-01 | 5.79E-01 | 5.14E-01 | 7.65E-01 | 0.00E+00 | 8.47E-01 |
| Std | 2.26E-02 | **9.03E-02** | 6.58E-02 | 2.24E-01 | 3.56E-01 | 2.06E-01 | 3.12E-01 | 9.07E-02 |
| RWMOP33 | | | | | | | | |
| Best | 6.46E-01 | **9.12E-01** | 8.60E-01 | 7.39E-01 | 0.00E+00 | 8.62E-01 | 0.00E+00 | 8.74E-01 |
| Worst | 4.94E-01 | **6.33E-01** | 4.30E-01 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 6.82E-01 |
| Average | 6.03E-01 | **8.48E-01** | 8.02E-01 | 4.48E-01 | 0.00E+00 | 6.02E-01 | 0.00E+00 | 8.12E-01 |
| median | 6.21E-01 | **8.72E-01** | 8.23E-01 | 5.27E-01 | 0.00E+00 | 7.47E-01 | 0.00E+00 | 8.36E-01 |
| Std | 4.01E-02 | **5.80E-02** | 7.64E-02 | 2.43E-01 | 0.00E+00 | 3.03E-01 | 0.00E+00 | 5.26E-02 |
| RWMOP34 | | | | | | | | |
| Best | 7.52E-01 | **9.13E-01** | 8.82E-01 | 8.07E-01 | 0.00E+00 | 8.69E-01 | 0.00E+00 | 8.92E-01 |
| Worst | 6.08E-01 | **4.62E-01** | 4.36E-01 | 2.23E-01 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 7.49E-01 |
| Average | 7.13E-01 | **8.47E-01** | 8.25E-01 | 5.65E-01 | 0.00E+00 | 7.08E-01 | 0.00E+00 | 8.36E-01 |
| median | 7.47E-01 | **8.77E-01** | 8.42E-01 | 5.48E-01 | 0.00E+00 | 8.07E-01 | 0.00E+00 | 8.44E-01 |
| Std | 4.69E-02 | **8.92E-02** | 7.68E-02 | 1.42E-01 | 0.00E+00 | 2.35E-01 | 0.00E+00 | 4.23E-02 |
| RWMOP35 | | | | | | | | |
| Best | 9.18E-01 | **9.76E-01** | 9.67E-01 | 9.33E-01 | 7.19E-01 | 9.61E-01 | 6.52E-01 | 9.71E-01 |
| Worst | 8.99E-01 | **9.51E-01** | 9.32E-01 | 5.14E-01 | 0.00E+00 | 2.06E-01 | 0.00E+00 | 8.73E-01 |
| Average | 9.17E-01 | **9.66E-01** | 9.51E-01 | 8.64E-01 | 1.62E-01 | 8.87E-01 | 1.91E-01 | 9.49E-01 |
| median | 9.18E-01 | **9.66E-01** | 9.52E-01 | 8.89E-01 | 5.75E-02 | 9.39E-01 | 1.13E-01 | 9.52E-01 |
| Std | 3.95E-03 | **5.45E-03** | 9.34E-03 | 7.97E-02 | 2.26E-01 | 1.60E-01 | 2.24E-01 | 1.75E-02 |

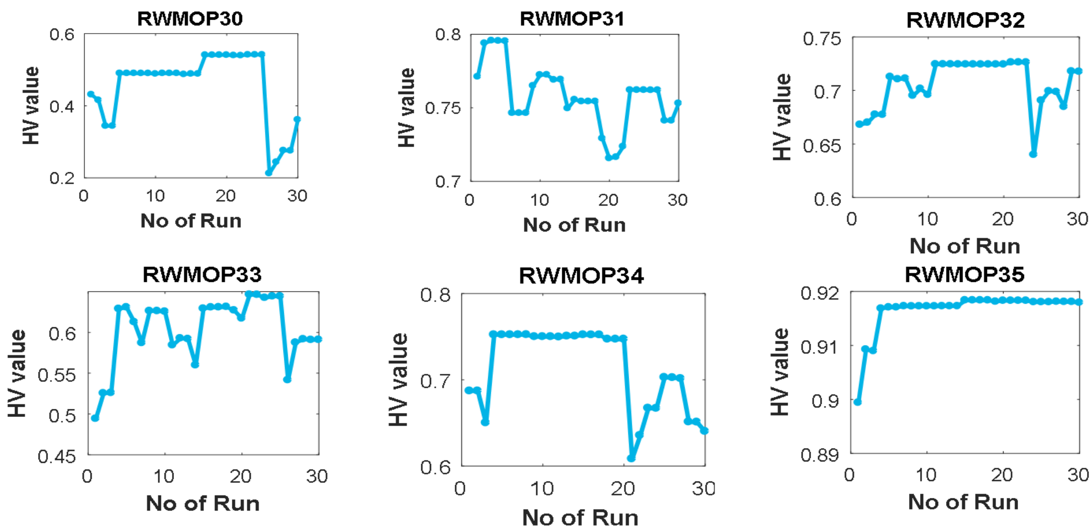


**Figure 11.** Power electronics problems' HV value curves.

Table 12 displays the results for the best, worst, average, median, and std for MOGSK, MOEAD, eMOEA, MOPSO, NSGAII, SPEA2, KnEA, and GrEA of the HV metric, using the power system optimization problems (fifteen problems in total), while Figure 12 shows the HV curve for the respective test problems. For the problems RWMOP36 to RWMOP39, all the algorithms, including MOGSK, gave the same results. RWMOP40's best result belonged to MOPSO, followed by MOGSK, outperforming the other comparative algorithms. The best result for RWMOP41 belonged to MOPSO and MOGSK, with close-ranged results, overcoming the remaining comparative algorithms. As for RWMOP42, the top results belonged to MOEAD, followed by GrEA and then eMOEA and SPEA2, with close results, while MOGSK, MOPSO, NSGAII, and KnEA came in third place, with the same results. The obtained results for RWMOP43 showed that the best result was yielded by MOEAD, SPEA2, and GrEA, with the same results, followed by MOGSK, with close results as well. For RWMOP44 and RWMOP45, MOGSK yielded the best results, surpassing all the comparative algorithms. The RWMOP46 problem's results showed the same pattern, while the best results with small margin were given by MOEAD and MOGSK taking over the remaining comparative algorithms. RWMOP47's best results were obtained by MOEAD, eMOEA, MOPSO, SPEA2 GrEA, and MOGSK, and lastly came NSGAII and KnEA. The same pattern was detected in RWMOP48. RWMOP49's obtained results show that the best was obtained by SPEA2 and MOGSK, outperforming the remaining algorithms. Lastly, the best result was shared by MOGSK, eMOEA, MOEAD, MOPSO, SPEA2, and GrEA, followed in second place by NSGAII and KnEA. Figure 12 supports the finding; overall, MOGSK performed well in this test problem series, where either it yielded the best results or gave close ones. Lastly, and by observing the results of both experiments and specially the second experiment we can confirm that MOGSK can be a good tool for optimization, as it was able to give good results with real-world problems, which reinforce the stance of MOGSK in the midst of optimization algorithms, and more specifically, evolutionary-based algorithms.

**Table 12.** HV results using the power system optimization problems.

| **Algorithm Problem** | **MOGSK** | **MOEAD** | **eMOEA** | **MOPSO** | **NSGAII** | **SPEA2** | **KnEA** | **GrEA** |
|---|---|---|---|---|---|---|---|---|
| RWMOP36 | | | | | | | | |
| Best | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| Worst | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| Average | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |

**Table 12.** *Cont.*

| Algorithm Problem | MOGSK | MOEAD | eMOEA | MOPSO | NSGAII | SPEA2 | KnEA | GrEA |
|---|---|---|---|---|---|---|---|---|
| median | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| Std | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| RWMOP37 | | | | | | | | |
| Best | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| Worst | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| Average | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| median | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| Std | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| RWMOP38 | | | | | | | | |
| Best | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| Worst | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| Average | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| median | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| Std | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| RWMOP39 | | | | | | | | |
| Best | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| Worst | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| Average | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| median | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| Std | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| RWMOP40 | | | | | | | | |
| Best | 6.17E+01 | 8.35E-01 | 9.18E-01 | **8.63E+00** | 5.12E+00 | 2.02E+00 | 4.87E+00 | 8.26E-01 |
| Worst | 3.14E-01 | 7.55E-01 | 8.08E-02 | **0.00E+00** | 0.00E+00 | 2.34E-01 | 0.00E+00 | 5.19E-01 |
| Average | 1.54E+01 | 7.91E-01 | 6.27E-01 | **1.09E+00** | 1.25E+00 | 8.10E-01 | 1.33E+00 | 7.10E-01 |
| median | 9.74E+00 | 7.93E-01 | 6.53E-01 | **3.96E-01** | 6.41E-02 | 8.10E-01 | 8.24E-02 | 7.08E-01 |
| Std | 1.69E+01 | 1.98E-02 | 2.44E-01 | **2.29E+00** | 1.82E+00 | 3.01E-01 | 1.66E+00 | 6.98E-02 |
| RWMOP41 | | | | | | | | |
| Best | 9.89E-01 | 2.99E-01 | 4.89E+01 | **9.93E+00** | 0.00E+00 | 2.36E+01 | 0.00E+00 | 3.20E+01 |
| Worst | 5.65E-02 | 0.00E+00 | 1.97E-01 | **0.00E+00** | 0.00E+00 | 0.00E+00 | 0.00E+00 | 4.80E-01 |
| Average | 8.42E-01 | 1.07E-01 | 6.51E+00 | **3.41E-01** | 0.00E+00 | 1.66E+00 | 0.00E+00 | 3.23E+00 |
| median | 9.55E-01 | 1.18E-01 | 8.62E-01 | **0.00E+00** | 0.00E+00 | 4.83E-01 | 0.00E+00 | 8.27E-01 |
| Std | 2.26E-01 | 9.03E-02 | 1.35E+01 | **1.81E+00** | 0.00E+00 | 5.01E+00 | 0.00E+00 | 7.71E+00 |
| RWMOP42 | | | | | | | | |
| Best | 0.00E+00 | **9.83E-01** | 6.98E-01 | 0.00E+00 | 0.00E+00 | 6.93E-01 | 0.00E+00 | 9.21E-01 |
| Worst | 0.00E+00 | **8.56E-01** | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| Average | 0.00E+00 | **9.39E-01** | 3.80E-02 | 0.00E+00 | 0.00E+00 | 2.31E-02 | 0.00E+00 | 1.98E-01 |
| median | 0.00E+00 | **9.49E-01** | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| Std | 0.00E+00 | **3.39E-02** | 1.48E-01 | 0.00E+00 | 0.00E+00 | 1.27E-01 | 0.00E+00 | 3.18E-01 |
| RWMOP43 | | | | | | | | |
| Best | 9.93E-01 | **1.00E+00** | 9.99E-01 | 0.00E+00 | 0.00E+00 | 1.00E+00 | 0.00E+00 | 1.00E+00 |
| Worst | 4.63E-01 | **1.00E+00** | 9.85E-01 | 0.00E+00 | 0.00E+00 | 9.97E-01 | 0.00E+00 | 9.99E-01 |
| Average | 8.67E-01 | **1.00E+00** | 9.95E-01 | 0.00E+00 | 0.00E+00 | 1.00E+00 | 0.00E+00 | 1.00E+00 |
| Median | 8.95E-01 | **1.00E+00** | 9.96E-01 | 0.00E+00 | 0.00E+00 | 1.00E+00 | 0.00E+00 | 1.00E+00 |
| Std | 1.22E-01 | **9.19E-05** | 3.26E-03 | 0.00E+00 | 0.00E+00 | 5.10E-04 | 0.00E+00 | 1.68E-04 |
| RWMOP44 | | | | | | | | |
| Best | **1.00E+00** | 9.98E-01 | 9.67E-01 | 0.00E+00 | 0.00E+00 | 9.06E-01 | 0.00E+00 | 9.40E-01 |
| Worst | **0.00E+00** | 9.29E-01 | 2.90E-01 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| Average | **9.17E-01** | 9.79E-01 | 6.66E-01 | 0.00E+00 | 0.00E+00 | 4.73E-01 | 0.00E+00 | 5.05E-01 |
| median | **1.00E+00** | 9.81E-01 | 6.55E-01 | 0.00E+00 | 0.00E+00 | 5.14E-01 | 0.00E+00 | 5.66E-01 |
| Std | **2.36E-01** | 1.52E-02 | 1.42E-01 | 0.00E+00 | 0.00E+00 | 2.69E-01 | 0.00E+00 | 2.68E-01 |
| RWMOP45 | | | | | | | | |
| Best | **1.00E+00** | 1.00E+00 | 9.99E-01 | 0.00E+00 | 0.00E+00 | 9.99E-01 | 0.00E+00 | 1.00E+00 |

**Table 12.** *Cont.*

| Algorithm Problem | MOGSK | MOEAD | eMOEA | MOPSO | NSGAII | SPEA2 | KnEA | GrEA |
|---|---|---|---|---|---|---|---|---|
| Worst | **1.15E-01** | 1.00E+00 | 9.30E-01 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 8.92E-01 |
| Average | **9.03E-01** | 1.00E+00 | 9.81E-01 | 0.00E+00 | 0.00E+00 | 8.24E-01 | 0.00E+00 | 9.84E-01 |
| median | **9.80E-01** | 1.00E+00 | 9.87E-01 | 0.00E+00 | 0.00E+00 | 9.44E-01 | 0.00E+00 | 9.96E-01 |
| Std | 1.98E-01 | 1.77E-05 | 1.73E-02 | 0.00E+00 | 0.00E+00 | 2.94E-01 | 0.00E+00 | 2.71E-02 |
| RWMOP46 | | | | | | | | |
| Best | 9.99E-01 | **1.00E+00** | 9.88E-01 | 0.00E+00 | 0.00E+00 | 6.79E-01 | 0.00E+00 | 1.00E+00 |
| Worst | 1.13E-01 | **7.71E-01** | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 |
| Average | 8.75E-01 | **9.76E-01** | 5.72E-01 | 0.00E+00 | 0.00E+00 | 1.91E-01 | 0.00E+00 | 6.21E-01 |
| median | 9.97E-01 | **9.93E-01** | 6.69E-01 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 7.51E-01 |
| Std | 2.54E-01 | **4.80E-02** | 3.16E-01 | 0.00E+00 | 0.00E+00 | 2.45E-01 | 0.00E+00 | 3.37E-01 |
| RWMOP47 | | | | | | | | |
| Best | 9.01E-01 | **1.00E+00** | **1.00E+00** | **1.00E+00** | 0.00E+00 | **1.00E+00** | 0.00E+00 | **1.00E+00** |
| Worst | 1.72E-01 | **1.00E+00** | **1.00E+00** | **0.00E+00** | 0.00E+00 | **1.00E+00** | 0.00E+00 | **1.00E+00** |
| Average | 2.05E-01 | **1.00E+00** | **1.00E+00** | **9.67E-01** | 0.00E+00 | **1.00E+00** | 0.00E+00 | **1.00E+00** |
| median | 1.74E-01 | **1.00E+00** | **1.00E+00** | **1.00E+00** | 0.00E+00 | **1.00E+00** | 0.00E+00 | **1.00E+00** |
| Std | 1.33E-01 | **0.00E+00** | **0.00E+00** | **1.83E-01** | 0.00E+00 | **0.00E+00** | 0.00E+00 | **0.00E+00** |
| RWMOP48 | | | | | | | | |
| Best | 9.87E-01 | **1.00E+00** | 1.00E+00 | 9.87E-01 | 0.00E+00 | **1.00E+00** | 0.00E+00 | **1.00E+00** |
| Worst | 7.31E-01 | **1.00E+00** | 7.75E-01 | 0.00E+00 | 0.00E+00 | **1.00E+00** | 0.00E+00 | **1.00E+00** |
| Average | 9.27E-01 | **1.00E+00** | 9.92E-01 | 2.63E-01 | 0.00E+00 | **1.00E+00** | 0.00E+00 | **1.00E+00** |
| median | 9.43E-01 | **1.00E+00** | 1.00E+00 | 0.00E+00 | 0.00E+00 | **1.00E+00** | 0.00E+00 | **1.00E+00** |
| Std | 6.19E-02 | **3.46E-05** | 4.12E-02 | 3.80E-01 | 0.00E+00 | **7.51E-06** | 0.00E+00 | **2.05E-05** |
| RWMOP49 | | | | | | | | |
| Best | 1.52E-01 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | **7.15E-01** | 0.00E+00 | 0.00E+00 |
| Worst | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | **0.00E+00** | 0.00E+00 | 0.00E+00 |
| Average | 9.55E-02 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | **2.38E-02** | 0.00E+00 | 0.00E+00 |
| Median | 9.99E-02 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | **0.00E+00** | 0.00E+00 | 0.00E+00 |
| Std | 3.34E-02 | 0.00E+00 | 0.00E+00 | 0.00E+00 | 0.00E+00 | **1.31E-01** | 0.00E+00 | 0.00E+00 |
| RWMOP50 | | | | | | | | |
| Best | **6.11E-01** | **6.11E-01** | **6.11E-01** | **6.11E-01** | 1.38E-02 | **6.11E-01** | 1.57E-02 | **6.11E-01** |
| Worst | **6.11E-01** | **6.11E-01** | **6.11E-01** | **6.07E-01** | 9.59E-03 | **6.07E-01** | 5.75E-03 | **6.08E-01** |
| Average | **6.11E-01** | **6.11E-01** | **6.11E-01** | **6.11E-01** | 1.18E-02 | **6.09E-01** | 1.18E-02 | **6.09E-01** |
| Median | **6.11E-01** | **6.11E-01** | **6.11E-01** | **6.11E-01** | 1.18E-02 | **6.09E-01** | 1.22E-02 | **6.09E-01** |
| Std | **5.47E-07** | **1.70E-10** | **1.13E-16** | **7.53E-04** | 9.39E-04 | **9.61E-04** | 1.64E-03 | **6.27E-04** |

Limitation

In this work, the proposed MOGSK was tested using two different benchmarks: the first one is the ZDT, DTLZ series test functions; the second one is the CEC 2021 (real-world constrained multiobjective optimization problems). MOGSK performed well in most of the test problems; however, as with any optimization problem, it was not able to yield good results in all of them. As can be seen in the ZDT test series using ZDT4, MOGSK was stuck in a local optimum known as the Pareto front of the ZDT4, which is a concave region, which makes it difficult for algorithms to explore and converge to the real global optima. Also, due to the deceptive nature of ZDT4, it makes a local optimum more fitting. In addition, for ZDT3, even though MOGSK was able to converge, it was not able to cover all the front, which is a premature convergence, which is one of the challenges of ZDT3 (discontinuous front); this feature of ZDT3 requires an optimization algorithm to have a delicate balance between exploration and exploitation in order to conduct extensive exploration to discover these regions, while also exploiting known solutions to improve convergence. However, the results of the HV metrics show that the algorithm is good compared to the comparative algorithms, which leaves room for improvement. While for the real-world CEC 2021 test problems, and with different problems and variations, we can tell in general that MOGSK did great, in cases where the algorithm did not give the best results, it was able to yield good results, such as the case of chemical engineering problems. MOGSK performed well with mechanical design problems, where it was able to give the best results for eleven test problems out of twenty-one test problems. Similarly, for process, design, and synthesis problems, MOGSK gave 90% of the best results in comparison with other algorithms. However, power electronics problems and power system optimization problems were quite challenging for MOGSK. Power electronics include conversion, control, and conditioning; these problems' difficulty is rather high due to different factors, one of which is nonlinearity, which causes the optimization problem to be highly nonconvex. Power system optimization, on the other hand, involves the generation, transmission, distribution, and utilization of electrical energy; this problem challenge lies in the high number of equality constraints. Therefore, for these two test series, MOGSK did not show good results. All in all, MOGSK uses a set of parameters, and this set can greatly affect the results and how well the algorithm can operate. One of the solutions that we proposed to further improve the results is adaptive parameters, which have already been proposed in a single-optimization version, but not yet in multiobjective optimization.

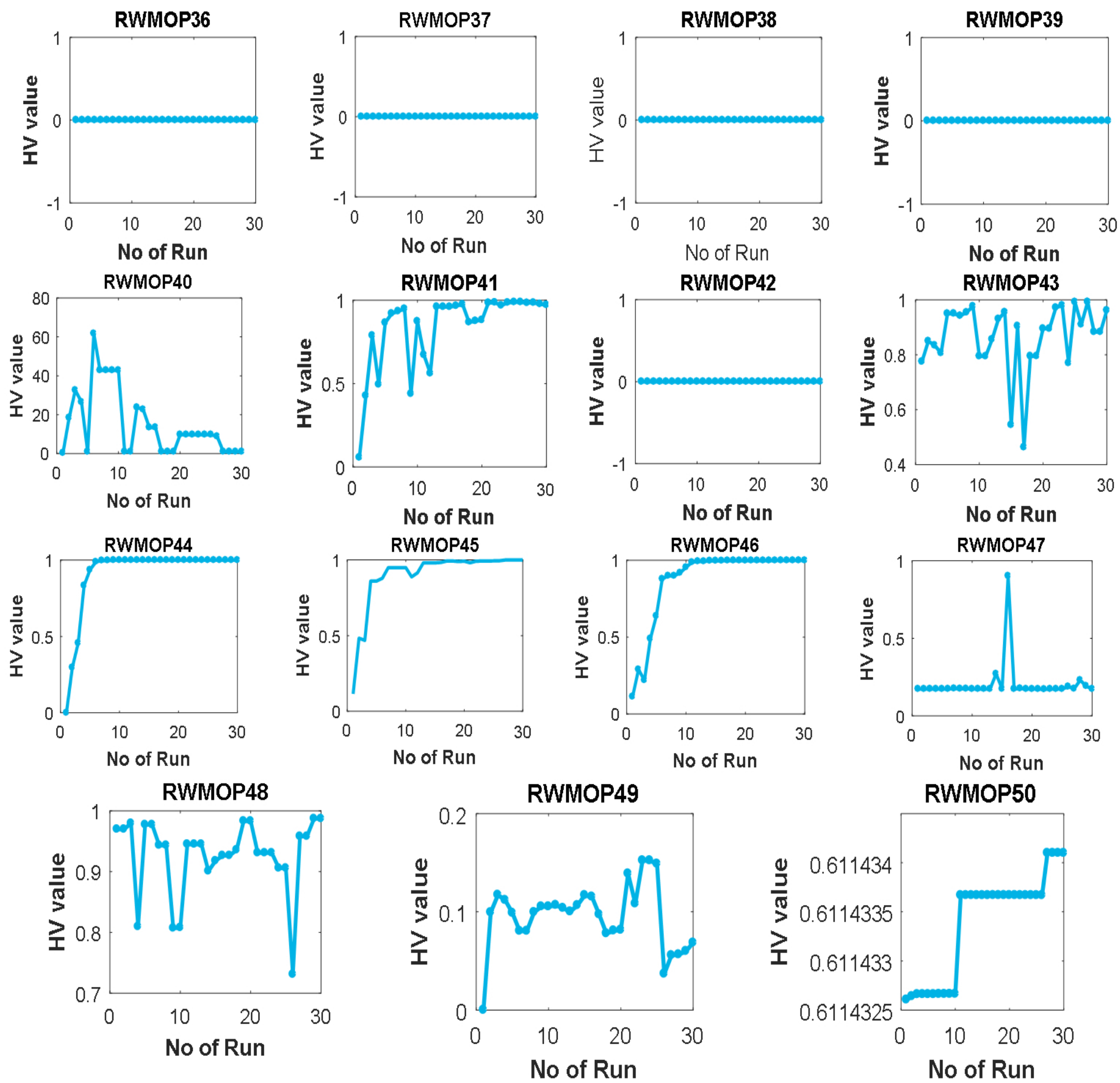


**Figure 12.** Power system optimization problems HV value curves.

## 5. Summary and Future Work

This study presented the initial extended version of the recently introduced gaining–sharing knowledge optimization to solve multiobjective optimization issues, named MOGSK. And in order to ensure the passage from a single-objective optimization algorithm towards a multiobjective one, several strategies were adapted. Firstly, the fast nondominated solution, also known as (FNS), and crowding distance (CD) techniques were employed to obtain the nondominated solution and to preserve the distribution and diversity of the solution along the exploitation process. Secondly, an external archive was used to safeguard the best solutions found so far, and to help in the update process by guiding the solutions around the Pareto optimal. Lastly, the archive solutions were updated using the epsilon dominance relation, which helps boost convergence in the direction of the Pareto optimal front. Our proposed MOGSK was evaluated using the biobjective test functions (ZDT), which include five problems and the three-objective test functions (DTLZ), including seven problems. In addition, the CEC 2021 (RWMOPs) problems were also used; this collection accommodates a variety of problems, such as chemical engineering problems, power electronics problems, mechanical design problems, and power system optimization problems, with a total of fifty problems. The MOGSK results were compared with known algorithms, including MOEAD, eMOEA, MOPSO, NSGAII, SPEA2, KnEA, and GrEA. The obtained results proved that MOGSK is a good tool of optimization. The aim for future work consists of improving the proposed algorithm so that it can solve more optimization problems, and exploring the propensity of the proposed algorithm in resolving real-world issues.

**Author Contributions:** Conceptualization, N.E.C. and A.A.; Methodology, N.E.C.; Validation, H.M.; Formal analysis, H.M.; Data curation, K.A.A. and H.M.Z.; Writing—original draft, N.E.C., A.A., K.A.A., H.M.Z., H.M. and A.W.M.; Writing—review & editing, N.E.C., A.A., K.A.A., H.M.Z., H.M. and A.W.M.; Supervision, A.W.M.; Funding acquisition, K.A.A. and A.W.M. All authors have read and agreed to the published version of the manuscript.

**Funding:** The research is funded by the Researchers Supporting Program at King Saud University, (RSP2023R305).

**Data Availability Statement:** Not applicable.

**Acknowledgments:** The authors present their appreciation to King Saud University for funding the publication of this research through the Researchers Supporting Program (RSP2023R305), King Saud University, Riyadh, Saudi Arabia.

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

## References

1. Gunantara, N. A review of multi-objective optimization: Methods and its applications. *Cogent Eng.* **2018**, *5*, 1502242. [CrossRef]
2. Mirjalili, S.; Jangir, P.; Saremi, S. Multi-objective ant lion optimizer: A multi-objective optimization algorithm for solving engineering problems. *Appl. Intell.* **2017**, *46*, 79–95. [CrossRef]
3. Branke, J.; Deb, K.; Dierolf, H.; Osswald, M. Finding knees in multi-objective optimization. In Proceedings of the International Conference on Parallel Problem Solving from Nature, Birmingham, UK, 18–22 September 2004; pp. 722–731.
4. Veldhuizen, D.; Lamont, G. Multiobjective evolutionary algorithms: Analyzing the state-of-the-art. *Evol. Comput.* **2000**, *8*, 125–147. [CrossRef] [PubMed]
5. Kim, I.; Weck, O. Adaptive weighted-sum method for bi-objective optimization: Pareto front generation. *Struct. Multidiscip. Optim.* **2005**, *29*, 149–158. [CrossRef]
6. Nedjah, N.; Mourelle, L.M. Evolutionary multi–objective optimisation: A survey. *Int. J. Bio-Inspired Comput.* **2015**, *7*, 1–25. [CrossRef]
7. Holland, J. Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence. Available online: https://ieeexplore.ieee.org/book/6267401 (accessed on 15 June 2023).
8. Srinivas, N.; Deb, K. Muiltiobjective optimization using nondominated sorting in genetic algorithms. *Evol. Comput.* **1994**, *2*, 221–248. [CrossRef]
9. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197. [CrossRef]
10. Knowles, J.; Corne, D. M-PAES: A memetic algorithm for multiobjective optimization. *Congr. Evol. Comput.* **2000**, *1*, 325–332.

11. Zhang, Q.; Li, H. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Trans. Evol. Comput.* **2007**, *11*, 712–731. [CrossRef]
12. Deb, K.; Mohan, M.; Mishra, S. Evaluating the $\epsilon$-domination based multi-objective evolutionary algorithm for a quick computation of Pareto-optimal solutions. *Evol. Comput.* **2005**, *13*, 501–525. [CrossRef]
13. Zitzler, E.; Laumanns, M.; Thiele, L. SPEA2: Improving the Strength Pareto Evolutionary Algorithm. *Evol. Methods Des. Optim. Control. Appl. Ind. Probl.* **2001**, *103*, 95–100. [CrossRef]
14. Zhang, X.; Tian, Y.; Jin, Y. A knee point-driven evolutionary algorithm for many-objective optimization. *IEEE Trans. Evol. Comput.* **2014**, *19*, 761–776. [CrossRef]
15. Yang, S.; Li, M.; Liu, X.; Zheng, J. A grid-based evolutionary algorithm for many-objective optimization. *IEEE Trans. Evol. Comput.* **2013**, *17*, 721–736. [CrossRef]
16. Salcedo-Sanz, S.; Manjarres, D.; Pastor-Sánchez, Á.; Ser, J.; Portilla-Figueras, J.; Gil-Lopez, S. One-way urban traffic reconfiguration using a multi-objective harmony search approach. *Expert Syst. Appl.* **2013**, *40*, 3341–3350. [CrossRef]
17. Sadollah, A.; Eskandar, H.; Bahreininejad, A.; Kim, J. Water cycle algorithm for solving multi-objective optimization problems. *Soft Comput.* **2015**, *19*, 2587–2603. [CrossRef]
18. Zouache, D.; Moussaoui, A.; Abdelaziz, F. A cooperative swarm intelligence algorithm for multi-objective discrete optimization with application to the knapsack problem. *Eur. J. Oper. Res.* **2018**, *264*, 74–88. [CrossRef]
19. Kumawat, I.; Nanda, S.; Maddila, R. Multi-objective whale optimization. In Proceedings of the Tencon 2017-2017 IEEE Region 10 Conference, Penang, Malaysia, 5–8 November 2017; pp. 2747–2752.
20. Abdel-Basset, M.; Mohamed, R.; Mirjalili, S. A novel Whale Optimization Algorithm integrated with Nelder–Mead simplex for multi-objective optimization problems. *Knowl.-Based Syst.* **2021**, *212*, 106619. [CrossRef]
21. Abdel-Basset, M.; Mohamed, R.; Mirjalili, S.; Chakrabortty, R.; Ryan, M. MOEO-EED: A multi-objective equilibrium optimizer with exploration–exploitation dominance strategy. *Knowl.-Based Syst.* **2021**, *214*, 106717. [CrossRef]
22. Wang, Z.; Li, H.; Yu, H. MOEA/UE: A novel multi-objective evolutionary algorithm using a uniformly evolving scheme. *Neurocomputing* **2021**, *458*, 535–545. [CrossRef]
23. Wang, W.; Tian, G.; Yuan, G.; Pham, D.T. Energy-time tradeoffs for remanufacturing system scheduling using an invasive weed optimization algorithm. *J. Intell. Manuf.* **2021**, *34*, 1065–1083. [CrossRef]
24. Eberhart, R.; Kennedy, J. A new optimizer using particle swarm theory. In Procedings of the MHS'95 Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan, 4–6 October 1995; pp. 39–43.
25. Coello, C.; Pulido, G.; Lechuga, M. Handling multiple objectives with particle swarm optimization. *IEEE Trans. Evol. Comput.* **2004**, *8*, 256–279. [CrossRef]
26. Dorigo, M.; Caro, G. Ant colony optimization: A new meta-heuristic. *Congr. Evol. Comput.* **1999**, *2*, 1470–1477.
27. Alaya, I.; Solnon, C.; Ghedira, K. Ant colony optimization for multi-objective optimization problems. In Procedings of the 19th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2007), Patras, Greece, 29–31 October 2007; Volume 1, pp. 450–457.
28. Mirjalili, S. The ant lion optimizer. *Adv. Eng. Softw.* **2015**, *83*, 80–98. [CrossRef]
29. Houssein, E.; Mahdy, M.; Shebl, D.; Manzoor, A.; Sarkar, R.; Mohamed, W. An efficient slime mould algorithm for solving multi-objective optimization problems. *Expert Syst. Appl.* **2022**, *187*, 115870. [CrossRef]
30. Chalabi, N.; Attia, A.; Bouziane, A.; Hassaballah, M. An improved marine predator algorithm based on epsilon dominance and Pareto archive for multi-objective optimization. *Eng. Appl. Artif. Intell.* **2023**, *119*, 105718. [CrossRef]
31. Mohamed, A.; Hadi, A.; Mohamed, A. Gaining-sharing knowledge based algorithm for solving optimization problems: A novel nature-inspired algorithm. *Int. J. Mach. Learn. Cybern.* **2020**, *11*, 1501–1529. [CrossRef]
32. Reynolds, R.G. An introduction to cultural algorithms. In *Evolutionary Programming—Proceedings of the Third Annual Conference*; Sebald, A.V., Fogel, L.J., Eds.; World Scientific Press: San Diego, CA, USA, 1994; pp. 131–139.
33. Geem, Z.W.; Kim, J.H.; Loganathan, G.V. A new heuristic optimization algorithm: Harmony search. *Simulation* **2001**, *76*, 60–68. [CrossRef]
34. Kashan, A.H. League championship algorithm: A new algorithm for numerical function optimization. In Proceedings of the 2009 International Conference of Soft Computing and Pattern recognition, Malacca, Malaysia, 4–7 December 2009; pp. 43–48.
35. Rao, R.V.; Savsani, V.J.; Vakharia, D. Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems. *Comput.-Aided Des.* **2011**, *43*, 303–315. [CrossRef]
36. Shi, Y. Brain storm optimization algorithm. In *Advances in Swarm Intelligence: Second International Conference, ICSI 2011, Chongqing, China, 12–15 June 2011, Proceedings, Part I 2*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 303–309.
37. Kulkarni, A.J.; Durugkar, I.P.; Kumar, M. Cohort intelligence: A self supervised learning behavior. In Procedings of the 2013 IEEE International Conference on Systems, Man, and Cybernetics, Manchester, UK, 13–16 October 2013; pp. 1396–1400.
38. Moosavian, N.; Roodsari, B.K. Soccer league competition algorithm: A novel meta-heuristic algorithm for optimal design of water distribution networks. *Swarm Evol. Comput.* **2014**, *17*, 14–24. [CrossRef]
39. Huan, T.T.; Kulkarni, A.J.; Kanesan, J.; Huang, C.J.; Abraham, A. Ideology algorithm: A socio-inspired optimization methodology. *Neural Comput. Appl.* **2017**, *28*, 845–876. [CrossRef]
40. Moghdani, R.; Salimifard, K. Volleyball Premier League Algorithm. *Appl. Soft Comput. J.* **2018**, *64*, 161–185. [CrossRef]
41. Khatri, A.; Gaba, A.; Rana, K.; Kumar, V. A novel life choice-based optimizer. *Soft Comput.* **2020**, *24*, 9121–9141. [CrossRef]

42. Elsisi, M. Future search algorithm for optimization. *Evol. Intell.* **2019**, *12*, 21–31. [CrossRef]
43. Shaheen, A.M.; Ginidi, A.R.; El-Sehiemy, R.A.; Ghoneim, S.S. A forensic-based investigation algorithm for parameter extraction of solar cell models. *IEEE Access* **2020**, *9*, 1–20. [CrossRef]
44. Wagan, A.I.; Shaikh, M.M. A new metaheuristic optimization algorithm inspired by human dynasties with an application to the wind turbine micrositing problem. *Appl. Soft Comput.* **2020**, *90*, 106176.
45. Emami, H. Anti-coronavirus optimization algorithm. *Soft Comput.* **2022**, *26*, 4991–5023. [CrossRef] [PubMed]
46. Agrawal, P.; Ganesh, T.; Oliva, D.; Mohamed, A. S-shaped and V-shaped gaining-sharing knowledge-based algorithm for feature selection. *Appl. Intell.* **2022**, *52*, 81–112. [CrossRef]
47. Xiong, G.; Yuan, X.; Mohamed, A.; Chen, J.; Zhang, J. Improved binary gaining-sharing knowledge-based algorithm with mutation for fault section location in distribution networks. *J. Comput. Des. Eng.* **2022**, *9*, 393–405. [CrossRef]
48. Agrawal, P.; Ganesh, T.; Mohamed, A. Solving knapsack problems using a binary gaining sharing knowledge-based optimization algorithm. *Complex Intell. Syst.* **2022**, *8*, 43–63. [CrossRef]
49. Li, C. Takagi–Sugeno fuzzy based power system fault section diagnosis models via genetic learning adaptive GSK algorithm. *Knowl.-Based Syst.* **2022**, *255*, 109773. [CrossRef]
50. Ortega-Sánchez, N. Identification of apple diseases in digital images by using the Gaining-sharing knowledge-based algorithm for multilevel thresholding. *Soft Comput.* **2022**, *26*, 2587–2623. [CrossRef]
51. Hassan, S.; Agrawal, P.; Ganesh, T.; Mohamed, A. A Novel Discrete Binary Gaining-Sharing Knowledge-Based Optimization Algorithm for the Travelling Counselling Problem for Utilization of Solar Energy. *Int. J. Swarm Intell. Res.* **2022**, *13*, 1–24. [CrossRef]
52. Xiong, G.; Li, L.; Mohamed, A.; Yuan, X.; Zhang, J. A new method for parameter extraction of solar photovoltaic models using gaining–sharing knowledge based algorithm. *Energy Rep.* **2021**, *7*, 3286–3301. [CrossRef]
53. Mohamed, A.; Hadi, A.; Mohamed, A.; Awad, N. Evaluating the Performance of Adaptive GainingSharing Knowledge Based Algorithm on CEC 2020 Benchmark Problems. In Proceedings of the 2020 IEEE Congress on Evolutionary Computation, CEC 2020, Glasgow, UK, 19–24 July 2020; pp. 1–8. [CrossRef]
54. Mohamed, A.; Abutarboush, H.; Hadi, A.; Mohamed, A. Gaining-sharing knowledge based algorithm with adaptive parameters for engineering optimization. *IEEE Access* **2021**, *9*, 65934–65946. [CrossRef]
55. Wolpert, D.; Macready, W. No free lunch theorems for optimization. *IEEE Trans. Evol. Comput.* **1997**, *1*, 67–82. [CrossRef]
56. Rai, R.; Das, A.; Ray, S.; Dhal, K.G. Human-inspired optimization algorithms: Theoretical foundations, algorithms, open-research issues and application for multi-level thresholding. *Arch. Comput. Methods Eng.* **2022**, *29*, 5313–5352. [CrossRef]
57. Zitzler, E.; Deb, K.; Thiele, L. Comparison of multiobjective evolutionary algorithms: Empirical results. *Evol. Comput.* **2000**, *8*, 173–195. [CrossRef]
58. Deb, K.; Thiele, L.; Laumanns, M.; Zitzler, E. Scalable multi-objective optimization test problems. *Congr. Evol. Comput.* **2002**, *1*, 825–830.
59. Kumar, A. A benchmark-suite of real-world constrained multi-objective optimization problems and some baseline results. *Swarm Evol. Comput.* **2021**, *67*, 100961. [CrossRef]
60. Zitzler, E. Evolutionary algorithms for multiobjective optimization. *Methods Appl.* **1999**, *63*, 1–134.
61. Tian, Y.; Cheng, R.; Zhang, X.; Jin, Y. PlatEMO: A MATLAB Platform for Evolutionary Multi-Objective Optimization[Educational Forum. *IEEE Comput. Intell. Mag.* **2017**, *12*, 73–87. [CrossRef]
62. Kannan, B.; Kramer, S. An augmented lagrange multiplier based method for mixed integer discrete continuous optimization and its applications to mechanical design. In Proceedings of the ASME Design Engineering Technical Conference, Albuquerque, NM, USA, 19–22 September 1993; Volume Part F1679, pp. 103–112. [CrossRef]
63. Narayanan, S.; Azarm, S. On improving multiobjective genetic algorithms for design optimization. *Struct. Optim.* **1999**, *18*, 146–155. [CrossRef]
64. Chiandussi, G.; Codegone, M.; Ferrero, S.; Varesio, F. Comparison of multi-objective optimization methodologies for engineering applications. *Comput. Math. Appl.* **2012**, *63*, 912–942. [CrossRef]
65. Deb, K. Evolutionary Algorithms for in Engineering Design. *Evol. Algorithms Eng. Comput. Sci.* **1999**, *2*, 135–161.
66. Osyczka, A.; Kundu, S. A Genetic Algorithm-Based Multicriteria Optimization Method. In Proceedings of the First World Congress of Structural and Multidisciplinary Optimization, Goslar, Germany, 28 May–2 June 1995; pp. 909–914.
67. Azarm, S.; Tits, A.; Fan, M. Tradeoff-driven optimization-based design of mechanical systems. In Proceedings of the 4th Symposium on Multidisciplinary Analysis and Optimization, Atlanta, GA, USA, 4–6 September 1999; p. 4758.
68. Ray, T.; Liew, K. A swarm metaphor for multiobjective design optimization. *Eng. Optim.* **2002**, *34*, 141–153. [CrossRef]
69. Deb, K.; Jain, H. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, Part I: Solving problems with box constraints. *IEEE Trans. Evol. Comput.* **2014**, *18*, 577–601. [CrossRef]
70. Cheng, F.; Li, X. Generalized center method for multiobjective engineering optimization. *Eng. Optim.* **1999**, *31*, 641–661. [CrossRef]
71. Huang, H.; Gu, Y.; Du, X. An interactive fuzzy multi-objective optimization method for engineering design. *Eng. Appl. Artif. Intell.* **2006**, *19*, 451–460. [CrossRef]
72. Steven, G. *Evolutionary Algorithms for Single and Multicriteria Design Optimization*; Osyczka, A., Ed.; Springer: Berlin/Heidelberg, Germany, 2002; Volume 24.

73. Coello, C.; Lamont, G.; Veldhuizen, D. *Evolutionary Algorithms for Solving Multi-Objective Problems*; Springer: Berlin/Heidelberg, Germany, 2007; Volume 5. [CrossRef]
74. Parsons, M.; Scott, R. Formulation of Multicriterion Design Optimization Problems for Solution with Scalar Numerical Optimization Methods. *J. Ship Res.* **2004**, *48*, 61–76. [CrossRef]
75. Fan, L.; Yoshino, T.; Xu, T.; Lin, Y.; Liu, H. A Novel Hybrid Algorithm for Solving Multiobjective Optimization Problems with Engineering Applications. *Math. Probl. Eng.* **2018**, *2018*, 5316379. [CrossRef]
76. Dhiman, G.; Kumar, V. Multi-objective spotted hyena optimizer: A Multi-objective optimization algorithm for engineering problems. *Knowl.-Based Syst.* **2018**, *150*, 175–197. [CrossRef]
77. Mahon, K.; Siddall, J. *Optimal Engineering Design: Principles and Applications (Mechanical Engineering Series*; CRC Press: Boca Raton, FL, USA, 1983; Volume 34. [CrossRef]
78. Zhang, H.; Peng, Y.; Hou, L.; Tian, G.; Li, Z. A hybrid multi-objective optimization approach for energy-absorbing structures in train collisions. *Inf. Sci.* **2019**, *481*, 491–506. [CrossRef]
79. Floudas, C. A Collection of Test Problems for Constrained Global Optimization Algorithms. Available online: https://www.amazon.com/Collection-Problems-Constrained-Optimization-Algorithms/dp/3540530320 (accessed on 5 May 2021).
80. Ryoo, H.; Sahinidis, N. Global optimization of nonconvex NLPs and MINLPs with applications in process design. *Comput. Chem. Eng.* **1995**, *19*, 551–566. [CrossRef]
81. Guillén-Gosálbez, G. A novel MILP-based objective reduction method for multi-objective optimization: Application to environmental problems. *Comput. Chem. Eng.* **2011**, *35*, 1469–1477. [CrossRef]
82. Kocis, G.; Grossmann, I. A modelling and decomposition strategy for the minlp optimization of process flowsheets. *Comput. Chem. Eng.* **1989**, *13*, 797–819. [CrossRef]
83. Kocis, G.; Grossmann, I. Global Optimization of Nonconvex Mixed-Integer Nonlinear Programming (Minlp) Problems in Process Synthesis. *Ind. Eng. Chem. Res.* **1988**, *27*, 1407–1421. [CrossRef]
84. Floudas, C. *Nonlinear and Mixed-Integer Optimization: Fundamentals and Applications*; Oxford University Press: Oxford, UK, 1995.
85. Rathore, A.; Holtz, J.; Boller, T. Synchronous optimal pulsewidth modulation for low-switching-frequency control of medium-voltage multilevel inverters. *IEEE Trans. Ind. Electron.* **2010**, *57*, 2374–2381. [CrossRef]
86. Rathore, A.; Holtz, J.; Boller, T. Optimal pulsewidth modulation of multilevel inverters for low switching frequency control of medium voltage high power industrial AC drives. In Proceedings of the 2010 IEEE Energy Conversion Congress and Exposition, ECCE 2010, Atlanta, GA, USA, 12–16 September 2010. [CrossRef]
87. Edpuganti, A.; Rathore, A. Fundamental Switching Frequency Optimal Pulsewidth Modulation of Medium-Voltage Cascaded Seven-Level Inverter. *IEEE Trans. Ind. Appl.* **2015**, *51*, 3485–3492. [CrossRef]
88. Edpuganti, A.; Dwivedi, A.; Rathore, A.; Srivastava, R. Optimal pulsewidth modulation of cascade nine-level (9L) inverter for medium voltage high power industrial AC drives. In Proceedings of the IECON 2015—41st Annual Conference of the IEEE Industrial Electronics Society, Yokohama, Japan, 9–12 November 2015; pp. 4259–4264. [CrossRef]
89. Edpuganti, A.; Rathore, A. Optimal pulsewidth modulation for common-mode voltage elimination scheme of medium-voltage modular multilevel converter-fed open-end stator winding induction motor drives. *IEEE Trans. Ind. Electron.* **2017**, *64*, 848–856. [CrossRef]
90. Mishra, S.; Kumar, A.; Singh, D.; Misra, R. Butterfly Optimizer for Placement and Sizing of Distributed Generation for Feeder Phase Balancing. In *Advances in Intelligent Systems and Computing*; Springer: Berlin/Heidelberg, Germany, 2019; Volume 799, pp. 519–530. [CrossRef]
91. Biswas, P.; Suganthan, P.; Mallipeddi, R.; Amaratunga, G. Multi-objective optimal power flow solutions using a constraint handling technique of evolutionary algorithms. *Soft Comput.* **2020**, *24*, 2999–3023. [CrossRef]
92. Kumar, A.; Das, S.; Mallipeddi, R. An Inversion-Free Robust Power-Flow Algorithm for Microgrids. *IEEE Trans. Smart Grid* **2021**, *12*, 2844–2859. [CrossRef]
93. Kumar, A.; Jha, B.; Das, S.; Mallipeddi, R. Power Flow Analysis of Islanded Microgrids: A Differential Evolution Approach. *IEEE Access* **2021**, *9*, 61721–61738. [CrossRef]
94. Jha, B.; Kumar, A.; Dheer, D.; Singh, D.; Misra, R. A modified current injection load flow method under different load model of EV for distribution system. *Int. Trans. Electr. Energy Syst.* **2020**, *30*, 12284. [CrossRef]
95. Kumar, A.; Jha, B.; Singh, D.; Misra, R. A New Current Injection Based Power Flow Formulation. *Electr. Power Compon. Syst.* **2020**, *48*, 268–280. [CrossRef]
96. Kumar, A.; Jha, B.; Dheer, D.; Singh, D.; Misra, R. Nested backward/forward sweep algorithm for power flow analysis of droop regulated islanded microgrids. *IET Gener. Transm. Distrib.* **2019**, *13*, 3086–3095. [CrossRef]
97. Kumar, A.; Jha, B.; Singh, D.; Misra, R. Current injection-based Newton–Raphson power-flow algorithm for droop-based islanded microgrids. *IET Gener. Transm. Distrib.* **2019**, *13*, 5271–5283. [CrossRef]

98. Kumar, A.; Jha, B.; Dheer, D.; Misra, R.; Singh, D. A Nested-Iterative Newton-Raphson based Power Flow Formulation for Droop-based Islanded Microgrids. *Electr. Power Syst. Res.* **2020**, *180*, 106131. [CrossRef]
99. Rivas-Dávalos, F.; Irving, M. An approach based on the strength pareto evolutionary algorithm 2 for power distribution system planning. In *Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 2005; Volume 3410, pp. 707–720. [CrossRef]