Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2016) 24: 2349 – 2373
© TÜBİTAK
doi:10.3906/elk-1402-310

# A new ABC-based multiobjective optimization algorithm with an improvement approach (IBMO: improved bee colony algorithm for multiobjective optimization)

**Tahir SAĞ**[1,*], **Mehmet ÇUNKAŞ**[2]

[1]Department of Computer Engineering, Faculty of Technology, Selçuk University, Konya, Turkey

[2]Department of Electrical and Electronics Engineering, Faculty of Technology, Selçuk University, Konya, Turkey

**Received:** 27.02.2014 • **Accepted/Published Online:** 15.07.2014 • **Final Version:** 15.04.2016

**Abstract:** This paper presents a new metaheuristic algorithm based on the artificial bee colony (ABC) algorithm for multiobjective optimization problems. The proposed hybrid algorithm, an improved bee colony algorithm for multiobjective optimization called IBMO, combines the main ideas of the simple ABC with nondominated sorting strategy corresponding to the principal framework of multiobjective optimization such as Pareto-dominance and crowding distance. A fixed-sized external archive to store the nondominated solutions and an improvement procedure to promote the convergence to true Pareto front are used. The presented approach, IBMO, is compared with four representatives of the state-of-the-art algorithms: NSGA2, SPEA2, OMOPSO, and AbYSS. IBMO and the selected algorithms from specialized literature are applied to several multiobjective benchmark functions by considering the number of function evaluations. Then four quality indicators are employed for performance evaluations: general distance, spread, maximum spread, and hypervolume. The results show that the IBMO is superior to the other methods.

**Key words:** Multiobjective optimization, artificial bee colony optimization, evolutionary algorithm, swarm intelligence

## 1. Introduction

In the real world, many optimization problems contain more than one objectives that are generally conflicting with each other to be either minimized or maximized at the same time. For example, the good design for a bridge construction is represented by lower mass and higher strength. A plane design needs to optimize fuel efficiency, weight, and cargo simultaneously. It is not possible to find a single optimal solution that can ensure compliance with all objectives. The best solution means that it is not the worst one for each objective and it is better than the others for one objective at least. In other words, an optimal solution is not dominated by any other solutions in the search space. This is called a nondominated solution or Pareto optimum, while an image formed by Pareto-optimal solutions under the space of objective functions is known as a Pareto front [1].

**Definition 1.** Multiobjective optimization problem (MOP) is defined as:

$$\begin{aligned} \text{maximize/ minimize } y &= f(x) = \{f_1(x), f_2(x), ... f_M(x)\} \\ \text{subject to } g(x) &= \{g_1(x), g_2(x), ..., g_J(x)\} \leq 0 \\ h(x) &= \{h_1(x), h_2(x), ..., h_K(x)\} = 0 \\ \text{where } x &= \{x_1, x_2, ..., x_N\} \in X \\ y &= \{y_1, y_2, ..., y_M\} \in Y, \end{aligned} \tag{1}$$

*Correspondence: tahirsag@selcuk.edu.tr

where $x$ is set of the decision vectors, $X$ is the parameter space, $y$ is the objective vector, and $Y$ is the objective space. $g(x)$ and $h(x)$ are inequality and equality constraints, respectively.

**Definition 2.** Pareto dominance: Let $\vec{u} = [u_1, u_2, \ldots, u_n]$ and $\vec{v} = [v_1, v_2, \ldots, v_n]$ be decision vectors (solutions) and $\vec{u}$ dominates $\vec{v}$ is shown as $\vec{u} \prec \vec{v}$, if and only if $\forall i \in \{1, 2, \ldots, n\} : u_i \leq v_i \bigwedge \exists j \in \{1, 2, \ldots, n\} : u_i < v_i$ and $u \neq v$.

The algorithms based on evolutionary computation (EC) are stochastic techniques that try to solve problems by using bio-inspired operators on a population of solutions. The population-based approaches can easily find multiple nondominated solutions and are able to approximate the whole Pareto front in a single run, which is an advantage for multiobjective optimization problems [2]. In the last few decades, EC has been studied for dealing with nondeterministic multiobjective optimization problems. The multiobjective evolutionary optimization algorithms (MOEAs) have become a very popular and fast-growing field [3–5].

Since Schaffer's pioneering work in 1985 [6], various MOEAs have been proposed in this field to solve several domains of science and engineering problems. Some of them are as follows: scheduling heuristics, data mining, rule extraction, assignment and management, wireless sensor network, circuit design, bioinformatics, control systems and robotics, image processing, and neural network training [7]. There are several possible ways to classify MOEAs. Coello et al. in [8] categorize the algorithms in three groups: aggregating function approaches, population-based approaches, and Pareto-based approaches. Aggregating function combines all objectives into a single one using arithmetical operations and is the oldest mathematical approach for multiobjective optimization. However, it has a poor selection pressure to obtain the whole Pareto front. Population-based approaches try to diversify the search space by generating subpopulations for each objective but they are inefficient for the concept of Pareto dominance. Pareto-based approaches use the selection process based on the concept of Pareto optimality. The most advanced and commonly used algorithms take place in this group. Some of them are the niched Pareto genetic algorithm [9], the nondominated sorting genetic algorithm (NSGA) [10] and its improved version NSGA-II [11], the Pareto archive evolutionary strategy (PAES) [12], and the strength Pareto evolutionary algorithm (SPEA) [13] and its improved version SPEA2 [14].

In recent years, one of the other popular stochastic techniques is the algorithms of swarm intelligence, inspired by collective working behaviors of social insect colonies or animal swarms. These algorithms have become very popular owing to the fact that they are simple to implement and produce good results at low computational cost. The representative algorithms of swarm intelligence are particle swarm optimization (PSO), developed by Eberhart and Kennedy in 1995 and inspired by the social behavior of bird flocking or fish schooling [15]; ant colony optimization (ACO), developed by Dorigo in 1999 and mimicking the foraging behavior of social ants [16]; and artificial bee colony (ABC) optimization, developed recently by Karaboğa in 2005, which simulates the intelligence foraging behavior of honeybee swarms [17].

ACO is a preferred method for discrete optimization problems due to its nature. In consideration of multiobjective optimization, ACO is generally applied to traveling salesman problems [18], vehicle routing problems [19], flow-shop scheduling problems [20], and portfolio selection [21].

Research has especially focused on developing multiobjective particle swarm optimization (MOPSO) algorithms due to their population-based nature and rapid convergence speed. A wide variety of algorithms are proposed in the literature [22]. The first version of PSO for multiobjective optimization based on Pareto dominance was developed by Moore and Chapman in 1999 [23]. Nondominated solutions are added to a continuously updated list, but it needs a diversification approach. Li presented a modified PSO that makes use of nondominated sorting (NSPSO) for a better dominance comparison [24]. A combined population is composed of all *pbest* solutions and all recently obtained solutions. The combined population is used to update the personal best of each particle unlike a standard PSO. This method assigns the leader randomly from an external set and also makes use of the niche count and crowding distance techniques for this selection process. Sierra and Coello introduced OMOPSO, which is another remarkable MOPSO approach based on Pareto dominance [25]. This approach utilizes the crowding factor of NSGA2 to select the leader particles. In order to fix archive size, it incorporates evolutionary mutation operators with the concept of $\varepsilon$-dominance. Nebro et al. reported a comparison of the performance of the most advanced MOPSOs in the literature [26]. They proposed a new MOPSO that depends on limiting the velocity values of particles in a swarm, so as to obtain more effective solutions quickly [27]. Similar to the other MOPSOs, their proposed approach also has a mutation operator and an external set to store nondominated solutions.

On the other hand, several successful hybrid metaheuristic algorithms have been proposed for solving MOPs in the literature such as SSPMO [28], AbYSS [29], and HMOEA [30]. Molina et al. [28] incorporated the tabu search procedure into well-known scatter-search method for SSPMO. This approach includes two phases. The first is the generation of an initial solution set. The algorithm makes use of various tabu searches for selecting the best solutions for each objective. The tabu search is applied for each objective in sequence by starting at an arbitrary point. In the second phase, the solutions selected from a reference set are combined by means of scatter search. AbYSS is a hybrid metaheuristic based on scatter search that is adapted for multiobjective optimization and that includes multiobjective procedures such as Pareto dominance, density estimation, and external archive, and also makes use of evolutionary operators such as mutation and crossover. As a consequence of experiments, Nebro et al. [29] achieved fairly competitive results with AbYSS. In HMOEA, Tang and Wang [30] presented another competitive hybrid MOEA that combines the principles of PSO and multiple crossover operators for updating population. A self-adaptive mechanism is used for choosing the crossover operator and the propagating mechanism. Therefore, the diversity of the external archive is improved.

In the present study, a new multiobjective optimization algorithm within a well-constructed improvement procedure that ensures fast-converging to the Pareto front has been developed. The proposed algorithm is also easy to use, whereby it is based on artificial bee colony (ABC) algorithm, for which success has been reported in several studies. The presented algorithm, called the improved bee colony algorithm for multiobjective optimization (IBMO), utilizes the improvement procedure to promote the convergence to the Pareto front in both phases of employed and onlooker bees. IBMO starts with a diversification method to generate the initial swarm so that randomness can be spread over all of the search space. Furthermore, the algorithm makes use of classical multiobjective optimization concepts such as Pareto dominance and external archive (EXA) to store the nondominated solutions. In order to accelerate an iteration and reduce the number of function evaluation, the EXA is used as fixed-sized. To deal with the fixing problem, IBMO employs the crowding distance strategy of NSGA2. The crowding distance is a measure of how close a solution is to its neighbors in PF. The large average crowding distance has a better diversity in the population. IBMO is a robust and stable hybrid approach that incorporates the advantages of NSGA2 with ABC. Explicit diversity preservation and elitism components of NSGA2 are applied to a modified ABC algorithm.

IBMO was compared with other popular multiobjective optimization algorithms: NSGA2, SPEA2, OMOPSO, and AbYSS. Other algorithms and IBMO were applied to 26 benchmark functions for 25,000 and 50,000 function evaluations. Four quality indicators were employed for performance evaluations: general distance, spread, maximum spread, and hypervolume. In consideration of the evaluations, IBMO achieves well-distributed and quite promising results.

## 2. Related works

This paper focuses on not only the ABC algorithm for solving MOPs, but also presents a local search using the main component of NSGA2 to improve the converging of the ABC to the Pareto front. In this section, brief explanations of the simple ABC are given and several studies based on a multiobjective ABC are examined.

## 3. Artificial bee colony optimization algorithm

The ABC algorithm, which is one of the most recently introduced optimization algorithms, simulates the intelligent foraging behavior of a honey bee swarm [17]. The collective intelligence of honeybee swarms consists of three essential components: food sources, employed foragers, and onlooker foragers.

In the model adapted by the ABC algorithm, the position of a food source represents a possible solution to the optimization problem. The nectar amount of a food source corresponds to the quality of the associated solution, and so it is defined as fitness of solution. The number of the employed bees is equal to the number of solutions in the swarm, while the number of onlooker bees is equal to the number of employed bees. The core steps of the algorithm can be summarized as in Figure 1.

Create initial food sources (swarm) with respect to equation 2.

$$x_{ij} = x_{min}^{j} + rand(0,1)(x_{max}^{j} - x_{min}^{j}) \tag{2}$$

where $x_i$ indicates a solution in the population, $i = \{1,2,\ldots SN\}$ and $SN$ is the size of population $j = \{1,2,\ldots D\}$ and $D$ is the number of parameters (decision variables) in a solution. According to the selected subrange, $x_{min}^{j}$, is the lower bound of $j^{\text{th}}$ parameter and $x_{max}^{j}$ is the upper bound of $j^{\text{th}}$ parameter.

Calculate fitness values of swarm with respect to equation 3.

$$fit_{in} = \begin{Bmatrix} 1/(1 + f_{in}) & f_{in} \geq 0 \\ 1 + abs(f_{in}) & f_{in} < 0 \end{Bmatrix} \tag{3}$$

where $fit_i$ is the fitness value of $i^{th}$ solution in the population.

REPEAT

Produce new solutions $v_i$ for the employed bees by using equation 4 and evaluate them

$$delta = \emptyset_{ij}(x_{ij} - x_{kj}) \quad \text{and} \quad v_{ij} = x_{ij} + delta \tag{4}$$

where $delta$ is the variance value of new parameter. $\emptyset_{ij}$ is a random number between [-1, 1]. $v_i$ represents new solution, $x_k$ is the neighbor solution selected randomly, $k \in \{1,2,\ldots,SN\}$. $x_{ij}$ is $j^{t}$ parameter of $i^{th}$ solution represented by current employed bee. $j$ is a random number that is stated in the range of $[1, D]$.

Calculate the probability values $p_i$ for the solutions $x_i$ by equation 5.

$$p_i = \frac{fit_i}{\sum_{n=1}^{SN} fit_i} \tag{5}$$

Produce the new solutions $v_i$ for the onlookers from the solutions $x_i$ selected depending on $p_i$ and evaluate them

Apply the greedy selection process for the onlookers

Determine the abandoned solution for the scout, if exists, and replace it with a new randomly produced solution xi by equation 2.

Memorize the best food source found so far

UNTIL requirements are met

**Figure 1.** Pseudocode of simple ABC algorithm.

## 4. Several related studies based on multiobjective ABC

In single-objective optimization, the ABC has been used for numeric multiple dimensional problems. Its success has been proven by obtaining superior or competitive performance compared to the other well-known evolutionary and swarm-intelligence algorithms [31,32]. Although more than 30 studies about MOPSO exist in the specialized literature, there are only a few remarkable ABC based multiobjective optimization articles published so far [26].

Omkar et al. developed a vector evaluated ABC algorithm based MOABC (VEABC). They applied the algorithm to solve a bi-objective optimization problem that is formulated with minimizing weight and total cost of the composite component [33]. VEABC is a population-based approach inspired by the main ideas of Shaffer's VEGA. However, the proposed approach is simple to implement and well suited for a specialized bi-objective problem. A typical disadvantage of VEGA is that it fails to keep diversity and converges more towards one of the objectives in the case of the existence of more than two objectives after several generations.

Hedayatzadeh et al. (2010) introduced a new MOABC with a grid-based method to sustain the diversity in a fixed-sized external archive [34]. In order to update the external archive, the $\varepsilon$-dominance method is used. The employee and onlooker bees fly for improving the solutions by means of neighbor of archive and food source, respectively. Finally the scout bees generate new solutions instead of expired solutions.

Zou et al. proposed two new algorithms based on ABC to deal with MOPs, NSABC and its improved version MOABC [35]. In contrast to the nature of the original ABC, NSABC combines the nondominated sorting strategy of NSGA2 with only the onlooker bee phase of the ABC. In addition to NSABC, Zou's MOABC includes a comprehensive learning method for improving the diversity of the swarm. Taking into account the outcomes of diversity and convergence metrics obtained from benchmark test problems, it can be seen from the standard deviation values that MOABC is better. However, the proposed algorithms do not have a stable running characteristic in multiple runs.

Zhang et al. presented the multihive artificial bee colony for constraint multiobjective optimization (MHABC-CMO) [36]. The proposed approach is based on two concepts. The first is the principle of nondominated sorting but it also uses multiple swarms. The second is the symbiosis theory, which includes procedures of solution exchange among swarms and information sharing within swarms for coevolution. This approach also uses a self-adaptive penalty method to deal with constraints.

Largo et al. proposed a new hybrid multiobjective algorithm (MO-ABC/DE) that combines the ABC with a differential evolution (DE) algorithm for unconstrained MOPs [37]. The algorithm makes use of Pareto dominance and crowding distance, which are well-known multiobjective procedures. In this method, the ABC is used as the main framework of the algorithm and the DE is used for improving the obtained solutions in both employed and onlooker bee phases. The researchers reported that their methods were not the best among the other compared algorithms. However, it gives a promising result for constrained problems as a future work.

Ríos et al. [38] implemented two multiobjective swarm algorithms, multiobjective artificial bee colony (MOABC) and multiobjective gravitational search algorithm (MOGSA), for meta-schedulers. Multiobjective properties such as the optimization of execution time and cost per experiment are improved. These algorithms are compared with the nondominated sorting genetic algorithm II (NSGA II).

## 5. Description of the proposed algorithm

The proposed algorithm, IBMO, consists of the adaptation of a single-objective ABC algorithm to multiobjective optimization by means of the nondominated sorting strategy of NSGA2 and the well-known principal concepts of Pareto-based multiobjective optimization. Moreover, a method is proposed to promote the convergence of the obtained solutions towards the true Pareto front. The main steps of IBMO are given in Figure 2. In this section, the execution of the algorithm is discussed in detail.

**THE MAIN STEPS OF IBMO ALGORITM**

```
00  Set the control parameters of the algorithm such as:
        NP:            the number of colony size (employed + onlooker bees)
                       thereby SN = NP / 2 – it is size of swarm
        maxFes
        or MNC:        maximum number of function evaluations or cycle as
                       termination criterion,
        nEXA:          size of external archive (EXA)
        limit:         A step value for improvement method.
01  Generate an initial swarm positions (x_i, i = 1 ...SN) with respect to diversification
    method detailed in Figure 3.
02  Evaluate an initial swarm
03  Store the nondominated solutions in EXA from initial swarm
04  REPEAT
05    Set newGeneratedFoods set as an empty set
06    for each employed bee {
07        Select a neighbor solution from the Foods set randomly
08        Determine the parameter to change randomly but also depend on frequency
          values explained in Figure 4.
09        Produce new solution u_i by using Eq. (4)
10        Improve the new produced solution by using the method in Figure 5.
11        Add the improved solution into newGeneratedFoods set
12    } // end of for each
13    Update EXA via newGeneratedFoods
14    Calculate the probabilities with respect to crowdings of EXA
15    Reset newGeneratedFoods set
16    for each onlooker bee {
17        Select a neighbor solution from the EXA correspond with probabilities
18        Determine a parameter to change randomly as in Figure 4.
19        Produce new solution u_i by using Eq. (4)
20        Improve the new produced solution by using the method in Figure 5.
21        Add the improved solution into newGeneratedFoods set
22    } // end of for each
23    Update EXA and Foods List
24  UNTIL requirements are met
```

**Figure 2.** Pseudocode of IBMO algorithm.

**Step #00 – Initialization:** The control parameters of the algorithm are set. These parameters are the constant values that are used to construct the required data structure in accordance with the analogy of bio-inspired algorithm. These are the size of colony ($NP$), the size of external archive ($nEXA$), maximum number of function evaluations ($maxFes$), maximum number of cycles ($MNC$), and a step value for the improvement method ($limit$). The proposed bee colony is composed of equal numbers of employed and onlooker bees.

Therefore, *NP* has to be defined at the beginning of a run. It is a problem-independent value due to the complexity of search space, the number of variables, and objectives. The external archive dependent on crowding distance is used to store the nondominated solutions found so far, which is a crucial factor for elitism. However, its size, *nEXA,* should be limited for accelerating the convergence of the swarm. *maxFes* and *MNC* are employed as the termination metric of the algorithm. Finally, *limit* specifies how many times the improvement method is applied to a solution.

**Step #01 – The first generation of the population with a diversification method:** The algorithm starts with a diversification technique used in [24,30,39] to generate the initial food sources (initial solutions). It works for obtaining a well-distributed population. Each parameter of a solution is divided into equal sized subranges as shown in Figure 3. A subrange is assigned inversely proportional to frequency count. Then the parameter values are generated randomly with respect to Eq. (2) for its selected range. The frequency count is incremented by one. This process is repeated for each parameter of a new solution to be generated. The more frequency count of a range, the less selection probability of the related range and vice versa.

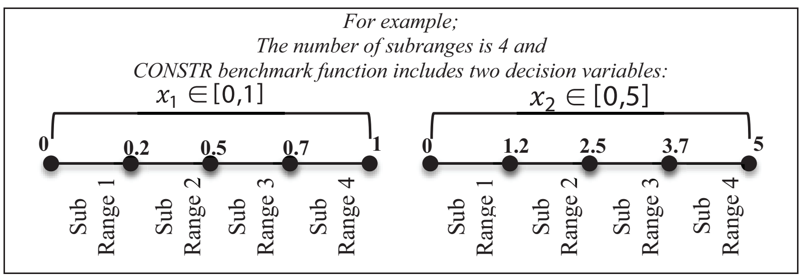


**Figure 3.** Diversification method.

**Step #02 – Evaluation:** All new generated solutions in the population are evaluated in this step. The fitness values of each objective are separately calculated with Eq. (3) suited to the nature of a single-objective ABC algorithm.

**Step #03:** Nondominated solutions are extracted from the current population and stored in the external archive by using the fitness of the solutions in the concept of Pareto dominance.

**Step #04:** The main loop of the algorithm starts and goes on until the termination conditions are satisfied. The criteria can be the number of cycles or function evaluations.

**Step #05:** Although a single-objective ABC algorithm uses a greedy selection mechanism between the old and new generated solution, the proposed multiobjective ABC algorithm keeps all solutions in another set, *newGeneratedFoods*, in order to increase the selection pressure towards the Pareto front by using the nondominated sorting procedure of NSGA2.

**Step #06 – 12 Employed bee phase:** In the nature of the ABC model, the employed bees are responsible for bringing nectar to the hive and sharing information about food sources with other bees in the hive. For each food source, there exists one employed bee. Here a food source means a separate solution. An employed bee determines a food source (solution) neighboring itself randomly. The neighbor solution is selected from the *Foods* set in this phase and a new solution is generated neighboring the existing solution by using Eq. (4). The newly generated solution is formed by changing one parameter of the current solution. The selected parameter index is assigned inversely proportional with frequency count to increase the selection probability of all parameters in the solutions like frequency approach used in the diversification method. The outline of this method is shown in Figure 4.

```
FUNCTION GetParam2Change
00  Calculate reverse frequencies of the parameters, reverseFrequency
01  Generate a random value between [ 1, reverseFrequency ]
02  if sum of reverseFrequency equals to zero, set param = 1 and go to line 7
03  While value > reverseFrequency[ param ] {
04      Set value = value - reverseFrequency[ param ]
05      Set param = param + 1
06  } // end of while
07  Increase the frequency of param by one
END
```

**Figure 4.** Pseudocode of determining the parameter to change.

The phase of improvement begins after calculating the new solution $v_i$ by using Eq. (4). A greedy selection process is applied to the solution $v_i$, which makes use of the constrained dominance method used in NSGAII. With the aim of promoting convergence speed towards the true Pareto front, the obtained delta value in Eq. (4) is added to $j^{th}$ parameter of $v_i$ until the new solution is better than the old one or the number of iterations exceeds the parameter value of limit. In the case that delta equals zero, it is randomly regenerated between [–0.5, 0.5]. This process provides an accelerated convergence effect within the selected parameter. However, a small number of iterations such as 5 is adequate and it is a crucial factor to prevent a great number of function evaluations. The pseudocode of the proposed improvement method is shown in Figure 5.

```
00  if delta equals to zero; set delta = rand(-0.5, 0.5) otherwise continue
01  Set newSol = v_i
02  Set temp = v_i
03  Set limit_count = 0
04  REPEAT
05      Set limit_count = limit_count + 1
06      Set temp_j = temp_j + delta * limit_count
07      if temp dominates newSol; break
08      Set newSol = temp
09  UNTIL limit_count < limit
10  Store newSol into newGeneratedFoods
```

**Figure 5.** Pseudocode of improvement method.

The new solution is added to the *newGeneratedFoods* list after the improvement so that the cardinality of this list becomes equal to population size at the end of the employed bee phase.

**Step #13 – Updating external archive:** The external archive is used for guaranteeing a well-distributed Pareto front by retaining the nondominated solutions found during the search process so that the algorithm can approximate the whole Pareto front. However, it gets hard to control if the new obtained solution will be stored or not in terms of Pareto dominance so long as the cardinality of the archive extends. Therefore, an updating mechanism is needed to limit the archive size without losing diversity. Two commonly used techniques for this purpose in the literature are an adaptive grid strategy formed by hypercubes of PAES [12] and the crowding distance strategy of NSGAII [11]. In IBMO, the external archive is updated by a strategy dependent on crowding distance and the *newGeneratedFoods* list. According to this strategy, all solutions in *newGeneratedFoods* are handled as a new solution in sequence. If this new solution is dominated by the archive, it is discarded to the *restList*. Otherwise, the solution is stored. Moreover, other solutions dominated by the new solution are removed if it exists. When the archive reaches maximum capacity, the solution that has the highest crowding distance value in the archive is removed and the new solution is stored for satisfying the diversity.

**Step #14 – Calculating the selection probabilities:** In this step, a cumulative sum array is calculated to use in the onlooker bee phase. The cumulative sum array determines the range of selection probabilities for each nondominated solution in a roulette wheel. If the current problem discussed here is a constrained optimization problem, there may not be any nondominated solution in the external archive for early iterations of running. If not, the best solutions in the population so far are selected as the nondominated solutions. In the situation of an empty external archive, the constraint values of the solutions in the *Foods* set are used. Otherwise, crowding distance values of the nondominated solutions in the *EXA* are deployed for obtaining the cumulative sum array. In this way, the algorithm can focus on the whole Pareto front as well as converging fast owing to the increment of selection probabilities for the leader solutions that satisfy the constraints better or in a less crowded position.

An outline of the algorithm of IBMO can be seen in Figure 6 schematically.

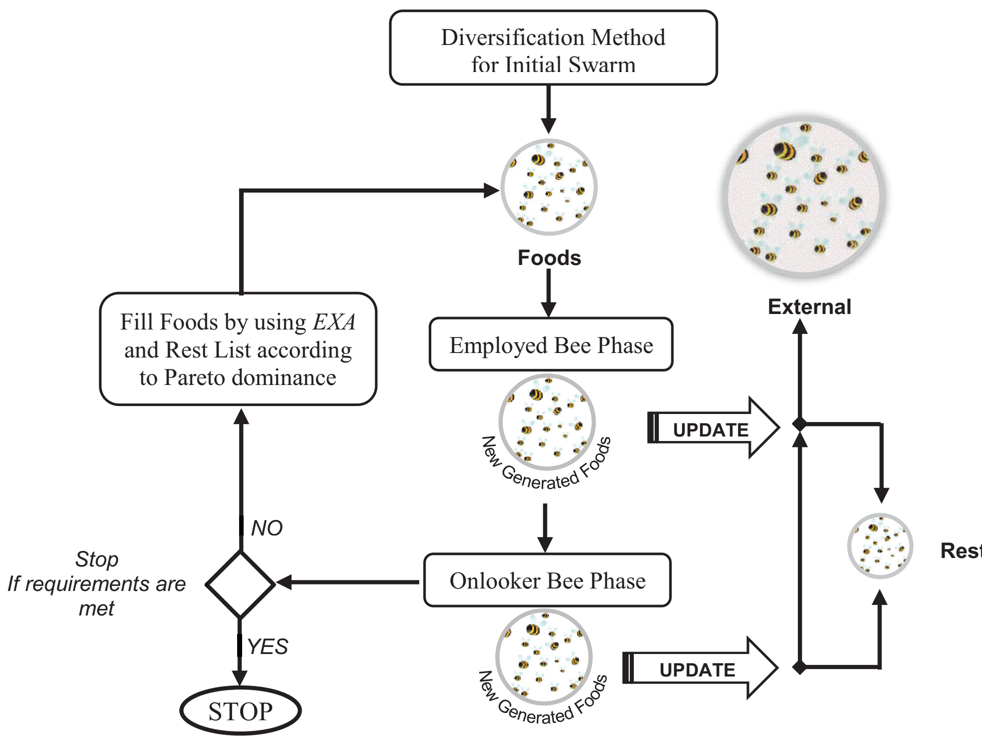


**Figure 6.** The outline of the presented IBMO algorithm.

**Step #15:** After the employed bee phase, the nondominated solutions in the *newGeneratedFoods* set are used to update *EXA* and rest of the solutions are transferred to *restList*. Then it is reinitialized for using in the onlooker bee phase.

**Step #16–22 Onlooker bee phase:** In the nature of the ABC model, onlooker bees obtain knowledge about the quality of food sources by means of following the dance of employed bees. Then they choose a source considering this knowledge. In the essential ABC algorithm, the roulette wheel method is used and the neighbor solution is selected from *Foods* with respect to the probabilities calculated as dependent on fitness values. As distinct from a native ABC, IBMO selects neighbor solutions from the external archive instead of *Foods* set and selection probabilities are calculated as mentioned above in step #14. The determination of the parameter to be changed, the generation of new solutions, and the improvement procedures are applied similarly to the employed bee phase.

**Step #23–24:** At the end of the onlooker bee phase, new solutions are obtained and stored in the *newGeneratedFoods* set. By means of this set, the external archive is updated in the same way as in step #13. Then the *Foods* set is reconstructed for the next iteration by using *EXA* and *restList*. During the search process, the best solutions are held in *EXA* and the rest of the solutions are discarded into *restList*. Primarily, the nondominated solutions in *EXA* are chosen to fill the *Foods* set in accordance with the concept of Pareto dominance. If the size of *EXA* is greater than the size of the *Foods* set, then the solutions are only selected from the archive randomly. Otherwise, after all members of external archive are added to the *Foods* set definitely, the solutions in *restList* are arranged according to Pareto dominance. The fitness of every solution is assigned a value equal to their nondominated level. The rest of the members of the population are chosen from subsequent nondominated front according to their ranking order. Finally, *restList* is reinitialized so as to use in the next iteration. This approach increases the selection pressure towards the true Pareto front.

## 6. Experimental results

A brief definition of test problems used for comparing the performance of IBMO with NSGAII, SPEA2, OMOPSO, and AbYSS is given first. Then the four performance indices used in this study are explained, and the parameter settings of IBMO and the other four algorithms are described. The experimental results, including the median and interquartile ranges of all indices, are given and analyzed for all test problems and algorithms. All algorithms have been run on a personal computer with an Intel core i7-2600 3.40 Ghz CPU, 10 GB memory, and Windows operating system.

## 7. Test problems

For general performance testing, the use of artificially constructed test functions is appropriate since they have many advantages, such as easy implementation, easy visualization, and known optima in advance. However, several benchmark functions exist in the literature that are designed differently. For the veracity of the comparisons, 26 test functions composed of constrained and unconstrained functions from the literature are used in this study.

The test functions can be handled in four categories: the first is bi-objective unconstrained problems: Schaffer [6], Fonseca [40], Kursawe [41], and Zitzler–Deb–Thiele's (ZDT) test suite [42]. The second is bi-objective constrained problems: Constr [11], Srinivas [10], Osyczka2 [43], Golinski [44], Tanaka [45], and Binh2 [46]. The third category of problems is Deb et al.'s DTLZ test suite [47], where problems are scalable to any number of objectives. This benchmark suite includes nine test problems of which the first seven are shown in the appendix section. Due to their side constraints and high dimensionality, DTLZ8 and DTLZ9 are disregarded here. Finally, the last selected test problem category is composed of the Viennet test suite [48] and Tamaki [49]. Viennet1, 2, and 3 are test problems with three objectives without constraints. Viennet4 has three objectives and three constraints. Viennet's test problem has a discrete set of Pareto fronts. Tamaki has three objectives and one constraint. The formulations of all test problems are included in Appendix B.

## 8. Performance indices

In order to quantify the performance of multiobjective optimization algorithms, two essential concepts are dictated by Pareto optimality: convergence and diversity. The first concept (convergence) means the distance between the obtained nondominated front and true Pareto front. This is crucial for determining the accuracy of the solutions obtained by an algorithm. The second one (diversity) demonstrates the distribution of the solutions found within Pareto-optimal solutions. It identifies to what extent the solutions of an algorithm show a consistent distribution in the different areas of the true Pareto front. In order to measure these two criteria, many methods for solving MOPs have been presented [11,13,50]. In this paper, four performance indices that are commonly used are chosen and their brief explanations are given below.

- **General Distance (GD):** General distance is a well-known convergence metric used for determining how far obtained nondominated solutions are from true Pareto optimal solutions on average. The ideal case is that metric value equals zero. The GD metric is mathematically expressed as follows [51]:

$$GD = \frac{\sqrt{\sum_{i=1}^{nEXA} d_i^2}}{nEXA}, \tag{2}$$

  where $nEXA$ is the cardinality of the obtained nondominated solution set, and $d_i$ is the Euclidean distance between the $i^{th}$ solution in the external archive and the nearest member of the true Pareto front.

- **Spread (SP):** The spread metric is used for measuring the distribution of nondominated solutions found along the true Pareto optimal front. It is based on computing the distance between two consecutive solutions that are nearest neighbors themselves on the normalized objective space. It is desired that SP converges to zero. The SP metric can be defined as follows [24]:

$$SP = \frac{\sum_{i=1}^{n} d\left(e_i, S\right) + \sum_{X \in S^*} \left|d\left(X, S\right)\right| - \bar{d}}{\sum_{i=1}^{n} d\left(e_i, S\right) + \left|S^*\right| . \bar{d}}, \tag{3}$$

  where $S$ is the nondominated solution set obtained by an algorithm, $S^*$ is the true Pareto optimal solution set known in advance, $(e_1, \ldots, e_n)$ are $n$ extreme solutions in $S^*$, $n$ is the number of objectives, and

$$d\left(X, S\right) = \left\|f\left(X\right) - f\left(Y\right)\right\|^2 \tag{4}$$

$$\bar{d} = \frac{1}{\left|S^*\right|} \sum_{X \in S^*} d\left(X, S\right) \tag{5}$$

  $d\left(X, S\right)$ is minimum distance of solution $X$ to true Pareto front, and $\bar{d}$ is the mean distance.

- **Maximum Spread (MS):** Maximum spread is another diversity metric proposed in [52]. This metric especially shows how well the algorithm can find the extreme solutions. It is defined as follows:

$$MS = \sqrt{\frac{1}{n}\sum_{l=1}^{n} \delta_l} \text{ and } \delta_l = \left(\frac{\left(f_l^{\max}, F_l^{\max}\right) - \left(f_l^{\min}, F_l^{\min}\right)}{F_l^{\max} - F_l^{\min}}\right)^2, \tag{6}$$

  where $f_l^{\max}$ is maximum and $f_l^{\min}$ is minimum of the $l^{th}$ objective in the external archive, and $F_l^{\max}$ is maximum and $F_l^{\min}$ is minimum of the $l^{th}$ objective in the true Pareto optimal front. In the case of $f_l^{\min} \geq F_l^{\max}$, it assumes that $\delta_l$ equals zero. It is desired that MS converges to one for the true Pareto front being completely covered by the obtained nondominated solutions.

- **Hypervolume (HV):** Hypervolume is a popular indicator suggested by Zitzler and Thiele for performance assessments; it is also known as S-metric or Lebesgue measure [53]. The obtained nondominated solution set is the points in objective space, and the hypervolume metric calculates the size of the region dominated by the solutions in this set. A larger hypervolume means a better set of trade-offs than a smaller one. For this reason, normalized objective function values are used. For example, hypervolume in three dimensions can be seen in Figure 7. The nondominated solutions, Q = {a, b, c, d}, are marked by circles. The HV metric is calculated with Eq. (7).

$$HV = volume\left(\bigcup_{i=1}^{|Q|} v_i\right) \tag{7}$$

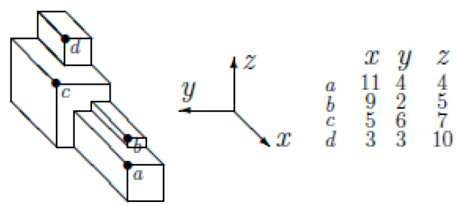


**Figure 7.** Hypervolume in three dimensions.

## 9. Parameter settings

All algorithms have been run for both 25,000 and 50,000 function evaluations. We have executed 100 independent runs for each problem. The median and interquartile range (IQR) of each problem are used as measures of location (or central tendency) and statistical dispersion, respectively. The best result for each test problem has a gray background. If more than one algorithm has the same best median value, then the one with the smaller *IQR* is assumed to be the best. The size of bee colony, *NP*, is 100; thereby population size becomes 50, which is created randomly by diversification method in the initial phase. The size of external archive *EXA* is determined as 100 for bi-objective problems and 400 for problems with three objectives for all considered algorithms. Finally, the limit parameter is assigned to 5 as the number of steps for the improvement method. Unlike other algorithms, the proposed IBMO has an advantage of employing fewer control parameters. These parameter values are also shown in Table 1.

**Table 1.** The outline of the presented IBMO algorithm.

| Parameter | Value |
|---|---|
| *maxFes* | 25,000 and 50,000 |
| *NP* | 100 |
| *SN* | 50 |
| *nEXA* | 100 for biobjective problems<br>400 for problems with more than 2 objectives |
| *limit* | 5 |

The proposed algorithm IBMO is implemented in MATLAB and compared with four state-of-the-art algorithms: NSGAII, SPEA2, AbYSS, and OMOPSO. The applications are implemented in Java using the jMetal framework, which can be downloaded from http://jmetal.sourceforge.net. For each algorithm, suggested parameter settings of the original authors have been used.

Due to the fact that obtained values are generated by stochastic algorithms, statistical reliability analyses are employed to control the results for promoting the confidence of applied processes. For this purpose, first a Kolmogorov–Smirnov test is performed to detect if obtained data have a normal distribution or not. In the case of normal distribution one-way ANOVA is used (for parametric statistical tests), otherwise the Kruskal–Wallis test (for nonparametric statistical tests). The confidence level 0.05 is used for all statistical tests. Multiple comparisons are performed with the aid of SPSS. Successful tests are marked with "+" symbols in the last column in all the tables; conversely, "−" means that no statistical confidence was found (P-value >0.05).

## 10. Results and discussion

The proposed algorithm IBMO is compared with two standard and well-known algorithms, NSGA2 and SPEA2, and two up-to-date algorithms, OMOPSO and AbYSS. These algorithms and the parameter settings used in the experiments are explained in detail in the previous sections.

The computational results of performance indices (GD, SP, MS, and HV) for 25,000 function evaluations are shown in Tables 2, 3, 4, and 5, respectively, according to all five algorithms: IBMO, OMOPSO, AbYSS, NSGA2, and SPEA2. In addition, the metric results after 50,000 function evaluations are given in Appendix A.

As shown in Table 2, the proposed IBMO obtains the lowest GD metric value for nine out of the 26 test problems and with statistical confidence in almost all problems. A low GD metric value of an algorithm on a problem is significant for accuracy of the obtained Pareto front. A well-distributed front is remarkable when approximating to the true Pareto front. IBMO specifically has a better performance (except for significantly two problems: Osycka2 and Golinsky), for the first two groups of problems, which consist of unconstrained and constrained bi-objective problems. Fuerthermore, IBMO achieves good results on the problems with more than two objectives in the last two groups. The IBMO is superior to the other algorithms for DTLZ3, DTLZ7, Viennet, Viennet2, and Tamaki. Concerning the remaining problems, IBMO obtains competitive results with especially AbYSS, which is a well-developed hybrid algorithm.

With respect to the spread metric in Table 3, IBMO achieves quite competitive results compared with other state-of-the-art algorithms on all MOPs. It outperforms the other algorithms on four problems. It should be noted that it is necessary to evaluate the diversification with accuracy of the obtained Pareto front. In this respect, IBMO has generally better GD values with competitive SP values. This case can be seen graphically in Figures 8–12. As shown in Figure 8, IBMO obtains the true Pareto front definitely on constrained and unconstrained MOPs. In addition, the next figures for DTLZ1, DTLZ3, DTLZ4, and DTLZ7 are related MOPs with more than two objectives, respectively. For DTLZ1 seen in Figure 9, IBMO obtains the true Pareto optimal surface almost completely, while OMOPSO and SPEA2 are far from the true Pareto optimal front, and NSGAII and AbYSS have an insufficient distribution along the Pareto front. For DTLZ3 seen in Figure 10, IBMO has the closest front to the true Pareto front and better diversification among the others. For DTLZ4 seen in Figure 11, SPEA2 has the best spreadout with an acceptable accuracy, which is clear from related GD and SP tables. Although IBMO, OMOPSO, SPEA2, and partially NSGA2 obtain the four discrete areas of the Pareto front on DTLZ7, AbYSS is not successful as seen in Figure 12. The algorithms achieve better outcomes for different MOPs. However, it is clear that IBMO has the best surfaces with AbYSS among these five algorithms. The performance of IBMO on these problems is quite good owing to the fact that the proposed improvement method stimulates the selection pressure to bounds of the Pareto surface. A good diversification is obtained by the crowding distance technique on the external archive.

The results of the MS metric are given in Table 4. The MS measures the range of extreme values on the Pareto front for obtained solutions. The values between zero and one are desired to be one. According to

**Table 2.** Median and interquartile range of the general distance metric (25,000 function evaluations).

| Problem | IBMO | | OMOPSO | | AbYSS | | NSGA2 | | SPEA2 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | $\tilde{x}$ | $IQR$ | $\tilde{x}$ | $IQR$ | $\tilde{x}$ | $IQR$ | $\tilde{x}$ | $IQR$ | $\tilde{x}$ | $IQR$ | |
| Schaffer | 2.267E-04 | 2.206E-05 | 2.280E-04 | 2.026E-05 | 2.307E-04 | 1.629E-05 | 4.660E-02 | 1.547E-01 | 4.791E-02 | 1.837E-01 | + |
| Fonseca | 1.353E-04 | 9.399E-06 | 1.239E-04 | 1.136E-05 | 2.327E-04 | 3.429E-05 | 3.530E-04 | 4.801E-05 | 2.162E-04 | 2.503E-05 | + |
| Kursawe | 1.317E-04 | 1.206E-05 | 5.377E-04 | 1.042E-04 | 1.538E-04 | 2.243E-05 | 2.131E-04 | 3.175E-05 | 1.575E-04 | 2.161E-05 | + |
| ZDT1 | 9.742E-05 | 4.633E-05 | 1.427E-04 | 3.821E-05 | 1.831E-04 | 3.199E-05 | 2.192E-04 | 4.767E-05 | 2.206E-04 | 2.510E-05 | + |
| ZDT2 | 5.775E-05 | 3.085E-05 | 7.799E-05 | 2.393E-05 | 1.103E-04 | 4.385E-05 | 1.761E-04 | 4.195E-05 | 1.814E-04 | 4.768E-05 | + |
| ZDT3 | 1.453E-04 | 3.103E-05 | 2.245E-04 | 4.070E-05 | 1.902E-04 | 2.009E-05 | 2.136E-04 | 1.530E-05 | 2.305E-04 | 2.131E-05 | – |
| ZDT4 | 2.558E-04 | 2.555E-04 | 7.393E-01 | 4.664E-01 | 5.737E-04 | 5.181E-04 | 4.286E-04 | 2.853E-04 | 5.209E-04 | 4.420E-04 | + |
| ZDT6 | 1.496E-04 | 1.328E-05 | 2.650E-02 | 4.634E-02 | 1.804E-04 | 1.874E-05 | 8.903E-04 | 1.277E-04 | 1.663E-03 | 2.625E-04 | + |
| Constr | 1.886E-04 | 4.229E-05 | 1.762E-04 | 2.143E-05 | 2.235E-04 | 3.261E-05 | 2.955E-04 | 4.094E-05 | 2.035E-04 | 2.546E-05 | + |
| Srinivas | 4.783E-05 | 1.620E-05 | 5.111E-05 | 1.939E-05 | 6.257E-05 | 2.055E-05 | 1.988E-04 | 3.558E-05 | 1.158E-04 | 2.634E-05 | + |
| Osycka2 | 5.612E-03 | 2.109E-03 | 1.895E-03 | 7.865E-04 | 1.151E-03 | 1.592E-02 | 1.049E-03 | 8.903E-05 | 1.427E-03 | 2.447E-04 | + |
| Golinski | 3.179E-04 | 3.550E-05 | 3.697E-04 | 5.605E-05 | 3.195E-04 | 3.110E-05 | 3.231E-04 | 2.673E-05 | 2.347E-04 | 2.440E-05 | – |
| Tanaka | 5.464E-04 | 2.059E-04 | 8.024E-04 | 1.161E-04 | 7.711E-04 | 9.256E-05 | 7.494E-04 | 1.134E-04 | 6.464E-04 | 1.128E-04 | + |
| Binh2 | 6.761E-04 | 2.797E-05 | 1.245E-03 | 4.450E-05 | 1.249E-03 | 4.456E-05 | 1.387E-03 | 9.750E-05 | 1.582E-03 | 4.950E-05 | + |
| DTLZ1 | 1.515E-02 | 5.174E-02 | 1.025E+01 | 1.471E+00 | 1.974E-03 | 3.543E-02 | 1.054E-02 | 5.824E-02 | 5.794E+00 | 2.410E+00 | + |
| DTLZ2 | 3.006E-04 | 5.841E-05 | 8.746E-04 | 5.099E-05 | 2.001E-04 | 2.757E-05 | 1.235E-03 | 2.862E-04 | 5.608E-04 | 1.222E-04 | + |
| DTLZ3 | 3.224E-01 | 1.891E-01 | 3.455E+01 | 5.756E+00 | 6.187E-01 | 7.386E-01 | 8.410E-01 | 7.291E-01 | 2.053E+01 | 3.672E+00 | + |
| DTLZ4 | 7.806E-04 | 1.514E-04 | 1.398E-03 | 2.024E-04 | 7.058E-04 | 1.978E-04 | 2.048E-03 | 1.518E-04 | 9.453E-04 | 1.140E-04 | + |
| DTLZ5 | 9.934E-02 | 4.844E-03 | 2.596E-04 | 5.758E-06 | 2.520E-04 | 1.704E-06 | 6.300E-04 | 5.525E-05 | 2.721E-04 | 8.066E-06 | + |
| DTLZ6 | 4.570E-01 | 1.184E-02 | 2.524E-04 | 1.733E-06 | 1.014E-01 | 2.762E-02 | 1.685E-01 | 3.117E-02 | 2.977E-01 | 1.427E-02 | + |
| DTLZ7 | 4.324E-04 | 2.051E-04 | 1.421E-03 | 4.009E-04 | 5.284E-04 | 2.114E-04 | 2.815E-03 | 5.203E-04 | 3.129E-03 | 7.753E-04 | + |
| Viennet | 4.672E-04 | 6.208E-05 | 7.029E-04 | 1.080E-04 | 5.287E-04 | 8.470E-05 | 3.411E-03 | 8.552E-04 | 8.982E-04 | 7.119E-05 | + |
| Viennet2 | 1.269E-04 | 3.995E-05 | 1.422E-04 | 3.666E-05 | 1.860E-04 | 7.390E-05 | 7.250E-04 | 3.769E-04 | 1.444E-04 | 1.876E-05 | + |
| Viennet3 | 4.377E-05 | 4.190E-05 | 3.309E-05 | 8.978E-06 | 3.509E-05 | 5.431E-06 | 2.544E-04 | 7.682E-05 | 4.270E-05 | 6.508E-06 | + |
| Viennet4 | 1.378E-04 | 3.383E-05 | 1.001E-04 | 9.411E-06 | 9.663E-05 | 1.359E-05 | 5.280E-04 | 2.104E-04 | 1.155E-04 | 1.558E-05 | + |
| Tamaki | 1.452E-03 | 2.474E-04 | 7.414E-02 | 1.599E-04 | 7.236E-02 | 4.630E-04 | 1.643E-01 | 1.319E-03 | 7.363E-02 | 2.513E-04 | + |

**Table 3.** Median and interquartile range of the spread metric (25,000 function evaluations).

| Problem | IBMO | | OMOPSO | | AbYSS | | NSGA2 | | SPEA2 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | $\tilde{x}$ | $IQR$ | $\tilde{x}$ | $IQR$ | $\tilde{x}$ | $IQR$ | $\tilde{x}$ | $IQR$ | $\tilde{x}$ | $IQR$ | |
| Schaffer | 1.136E-01 | 2.178E-02 | 1.114E-01 | 2.153E-02 | 1.284E-01 | 2.215E-02 | 6.450E-01 | 4.484E-01 | 4.921E-01 | 5.830E-01 | + |
| Fonseca | 7.055E-02 | 1.266E-02 | 5.981E-02 | 1.796E-02 | 1.032E-01 | 1.979E-02 | 3.650E-01 | 5.502E-02 | 1.236E-01 | 2.382E-02 | + |
| Kursawe | 2.290E-01 | 7.056E-03 | 2.987E-01 | 3.078E-02 | 2.410E-01 | 1.003E-02 | 5.424E-01 | 8.873E-02 | 2.947E-01 | 2.124E-02 | + |
| ZDT1 | 3.102E-01 | 6.050E-02 | 8.589E-02 | 1.853E-02 | 1.128E-01 | 1.869E-02 | 3.907E-01 | 7.425E-02 | 1.322E-01 | 1.805E-02 | + |
| ZDT2 | 2.507E-01 | 4.962E-02 | 8.697E-02 | 1.715E-02 | 1.153E-01 | 2.047E-02 | 3.920E-01 | 5.727E-02 | 1.410E-01 | 1.813E-02 | + |
| ZDT3 | 8.122E-01 | 2.047E-01 | 1.562E-01 | 3.502E-02 | 1.406E-01 | 2.107E-01 | 3.822E-01 | 8.212E-02 | 1.668E-01 | 2.281E-02 | + |
| ZDT4 | 6.230E-01 | 1.445E-01 | 8.837E-01 | 9.390E-02 | 1.333E-01 | 2.880E-02 | 3.985E-01 | 7.499E-02 | 2.081E-01 | 1.040E-01 | + |
| ZDT6 | 1.084E-01 | 2.679E-02 | 9.459E-01 | 1.250E+00 | 9.501E-02 | 1.477E-02 | 3.562E-01 | 6.074E-02 | 2.045E-01 | 2.823E-02 | + |
| Constr | 2.787E-01 | 1.072E-01 | 1.231E-01 | 2.303E-02 | 1.601E-01 | 2.169E-02 | 3.909E-01 | 5.702E-02 | 5.233E-01 | 3.461E-02 | + |
| Srinivas | 8.431E-02 | 1.548E-02 | 7.759E-02 | 1.381E-02 | 7.818E-02 | 1.841E-02 | 3.551E-01 | 5.496E-02 | 1.505E-01 | 1.836E-02 | + |
| Osycka2 | 3.315E-01 | 1.936E-01 | 7.450E-01 | 2.215E-01 | 4.279E-01 | 1.189E-01 | 6.797E-01 | 1.508E-01 | 6.547E-01 | 1.272E-01 | + |
| Golinski | 1.390E-01 | 3.754E-02 | 2.969E-01 | 5.860E-02 | 1.406E-01 | 3.241E-02 | 5.282E-01 | 8.745E-02 | 6.958E-01 | 3.771E-02 | + |
| Tanaka | 7.181E-01 | 1.328E-01 | 3.550E-01 | 7.258E-02 | 3.049E-01 | 4.832E-02 | 3.677E-01 | 6.166E-02 | 3.151E-01 | 5.098E-02 | + |
| Binh2 | 3.917E-01 | 1.548E-02 | 3.079E-01 | 2.005E-02 | 3.207E-01 | 1.566E-02 | 5.412E-01 | 4.966E-02 | 5.559E-01 | 2.354E-02 | + |
| DTLZ1 | 3.716E-01 | 3.265E-01 | 5.746E-01 | 9.561E-02 | 6.711E-01 | 3.709E-01 | 5.310E-01 | 2.029E-01 | 1.442E+00 | 1.435E-01 | + |
| DTLZ2 | 4.347E-01 | 2.665E-02 | 3.681E-01 | 2.020E-02 | 5.574E-01 | 3.451E-02 | 5.132E-01 | 5.589E-02 | 9.251E-02 | 1.004E-02 | + |
| DTLZ3 | 8.279E-01 | 8.487E-02 | 8.075E-01 | 1.539E-01 | 9.466E-01 | 7.809E-02 | 8.738E-01 | 1.260E-01 | 1.049E+00 | 1.140E-01 | + |
| DTLZ4 | 6.322E-01 | 4.097E-01 | 9.949E-01 | 1.751E-01 | 5.913E-01 | 1.299E-01 | 5.023E-01 | 6.311E-02 | 9.455E-02 | 1.643E-02 | + |
| DTLZ5 | 7.285E-01 | 3.598E-02 | 2.370E-01 | 2.438E-02 | 1.675E-01 | 1.360E-02 | 4.432E-01 | 7.211E-02 | 2.289E-01 | 2.241E-02 | + |
| DTLZ6 | 7.370E-01 | 5.292E-02 | 1.553E-01 | 2.450E-02 | 8.030E-01 | 5.052E-02 | 6.503E-01 | 5.915E-02 | 6.746E-01 | 3.573E-02 | + |
| DTLZ7 | 6.991E-01 | 8.143E-02 | 3.926E-01 | 2.389E-02 | 5.956E-01 | 8.279E-02 | 5.019E-01 | 7.312E-02 | 3.332E-01 | 3.862E-02 | + |
| Viennet | 4.583E-01 | 2.513E-02 | 4.265E-01 | 1.924E-02 | 5.762E-01 | 3.939E-02 | 4.557E-01 | 6.339E-02 | 1.467E-01 | 7.221E-03 | + |
| Viennet2 | 5.055E-01 | 3.971E-02 | 4.723E-01 | 2.306E-02 | 4.462E-01 | 3.232E-02 | 4.531E-01 | 6.871E-02 | 2.151E-01 | 1.523E-02 | + |
| Viennet3 | 3.941E-01 | 7.292E-02 | 3.912E-01 | 2.567E-02 | 3.148E-01 | 2.019E-02 | 4.049E-01 | 5.117E-02 | 5.491E-01 | 1.962E-02 | + |
| Viennet4 | 5.634E-01 | 3.831E-02 | 5.080E-01 | 2.563E-02 | 6.808E-01 | 3.203E-02 | 5.481E-01 | 8.628E-02 | 3.298E-01 | 1.535E-02 | + |
| Tamaki | 5.466E-01 | 8.235E-02 | 4.933E-01 | 1.623E-02 | 6.885E-01 | 3.176E-02 | 6.898E-01 | 3.917E-02 | 3.004E-01 | 1.411E-02 | + |

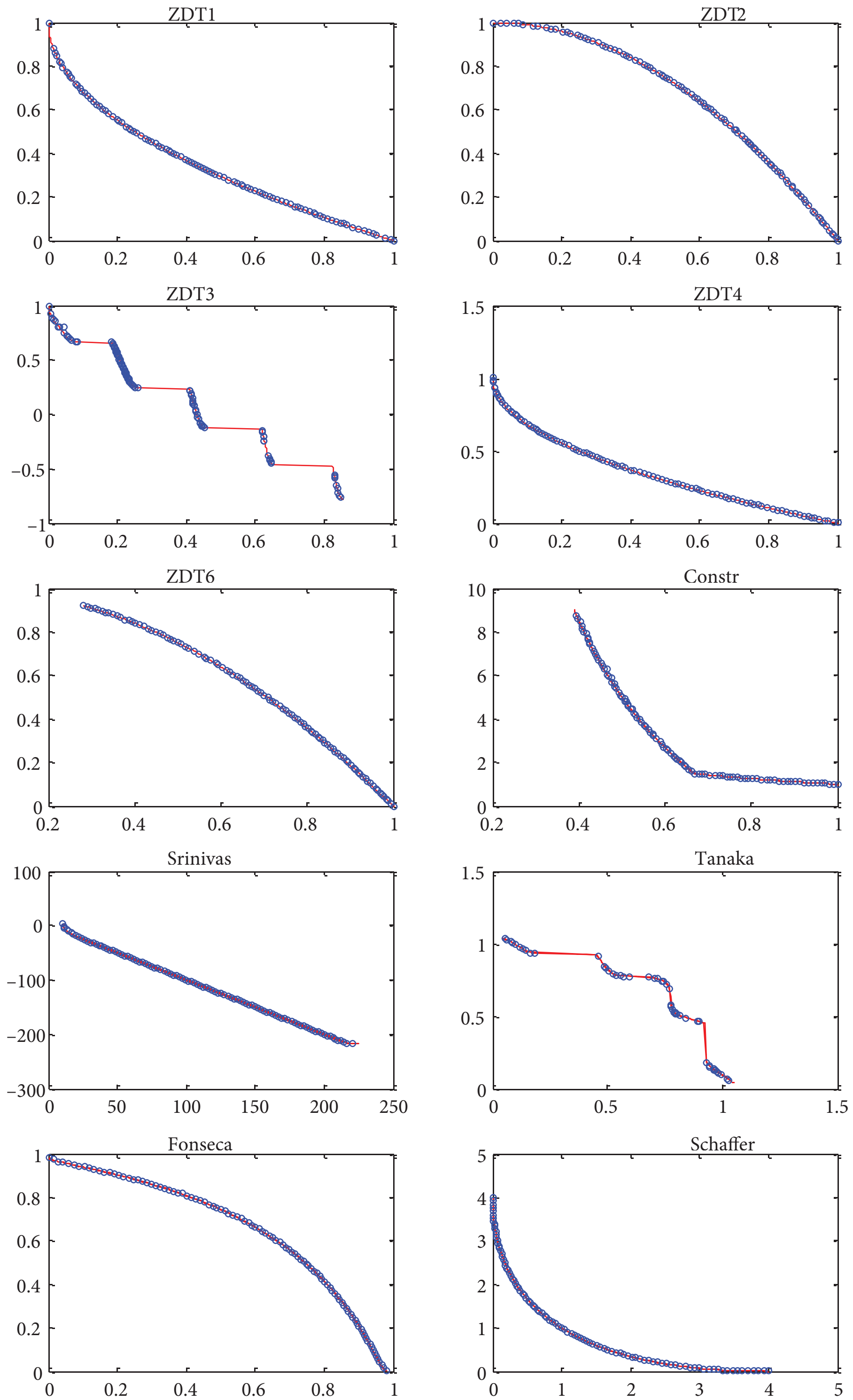


**Figure 8.** Pareto fronts obtained by IBMO on several biobjective MOPs.

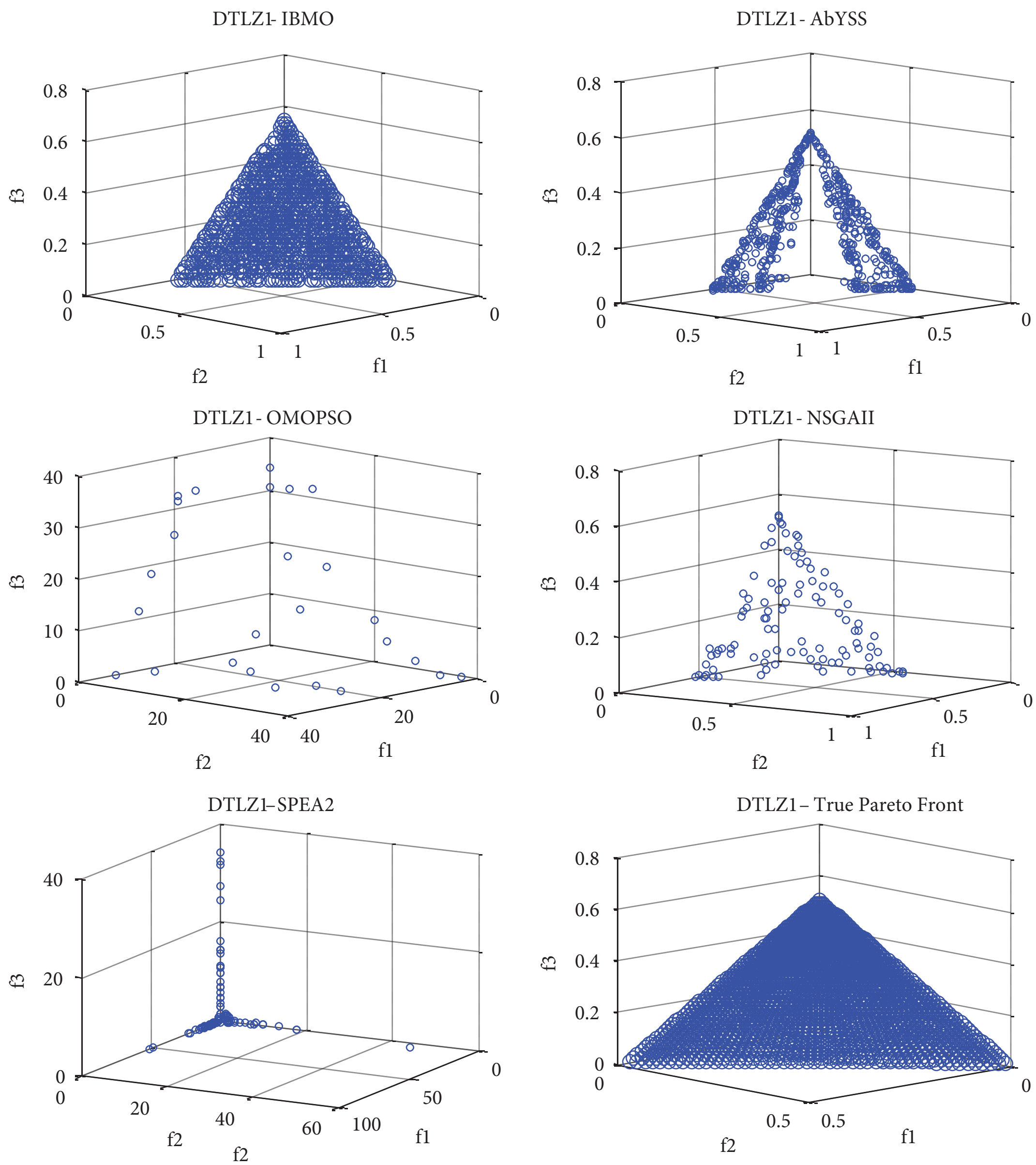


**Figure 9.** Nondominated solutions obtained by IBMO, AbySS, OMOPSO, NSGAII, and SPEA2 with true Pareto front on problem DTLZ1.

the calculated MS values, IBMO obtains extreme values for all MOPs except Binh2 and DTLZ6. However, all other algorithms do not adequately get close to one value either.

Hypervolume indicator encapsulates a spreadout measure of the obtained solutions along the Pareto front as well as the closeness of the solutions to the Pareto front. In other words, it tries to give a measure in a single value by combining the concepts of GD and SP. Based on the computational results of the HV metric shown in Table 5, IBMO obtains the best values for ten problems. All algorithms except OMOPSO cannot produce an acceptable front for DTLZ3 and DTLZ6 problems according to this metric. As to remaining MOPs, IBMO has the competitive outcomes.

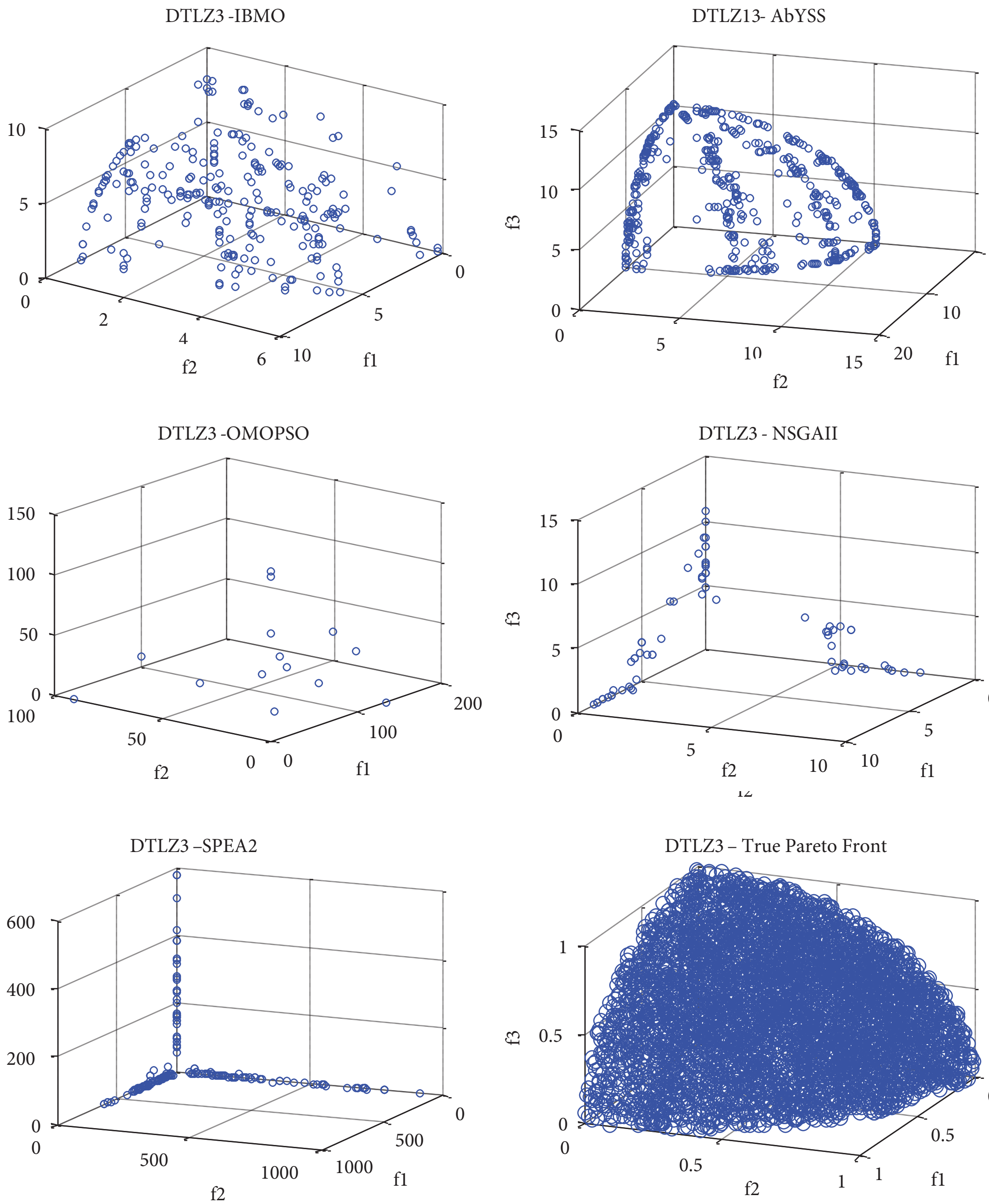


**Figure 10.** Nondominated solutions obtained by IBMO, AbySS, OMOPSO, NSGAII, and SPEA2 with true Pareto front on problem DTLZ3.

As aforementioned, results for the GD, SP, MS, and HV metrics after 50,000 function evaluations are given in Tables 6, 7, 8, and 9 in Appendix A, respectively. With regard to these outcomes, it can be reported again that IBMO has superior or competitive performance compared to the state-of-the-art techniques. At the end of the study in Appendix B, definitions of the test problems are given in Tables 10, 11, 12, and 13, respectively.

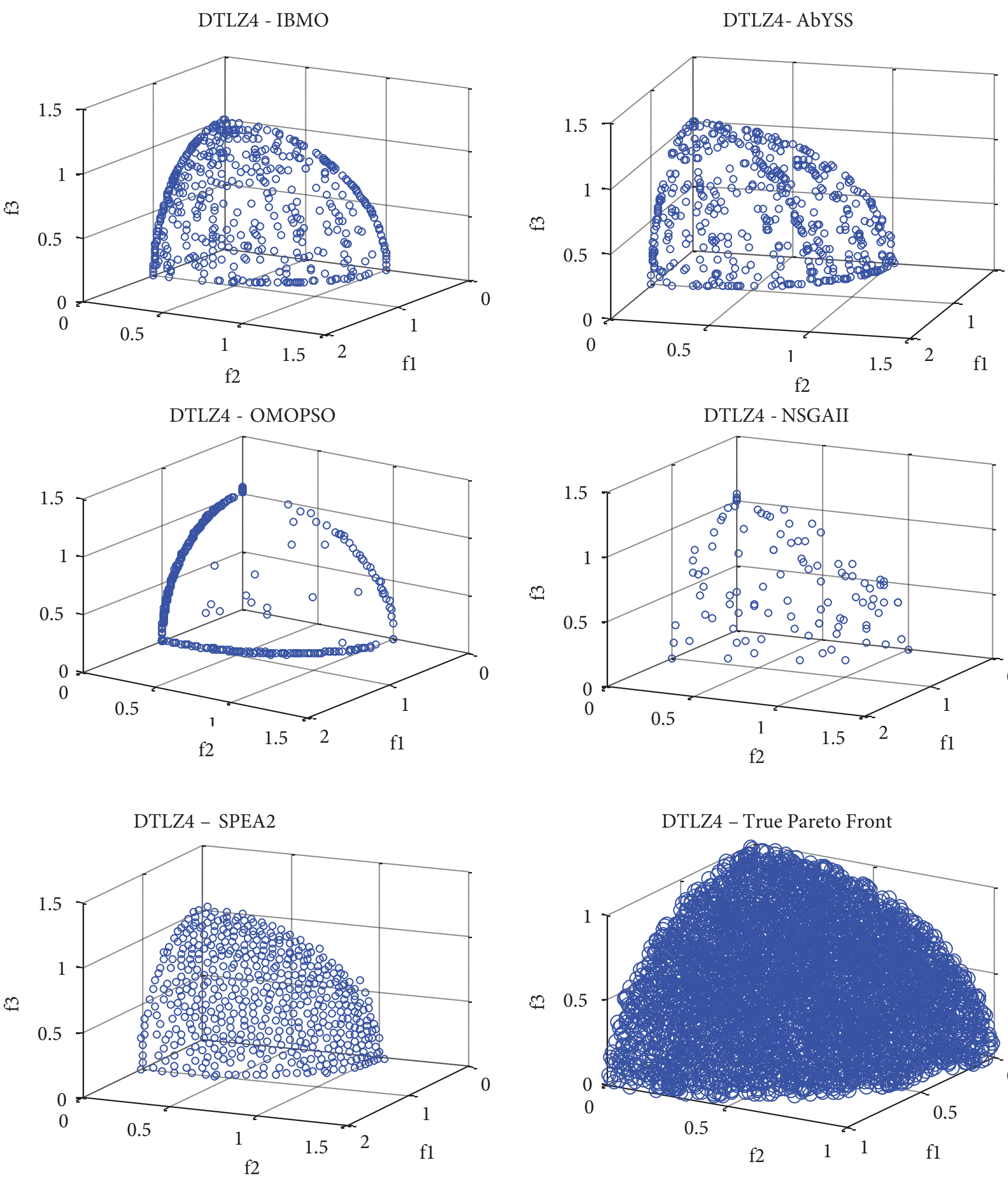


**Figure 11.** Nondominated solutions obtained by IBMO, AbySS, OMOPSO, NSGAII, and SPEA2 with true Pareto front on problem DTLZ4.

## 11. Conclusion

A new technique called the IBMO algorithm for solving multiobjective optimization problems is presented in this paper. ABC and the nondominated sorting strategy of NSGA2 are combined and a new improvement procedure is proposed to promote the convergence to the Pareto front in both phases of employed and onlooker bees. The crowding operator to improve the distribution of nondominated solutions and a fixed-sized external archive are used. Moreover, a diversification method is used to generate the initial population that disperses to search space in a better way.

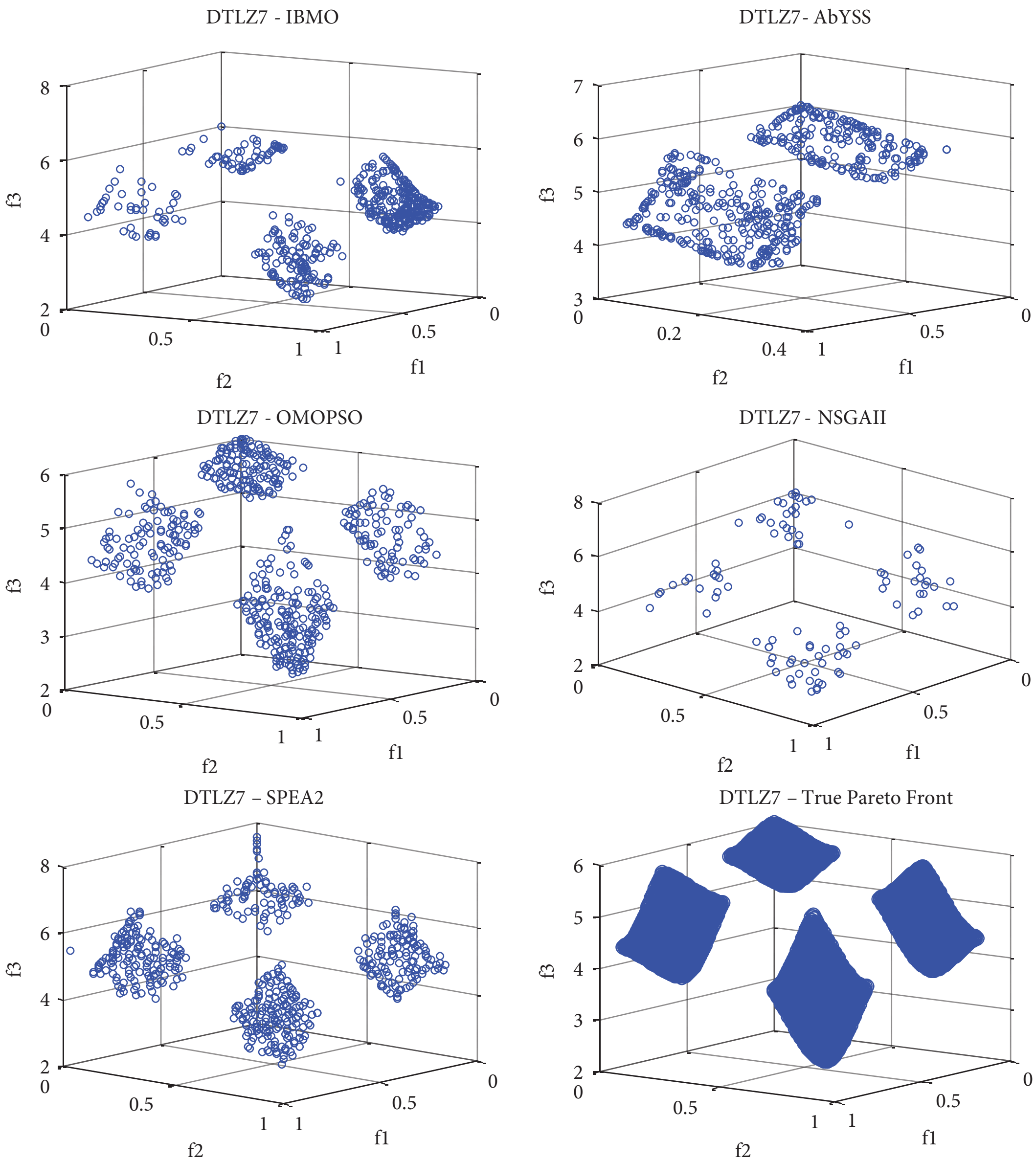


**Figure 12.** Nondominated solutions obtained by IBMO, AbySS, OMOPSO, NSGAII, and SPEA2 with true Pareto front on problem DTLZ7.

The proposed IBMO is compared with four state-of-the-art multiobjective optimization algorithms: NSGAII, SPEA2, AbYSS, and OMOPSO. In order to evaluate the performance of the algorithms, 26 test problems with two and three objectives are selected from the specialized literature. The dispersion of solutions and their proximity to the Pareto front are tested by using four different quality indicators.

IBMO is an effective and stable-running hybrid method compared to other algorithms. Its main advantages are its ease of use, few parameters that need to be adjusted, and well-constructed improvement procedure that ensures fast converging. According to experiments done with test functions, it can be concluded that IBMO has highly competitive performance and its results are better than the others for many of the test functions. IBMO is a suitable technique to solve MOPs as well as the best multiobjective approaches.

**Table 4.** Median and interquartile range of the maximum spread metric (25,000 function evaluations).

| Problem | IBMO | | OMOPSO | | AbYSS | | NSGA2 | | SPEA2 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | $\tilde{x}$ | *IQR* | $\tilde{x}$ | *IQR* | $\tilde{x}$ | *IQR* | $\tilde{x}$ | *IQR* | $\tilde{x}$ | *IQR* | |
| Schaffer | 1.000E+00 | 1.641E-04 | 1.000E+00 | 7.366E-05 | 1.000E+00 | 4.538E-05 | 8.711E-01 | 2.524E-01 | 8.550E-01 | 2.221E-01 | + |
| Fonseca | 9.995E-01 | 5.704E-04 | 9.999E-01 | 9.848E-05 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 1.306E-05 | 9.996E-01 | 4.810E-04 | + |
| Kursawe | 9.996E-01 | 4.685E-04 | 9.907E-01 | 6.361E-03 | 1.000E+00 | 3.424E-05 | 9.999E-01 | 1.327E-04 | 9.993E-01 | 1.218E-03 | + |
| ZDT1 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 8.511E-05 | 9.996E-01 | 1.900E-04 | 9.990E-01 | 5.165E-04 | + |
| ZDT2 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 6.945E-05 | 9.992E-01 | 3.186E-04 | 9.978E-01 | 9.446E-04 | – |
| ZDT3 | 9.973E-01 | 8.535E-03 | 9.984E-01 | 1.686E-03 | 9.998E-01 | 2.053E-01 | 9.997E-01 | 2.102E-04 | 9.993E-01 | 3.937E-04 | + |
| ZDT4 | 9.992E-01 | 1.045E-03 | 7.071E-01 | 0.000E+00 | 9.978E-01 | 2.769E-03 | 9.977E-01 | 2.147E-03 | 9.733E-01 | 9.728E-02 | + |
| ZDT6 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 9.994E-01 | 8.501E-05 | 9.933E-01 | 1.533E-03 | 9.863E-01 | 2.845E-03 | + |
| Constr | 9.035E-01 | 9.220E-02 | 9.965E-01 | 3.446E-03 | 9.990E-01 | 1.505E-03 | 9.974E-01 | 3.286E-03 | 9.960E-01 | 4.733E-03 | + |
| Srinivas | 9.826E-01 | 1.501E-02 | 9.867E-01 | 7.348E-03 | 9.921E-01 | 4.859E-03 | 9.926E-01 | 5.533E-03 | 9.921E-01 | 5.342E-03 | + |
| Osycka2 | 8.189E-01 | 2.010E-01 | 8.048E-01 | 2.954E-01 | 8.137E-01 | 6.668E-02 | 9.803E-01 | 1.872E-01 | 7.904E-01 | 2.699E-01 | + |
| Golinski | 9.624E-01 | 4.478E-02 | 9.914E-01 | 1.302E-02 | 9.889E-01 | 2.606E-02 | 9.783E-01 | 3.387E-02 | 9.515E-01 | 2.851E-02 | + |
| Tanaka | 9.632E-01 | 2.586E-02 | 9.914E-01 | 6.635E-03 | 9.981E-01 | 5.712E-04 | 9.979E-01 | 1.963E-03 | 9.958E-01 | 3.533E-03 | + |
| Binh2 | 7.694E-01 | 2.788E-04 | 8.111E-01 | 0.000E+00 | 8.111E-01 | 3.771E-10 | 8.111E-01 | 2.893E-10 | 8.111E-01 | 2.091E-08 | + |
| DTLZ1 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 6.661E-16 | + |
| DTLZ2 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 2.044E-12 | 1.000E+00 | 3.933E-11 | 1.000E+00 | 5.571E-06 | + |
| DTLZ3 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 1.110E-16 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 1.022E-10 | – |
| DTLZ4 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | + |
| DTLZ5 | 6.075E-01 | 1.673E-02 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 3.875E-14 | 1.000E+00 | 9.466E-10 | 1.000E+00 | 2.103E-05 | + |
| DTLZ6 | 8.165E-01 | 0.000E+00 | 1.000E+00 | 8.869E-09 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 3.614E-13 | 1.000E+00 | 2.674E-06 | + |
| DTLZ7 | 9.887E-01 | 5.866E-02 | 9.899E-01 | 8.378E-03 | 7.086E-01 | 2.929E-01 | 9.986E-01 | 6.368E-04 | 9.733E-01 | 7.641E-03 | + |
| Viennet | 9.993E-01 | 2.168E-03 | 1.000E+00 | 5.893E-04 | 1.000E+00 | 5.464E-06 | 1.000E+00 | 1.100E-04 | 9.984E-01 | 3.491E-03 | + |
| Viennet2 | 9.772E-01 | 4.864E-02 | 9.953E-01 | 9.259E-03 | 9.974E-01 | 5.015E-04 | 9.972E-01 | 2.093E-03 | 9.928E-01 | 1.142E-02 | + |
| Viennet3 | 9.953E-01 | 6.446E-03 | 9.994E-01 | 3.199E-03 | 1.000E+00 | 3.893E-05 | 9.999E-01 | 2.596E-04 | 9.985E-01 | 3.259E-03 | + |
| Viennet4 | 9.744E-01 | 2.198E-02 | 9.918E-01 | 7.614E-03 | 9.999E-01 | 6.201E-04 | 9.980E-01 | 6.219E-03 | 9.837E-01 | 1.966E-02 | + |
| Tamaki | 8.958E-01 | 3.791E-02 | 1.020E-02 | 0.000E+00 | 1.020E-02 | 1.388E-17 | 1.020E-02 | 3.253E-14 | 1.021E-02 | 3.985E-06 | + |

**Table 5.** Median and interquartile range of the hypervolume metric (25,000 function evaluations).

| Problem | IBMO | | OMOPSO | | AbYSS | | NSGA2 | | SPEA2 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | $\tilde{x}$ | *IQR* | $\tilde{x}$ | *IQR* | $\tilde{x}$ | *IQR* | $\tilde{x}$ | *IQR* | $\tilde{x}$ | *IQR* | |
| Schaffer | 8.299E-01 | 3.609E-05 | 8.299E-01 | 4.209E-05 | 8.299E-01 | 4.903E-05 | 5.732E-01 | 2.476E-01 | 5.850E-01 | 2.334E-01 | + |
| Fonseca | 3.119E-01 | 8.878E-05 | 3.124E-01 | 4.601E-05 | 3.104E-01 | 3.408E-04 | 3.080E-01 | 4.158E-04 | 3.107E-01 | 2.884E-04 | + |
| Kursawe | 4.015E-01 | 6.544E-05 | 3.972E-01 | 7.238E-04 | 4.011E-01 | 2.133E-04 | 3.996E-01 | 3.063E-04 | 4.008E-01 | 1.920E-04 | + |
| ZDT1 | 6.607E-01 | 7.884E-04 | 6.611E-01 | 4.267E-04 | 6.614E-01 | 2.955E-04 | 6.593E-01 | 4.637E-04 | 6.600E-01 | 3.819E-04 | + |
| ZDT2 | 3.282E-01 | 4.334E-04 | 3.280E-01 | 3.944E-04 | 3.281E-01 | 3.655E-04 | 3.261E-01 | 4.737E-04 | 3.263E-01 | 7.469E-04 | + |
| ZDT3 | 5.130E-01 | 2.224E-03 | 5.141E-01 | 1.359E-03 | 5.158E-01 | 3.462E-03 | 5.149E-01 | 2.017E-04 | 5.142E-01 | 2.924E-04 | + |
| ZDT4 | 6.583E-01 | 6.610E-03 | 0.000E+00 | 0.000E+00 | 6.546E-01 | 7.715E-03 | 6.552E-01 | 4.134E-03 | 6.529E-01 | 9.278E-03 | + |
| ZDT6 | 4.014E-01 | 5.487E-05 | 4.012E-01 | 9.551E-05 | 4.003E-01 | 1.927E-04 | 3.883E-01 | 1.898E-03 | 3.786E-01 | 3.455E-03 | + |
| Constr | 7.732E-01 | 6.442E-03 | 7.761E-01 | 2.201E-04 | 7.761E-01 | 2.015E-04 | 7.743E-01 | 3.807E-04 | 7.756E-01 | 3.335E-04 | + |
| Srinivas | 5.408E-01 | 9.853E-05 | 5.408E-01 | 1.006E-04 | 5.407E-01 | 1.317E-04 | 5.380E-01 | 5.511E-04 | 5.400E-01 | 2.170E-04 | + |
| Osycka2 | 6.976E-01 | 4.397E-02 | 7.054E-01 | 1.200E-02 | 7.437E-01 | 2.542E-01 | 7.505E-01 | 7.583E-03 | 7.358E-01 | 3.578E-02 | + |
| Golinski | 9.696E-01 | 8.209E-05 | 9.674E-01 | 5.688E-04 | 9.693E-01 | 1.814E-04 | 9.690E-01 | 1.683E-04 | 9.672E-01 | 7.323E-04 | – |
| Tanaka | 3.032E-01 | 2.098E-03 | 3.056E-01 | 9.505E-04 | 3.075E-01 | 5.612E-04 | 3.077E-01 | 5.319E-04 | 3.085E-01 | 6.258E-04 | + |
| Binh2 | 7.990E-01 | 4.865E-05 | 8.007E-01 | 2.799E-05 | 8.006E-01 | 5.954E-05 | 7.994E-01 | 2.805E-04 | 7.999E-01 | 1.779E-04 | + |
| DTLZ1 | 5.409E-01 | 5.284E-01 | 0.000E+00 | 0.000E+00 | 7.436E-01 | 7.070E-01 | 7.274E-01 | 4.474E-02 | 0.000E+00 | 0.000E+00 | + |
| DTLZ2 | 4.335E-01 | 3.171E-03 | 4.165E-01 | 2.724E-03 | 4.333E-01 | 3.078E-03 | 3.751E-01 | 8.087E-03 | 4.390E-01 | 1.344E-03 | + |
| DTLZ3 | 0.000E+00 | 0.000E+00 | 0.000E+00 | 0.000E+00 | 0.000E+00 | 0.000E+00 | 0.000E+00 | 0.000E+00 | 0.000E+00 | 0.000E+00 | + |
| DTLZ4 | 4.377E-01 | 1.748E-02 | 3.850E-01 | 2.435E-02 | 4.357E-01 | 6.297E-03 | 3.768E-01 | 5.934E-03 | 4.354E-01 | 2.643E-03 | + |
| DTLZ5 | 1.081E-02 | 1.436E-02 | 9.470E-02 | 1.252E-04 | 9.533E-02 | 1.685E-05 | 9.222E-02 | 2.360E-04 | 9.387E-02 | 1.987E-04 | + |
| DTLZ6 | 0.000E+00 | 0.000E+00 | 9.537E-02 | 9.837E-06 | 0.000E+00 | 0.000E+00 | 0.000E+00 | 0.000E+00 | 0.000E+00 | 0.000E+00 | + |
| DTLZ7 | 3.142E-01 | 2.147E-02 | 2.937E-01 | 1.262E-02 | 2.709E-01 | 4.654E-02 | 2.804E-01 | 5.503E-03 | 2.388E-01 | 1.595E-02 | + |
| Viennet | 7.371E-01 | 5.807E-04 | 7.376E-01 | 3.496E-04 | 7.364E-01 | 5.482E-04 | 7.115E-01 | 1.959E-03 | 7.402E-01 | 1.337E-04 | + |
| Viennet2 | 9.293E-01 | 3.400E-04 | 9.295E-01 | 1.714E-04 | 9.287E-01 | 3.409E-04 | 9.200E-01 | 1.734E-03 | 9.300E-01 | 7.562E-05 | + |
| Viennet3 | 8.402E-01 | 1.648E-04 | 8.398E-01 | 1.061E-04 | 8.398E-01 | 1.306E-04 | 8.326E-01 | 1.005E-03 | 8.374E-01 | 2.148E-04 | + |
| Viennet4 | 8.675E-01 | 1.067E-03 | 8.698E-01 | 5.668E-04 | 8.683E-01 | 1.421E-03 | 8.554E-01 | 3.138E-03 | 8.718E-01 | 6.207E-04 | + |
| Tamaki | 3.213E-01 | 1.370E-02 | 4.184E-01 | 2.053E-03 | 4.139E-01 | 3.245E-03 | 3.572E-01 | 8.966E-03 | 4.318E-01 | 8.715E-04 | + |

## References


[1] Osyczka A. Evolutionary Algorithms for Single and Multicriteria Design Optimization. New York, NY, USA: Physica Verlag, 2002.

[2] Deb K. Multi-Objective Optimization Using Evolutionary Algorithms: An Introduction. KanGAL Report Number 2011003, 2011.

[3] Sağ T, Çunkaş M. A tool for multiobjective evolutionary algorithms. Adv Eng Softw 2009; 40: 902-912.

[4] Yahia H, Liouane N, Dhifaoui R. Multiobjective differential evolution-based performance optimization for switched reluctance motor drives. Turk J Elec Eng & Comp Sci 2013; 21: 1061-1076.

[5] Koodalsamy C, Simon S. Fuzzified artificial bee colony algorithm for nonsmooth and nonconvex multiobjective economic dispatch problem. Turk J Elec Eng & Comp Sci 2013; 21: 1995-2014.

[6] Schaffer JD. Multiple objective optimization with vector evaluated genetic algorithms. In: 1st International Conference on Genetic Algorithms 1985; pp. 93-100.

[7] Zhoua A, Qub B, Lic H, Zhaob S, Suganthanb PN, Zhangd Q. Multiobjective evolutionary algorithms: a survey of the state of the art. Swarm and Evolutionary Computation 2011; 1: 32-49.

[8] Coello Coello CA, Lamont GB (eds). Applications of Multi-Objective Evolutionary Algorithms. Singapore: World Scientific, 2004.

[9] Horn J, Nafpliotis N, Goldberg DE. A niched Pareto genetic algorithm for multiobjective optimization. In: Proc 1. IEEE Conf Evol Comp, 1994; vol. 1, pp. 82-87.

[10] Srinivas N, Deb K. Multiobjective function optimization using nondominated sorting genetic algorithms. IEEE T Evolut Comput 1995; 2: 221-248.

[11] Deb K, Agrawal S, Pratap A, Meyarivan T. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE T Evolut Comput 2002; 6: 182-197.

[12] Knowles JD, Corne DW. Approximating the nondominated front using the Pareto archived evolution strategy. IEEE T Evolut Comput 2000; 8: 149-172.

[13] Zitzler E, Thiele L. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. IEEE T Evolut Comput 1999; 3: 257-271.

[14] Zitzler E, Laumanns M, Thiele L. SPEA2: improving the strength Pareto evolutionary algorithm. Comput Eng Networks Lab, Swiss Fed Inst Technol, Zurich, Switzerland, Tech Rep 103, 2001.

[15] Kennedy J, Eberhart R. Particle swarm optimization. In: Fourth IEEE International Conference on Neural Networks 1995; pp: 1942-1948.

[16] Dorigo M, Stützle T. Ant Colony Optimization. Cambridge, MA, USA: MIT Press, 2004.

[17] Karaboga D. An idea based on honey bee swarm for numerical optimization, Erciyes University, Engineering Faculty, Computer Eng Department, Technical Report, 2005.

[18] Angus D. Crowding population-based ant colony optimization for the multi-objective travelling salesman problem. In: IEEE Symposium on Computational Intelligence in Multicriteria Decision Making, MCDM 2007, 2007, pp. 333-340.

[19] Chitty DM, Hernandez ML. A hybrid ant colony optimization technique for dynamic vehicle routing. In: Conference on Genetic and Evolutionary Computation, GECCO 2004. In: LNCS, vol. 3102, 2004, pp. 48-59.

[20] Pasia JM, Hartl RF, Doerner KF. Solving a bi-objective flowshop scheduling problem by Pareto-ant colony optimization. In: 5th International Workshop on Ant Colony Optimization and Swarm Intelligence, ANTS 2006, 2006, pp. 294-305.

[21] Doerner K, Gutjahr WJ, Hartl RF, Strauss C, Stummer C. Pareto ant colony optimization: a metaheuristic approach to multiobjective portfolio selection. Ann Oper Res 2004; 31: 77-79.

[22] Reyes MS, Coello C. Multi-objective particle swarm optimizers: a survey of the state-of-the-art. Int J Comput Int Res 2006; 2: 287-308.

[23] Moore J, Chapman R. Application of particle swarm to multiobjective optimization. Technical report, Department of Computer Science and Software Engineering, Auburn University, 1999.

[24] Li X. A non-dominated sorting particle swarm optimizer for multiobjective optimization. Lect Notes Comput Sc 2003; 2723: 37-48.

[25] Sierra MS, Coello CA. Improving PSO-Based Multi-objective Optimization Using Crowding, Mutation and o-Dominance. In: Evolutionary Multi-Criterion Optimization (EMO 2005); LNCS 3410, pp: 505-519.

[26] Durillo JJ, Nieto JG, Nebro AJ, Coello CA, Luna F, Alba E. Multi-objective particle swarm optimizers: an experimental comparison. In: 5th International Conference on Evolutionary Multi-Criterion Optimization 2009; vol: 5467, pp. 495-509.

[27] Nebro AJ, Durillo JJ, Nieto G, Coello CC, Luna F, Alba E. SMPSO: a new PSO-based metaheuristic for multi-objective optimization. In: IEEE Symposium on Computational Intelligence in Multicriteria Decision-Making 2009; 1: 66-73.

[28] Molina J, Laguna M, Martí R, Caballero R. SSPMO: A scatter tabu search procedure for non-linear multiobjective optimization. Informs J Comp 2007; 19: 91-100.

[29] Nebro AJ, Luna F, Alba E, Dorronsoro B, Durillo JJ, Beham A. AbYSS: adapting scatter search to multiobjective optimization. IEEE T Evol Comput 2008; 12: 439-457.

[30] Tang L, Wang X. A hybrid multiobjective evolutionary algorithm for multiobjective optimization problems. IEEE T Evolut Comput 2013; 17: 20-45.

[31] Basturk B, Karaboga D. An Artificial Bee Colony (ABC) algorithm for numeric function optimization. In: IEEE International Conference on Neural Networks. IEEE Swarm Intelligence Symposium May 2006; Indianapolis, Indiana, USA.

[32] Karaboga D, Basturk B. A powerful and efficient algorithm for numerical function optimization: artificial bee colony algorithm. J Global Optim 2007; 39: 459-471.

[33] Omkar SN, Senthilnath J, Khandelwal R, Narayana NG, Gopalakrishnan S. Artificial Bee Colony (ABC) for multi-objective design optimization of composite structures. Appl Soft Comput 2011; 11: 489-499.

[34] Hedayatzadeh R, Hasanizadeh B, Akbari R, Ziarati K. A multi-objective artificial bee colony for optimizing multi-objective problems. In: 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE) Aug 2010; Chengdu, China, pp. 277-281.

[35] Zou W, Zhu Y, Chen H, Zhang B. Solving multiobjective optimization problems using artificial bee colony algorithm. Discrete Dyn Nat Soc 2011; 1-37.

[36] Zhang H, Zhu Y, Yan X. Multi-hive artificial bee colony algorithm for constrained multi-objective optimization. In: IEEE Congress on Evolutionary Computation 2012; pp: 1-8.

[37] Largo RA, Gonzalez-Alvarez DL, Vega-Rodriguez MA, Gomez-Pulido JA, Sanchez-Perez JM. In: CINTI2012 13th IEEE International Symposium on Computational Intelligence and Informatics; 20–22 November 2012, Budapest, Hungary.

[38] Ríos MA, Vega-Rodríguez MA, Castrillo FP. Meta-schedulers for grid computing based on multi-objective swarm algorithms. Appl Soft Comp 2013; 13: 1567-1582.

[39] Glover F, Laguna M, Martí R. Scatter search. In: Ghosh A, Tsutsui S, editors. Advances in Evolutionary Computing: Theory and Applications. Berlin, Germany: Springer-Verlag, 2003; pp. 519-537.

[40] Fonseca C, Flemming P. Multiobjective optimization and multiple constraint handling with evolutionary algorithms - part II: Application example. IEEE T Syst Man Cyb 1998; 28: 38-47.

[41] Kursawe F. A variant of evolution strategies for vector optimization. In: Schwefel H, Manner R, editors. Parallel Problem Solving for Nature, Berlin, Germany: Springer-Verlag, 1990; pp. 193-197.

[42] Zitzler E, Deb K, Thiele L. Comparison of multiobjective evolutionary algorithms: empirical results. IEEE T Evolut Comput 2000; 8: 173-195.

[43] Osyczka A, Kundo S. A new method to solve generalized multicriteria optimization problems using a simple genetic algorithm. Struct Optimization 1995; 10: 94-99.

[44] Kurpati A, Azarm S, Wu J. Constraint handling improvements for multi-objective genetic algorithms. Struct Multidiscip O 2002; 23: 204-213.

[45] Tanaka M, Watanabe H, Furukawa Y, Tanino T. GA-based decision support system for multicriteria optimization. In: IEEE International Conference on Systems, Man, and Cybernetics 1995; vol. 2, pp. 1556-1561.

[46] Binh TT, Ulrich K. Multiobjective evolution strategy with linear and nonlinear constraints. In: 15th IMACS World Congr Sci Comp Modeling Appl Math, 1997, pp. 357-362.

[47] Deb K, Thiele L, Laumanns M, Zitzler E. Scalable Test Problems for Evolutionary Multi-Objective Optimization. Kanpur, India: Kanpur Genetic Algorithms Lab. (KanGAL), Indian Inst Technol 2001; KanGAL Report 2 001 001.

[48] Viennet R, Fontiex C, Marc I. Multicriteria optimization using a genetic algorithm for determining a Pareto set. J Syst Sci 1996; 27: 255-260.

[49] Tamaki H, Kita H, Kobayashi S. Multiobjective optimization by genetic algorithms: A review. In: 3rd IEEE Conf Evol Comput May 1996; pp. 517-522.

[50] Zitzler E, Thiele L, Laumanns M, Fonseca CM, Fonseca VG. Performance assessment of multiobjective optimizers: an analysis and review. IEEE T Evolut Comput 2003; 7: 117-132.

[51] Rendon MV, Uresti-charre E. A non-generational genetic algorithm for multiobjective optimization. In: Seventh International Conference on Genetic Algorithms 1997; pp. 658-665.

[52] Tsou CS, Fang HH, Chang HH, Kao CH. An improved particle swarm Pareto optimizer with local search and clustering. In: SEAL: 6th Int Conf Simul Evol Learning, 2006; vol. LNCS 4247, pp. 400-407.

[53] Zitzler E, Thiele L. Multiobjective optimization using evolutionary algorithms—a comparative case study. In: the 5th International Conference on Parallel Problem Solving from Nature –PPSN V, vol: 1498 of Lecture Notes in Computer Science 1998; pp: 292-301. Springer, Berlin.

## Appendix A

**Table 6.** Median and interquartile range of the general distance metric (50,000 function evaluations).

| Problem | IBMO | | OMOPSO | | AbYSS | | NSGA2 | | SPEA2 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | $\tilde{x}$ | $IQR$ | $\tilde{x}$ | $IQR$ | $\tilde{x}$ | $IQR$ | $\tilde{x}$ | $IQR$ | $\tilde{x}$ | $IQR$ | |
| Schaffer | 2.255E-04 | 2.510E-05 | 2.260E-04 | 2.647E-05 | 2.337E-04 | 1.858E-05 | 1.691E-02 | 4.640E-02 | 1.507E-02 | 4.364E-02 | + |
| Fonseca | 1.281E-04 | 9.087E-06 | 1.212E-04 | 8.787E-06 | 1.924E-04 | 2.555E-05 | 3.475E-04 | 3.970E-05 | 2.164E-04 | 2.757E-05 | + |
| Kursawe | 1.288E-04 | 1.109E-05 | 4.039E-04 | 7.752E-05 | 1.381E-04 | 1.234E-05 | 2.081E-04 | 3.264E-05 | 1.533E-04 | 1.815E-05 | + |
| ZDT1 | 6.561E-05 | 2.787E-05 | 1.256E-04 | 3.455E-05 | 1.546E-04 | 2.318E-05 | 1.923E-04 | 4.928E-05 | 1.730E-04 | 1.745E-05 | + |
| ZDT2 | 4.658E-05 | 2.893E-06 | 5.487E-05 | 7.645E-06 | 5.324E-05 | 1.307E-05 | 1.294E-04 | 3.862E-05 | 6.739E-05 | 2.746E-05 | + |
| ZDT3 | 1.592E-04 | 2.342E-05 | 1.943E-04 | 1.548E-05 | 1.985E-04 | 1.644E-05 | 2.102E-04 | 1.299E-05 | 2.147E-04 | 1.542E-05 | + |
| ZDT4 | 7.272E-05 | 3.019E-05 | 5.651E-01 | 4.885E-01 | 2.435E-04 | 1.333E-04 | 1.759E-04 | 7.532E-05 | 1.882E-04 | 5.833E-05 | + |
| ZDT6 | 1.497E-04 | 1.112E-05 | 1.732E-02 | 3.782E-02 | 1.633E-04 | 1.196E-05 | 1.597E-04 | 1.841E-05 | 1.672E-04 | 1.703E-05 | + |
| Constr | 1.428E-04 | 2.436E-05 | 1.315E-04 | 1.688E-05 | 1.797E-04 | 3.062E-05 | 2.987E-04 | 4.231E-05 | 1.957E-04 | 2.215E-05 | + |
| Srinivas | 4.299E-05 | 1.893E-05 | 4.419E-05 | 1.624E-05 | 4.571E-05 | 1.517E-05 | 1.948E-04 | 4.105E-05 | 1.140E-04 | 2.731E-05 | + |
| Osycka2 | 4.738E-03 | 3.496E-03 | 1.819E-03 | 2.287E-03 | 1.138E-03 | 1.697E-02 | 1.014E-03 | 7.722E-05 | 1.344E-03 | 1.540E-04 | + |
| Golinski | 3.121E-04 | 2.913E-05 | 3.524E-04 | 5.040E-05 | 3.081E-04 | 2.423E-05 | 3.262E-04 | 3.013E-05 | 2.256E-04 | 2.109E-05 | + |
| Tanaka | 6.653E-04 | 3.983E-04 | 7.743E-04 | 9.179E-05 | 7.813E-04 | 6.853E-05 | 8.115E-04 | 9.049E-05 | 7.159E-04 | 8.757E-05 | + |
| Binh2 | 6.732E-04 | 2.530E-05 | 1.253E-03 | 3.337E-05 | 1.251E-03 | 3.795E-05 | 1.386E-03 | 7.615E-05 | 1.586E-03 | 5.363E-05 | + |
| DTLZ1 | 2.668E-03 | 1.987E-03 | 9.276E+00 | 1.354E+00 | 5.503E-04 | 6.508E-05 | 1.343E-03 | 1.747E-04 | 4.786E-01 | 9.615E-01 | + |
| DTLZ2 | 1.644E-04 | 1.613E-05 | 6.973E-04 | 6.129E-05 | 1.604E-04 | 1.396E-05 | 1.218E-03 | 2.494E-04 | 3.034E-04 | 4.006E-05 | + |
| DTLZ3 | 5.964E-02 | 8.691E-02 | 3.438E+01 | 6.029E+00 | 4.176E-03 | 7.461E-02 | 6.292E-03 | 1.849E-02 | 8.352E+00 | 2.948E+00 | + |
| DTLZ4 | 6.342E-04 | 6.125E-05 | 1.236E-03 | 1.637E-04 | 6.207E-04 | 1.477E-05 | 2.049E-03 | 2.050E-04 | 7.651E-04 | 5.406E-05 | + |
| DTLZ5 | 1.110E-01 | 2.428E-03 | 2.524E-04 | 1.439E-06 | 2.517E-04 | 1.273E-06 | 6.114E-04 | 4.204E-05 | 2.502E-04 | 2.646E-06 | + |
| DTLZ6 | 4.292E-01 | 1.117E-02 | 2.526E-04 | 1.098E-06 | 4.670E-03 | 2.560E-03 | 2.005E-03 | 2.369E-03 | 1.336E-01 | 8.580E-03 | + |
| DTLZ7 | 4.267E-04 | 1.201E-04 | 1.066E-03 | 1.537E-04 | 4.060E-04 | 2.327E-04 | 2.665E-03 | 5.604E-04 | 8.379E-04 | 9.706E-05 | + |
| Viennet1 | 4.342E-04 | 7.385E-05 | 6.809E-04 | 8.280E-05 | 4.957E-04 | 6.756E-05 | 3.292E-03 | 7.982E-04 | 8.980E-04 | 7.123E-05 | + |
| Viennet2 | 1.270E-04 | 3.828E-05 | 1.483E-04 | 3.264E-05 | 1.713E-04 | 6.093E-05 | 7.205E-04 | 3.641E-04 | 1.435E-04 | 1.448E-05 | + |
| Viennet3 | 3.160E-05 | 1.049E-05 | 3.115E-05 | 5.285E-06 | 3.243E-05 | 5.606E-06 | 2.380E-04 | 5.885E-05 | 4.171E-05 | 6.207E-06 | + |
| Viennet4 | 1.003E-04 | 2.006E-05 | 9.072E-05 | 8.155E-06 | 8.952E-05 | 1.037E-05 | 4.709E-04 | 1.287E-04 | 1.078E-04 | 1.273E-05 | + |
| Tamaki | 1.428E-03 | 1.997E-04 | 7.394E-02 | 1.414E-04 | 7.287E-02 | 3.151E-04 | 1.642E-01 | 1.112E-03 | 7.333E-02 | 1.510E-04 | + |

**Table 7.** Median and interquartile range of the spread metric (50,000 function evaluations).

| Problem | IBMO | | OMOPSO | | AbYSS | | NSGA2 | | SPEA2 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | $\tilde{x}$ | $IQR$ | $\tilde{x}$ | $IQR$ | $\tilde{x}$ | $IQR$ | $\tilde{x}$ | $IQR$ | $\tilde{x}$ | $IQR$ | |
| Schaffer | 1.112E-01 | 2.865E-02 | 1.084E-01 | 2.437E-02 | 1.162E-01 | 3.507E-02 | 7.611E-01 | 3.435E-01 | 7.470E-01 | 2.923E-01 | + |
| Fonseca | 6.211E-02 | 1.822E-02 | 5.457E-02 | 1.488E-02 | 8.698E-02 | 1.607E-02 | 3.741E-01 | 6.559E-02 | 1.242E-01 | 1.994E-02 | + |
| Kursawe | 2.242E-01 | 5.995E-03 | 2.731E-01 | 3.417E-02 | 2.302E-01 | 6.644E-03 | 5.437E-01 | 9.784E-02 | 2.890E-01 | 1.449E-02 | + |
| ZDT1 | 1.541E-01 | 2.121E-02 | 7.763E-02 | 1.677E-02 | 8.170E-02 | 1.823E-02 | 3.871E-01 | 5.823E-02 | 1.275E-01 | 1.819E-02 | + |
| ZDT2 | 1.364E-01 | 1.605E-02 | 7.356E-02 | 1.858E-02 | 8.592E-02 | 1.920E-02 | 4.074E-01 | 7.563E-02 | 1.263E-01 | 1.992E-02 | + |
| ZDT3 | 2.821E-01 | 8.374E-02 | 1.207E-01 | 1.926E-02 | 1.104E-01 | 2.169E-01 | 3.844E-01 | 6.655E-02 | 1.578E-01 | 2.362E-02 | + |
| ZDT4 | 2.392E-01 | 5.043E-02 | 8.461E-01 | 1.319E-01 | 8.899E-02 | 2.553E-02 | 4.165E-01 | 7.331E-02 | 1.085E-01 | 1.859E-02 | + |
| ZDT6 | 8.128E-02 | 1.263E-02 | 8.824E-01 | 1.224E+00 | 6.469E-02 | 1.365E-02 | 4.122E-01 | 7.221E-02 | 1.217E-01 | 1.856E-02 | + |
| Constr | 2.498E-01 | 9.214E-02 | 9.666E-02 | 2.041E-02 | 1.431E-01 | 1.978E-02 | 3.950E-01 | 6.017E-02 | 5.236E-01 | 3.794E-02 | + |
| Srinivas | 7.456E-02 | 2.183E-02 | 7.069E-02 | 1.542E-02 | 6.973E-02 | 1.379E-02 | 3.591E-01 | 5.079E-02 | 1.482E-01 | 1.431E-02 | + |
| Osycka2 | 3.123E-01 | 2.033E-01 | 6.587E-01 | 1.784E-01 | 3.757E-01 | 2.797E-01 | 6.445E-01 | 1.243E-01 | 5.970E-01 | 1.298E-01 | + |
| Golinski | 1.185E-01 | 3.877E-02 | 2.270E-01 | 4.492E-02 | 9.871E-02 | 1.913E-02 | 5.252E-01 | 9.539E-02 | 6.926E-01 | 3.967E-02 | + |
| Tanaka | 7.426E-01 | 1.695E-01 | 2.636E-01 | 4.739E-02 | 2.403E-01 | 4.495E-02 | 3.566E-01 | 5.173E-02 | 2.327E-01 | 4.012E-02 | + |
| Binh2 | 3.888E-01 | 1.501E-02 | 3.122E-01 | 1.909E-02 | 3.127E-01 | 2.107E-02 | 5.415E-01 | 4.507E-02 | 5.531E-01 | 2.457E-02 | + |
| DTLZ1 | 1.805E-01 | 9.150E-02 | 5.371E-01 | 1.082E-01 | 5.536E-01 | 3.115E-02 | 4.939E-01 | 6.106E-02 | 1.707E+00 | 4.271E-01 | + |
| DTLZ2 | 4.207E-01 | 2.891E-02 | 3.730E-01 | 2.184E-02 | 5.522E-01 | 2.909E-02 | 5.134E-01 | 5.199E-02 | 8.150E-02 | 7.548E-03 | + |
| DTLZ3 | 8.015E-01 | 2.051E-01 | 7.842E-01 | 1.111E-01 | 7.275E-01 | 2.428E-01 | 5.800E-01 | 9.881E-02 | 1.324E+00 | 1.728E-01 | + |
| DTLZ4 | 4.173E-01 | 6.182E-02 | 9.295E-01 | 4.138E-01 | 5.121E-01 | 3.297E-02 | 4.972E-01 | 6.418E-02 | 8.120E-02 | 5.898E-03 | + |
| DTLZ5 | 6.926E-01 | 3.824E-02 | 1.511E-01 | 1.029E-02 | 1.329E-01 | 1.080E-02 | 4.323E-01 | 6.491E-02 | 1.918E-01 | 1.329E-02 | + |
| DTLZ6 | 7.228E-01 | 3.582E-02 | 1.128E-01 | 6.332E-03 | 4.917E-01 | 9.090E-02 | 4.917E-01 | 1.105E-01 | 6.442E-01 | 3.783E-02 | + |
| DTLZ7 | 5.095E-01 | 6.265E-02 | 4.007E-01 | 2.239E-02 | 5.866E-01 | 1.379E-01 | 4.792E-01 | 5.570E-02 | 2.461E-01 | 1.255E-02 | + |
| Viennet1 | 4.645E-01 | 2.496E-02 | 4.305E-01 | 2.705E-02 | 5.823E-01 | 2.991E-02 | 4.593E-01 | 6.091E-02 | 1.465E-01 | 7.754E-03 | + |
| Viennet2 | 4.517E-01 | 3.407E-02 | 4.537E-01 | 2.457E-02 | 4.302E-01 | 3.194E-02 | 4.711E-01 | 7.230E-02 | 2.003E-01 | 1.429E-02 | + |
| Viennet3 | 3.325E-01 | 3.821E-02 | 3.447E-01 | 2.059E-02 | 2.942E-01 | 2.409E-02 | 4.124E-01 | 6.276E-02 | 5.602E-01 | 1.484E-02 | + |
| Viennet4 | 5.851E-01 | 2.937E-02 | 5.462E-01 | 2.624E-02 | 6.703E-01 | 2.801E-02 | 5.288E-01 | 7.945E-02 | 3.281E-01 | 1.800E-02 | + |
| Tamaki | 5.607E-01 | 1.087E-01 | 4.894E-01 | 2.154E-02 | 6.517E-01 | 2.574E-02 | 7.064E-01 | 4.212E-02 | 2.728E-01 | 9.564E-03 | + |

**Table 8.** Median and interquartile range of the maximum spread metric (50,000 function evaluations).

| Problem | IBMO | | OMOPSO | | AbYSS | | NSGA2 | | SPEA2 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | $\tilde{x}$ | *IQR* | $\tilde{x}$ | *IQR* | $\tilde{x}$ | *IQR* | $\tilde{x}$ | *IQR* | $\tilde{x}$ | *IQR* | |
| Schaffer | 1.000E+00 | 7.837E-05 | 1.000E+00 | 4.499E-05 | 1.000E+00 | 1.824E-05 | 9.442E-01 | 9.164E-02 | 9.291E-01 | 1.092E-01 | + |
| Fonseca | 1.000E+00 | 2.488E-04 | 1.000E+00 | 4.106E-05 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 1.081E-04 | + |
| Kursawe | 9.998E-01 | 2.483E-04 | 9.926E-01 | 4.550E-03 | 1.000E+00 | 1.311E-05 | 1.000E+00 | 4.358E-05 | 9.999E-01 | 1.007E-04 | + |
| ZDT1 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 1.274E-06 | 1.000E+00 | 2.251E-05 | 9.999E-01 | 1.381E-04 | + |
| ZDT2 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 8.286E-07 | 1.000E+00 | 1.390E-05 | 9.999E-01 | 5.808E-05 | + |
| ZDT3 | 9.999E-01 | 2.464E-03 | 9.995E-01 | 6.975E-04 | 9.999E-01 | 2.053E-01 | 9.999E-01 | 5.727E-05 | 9.998E-01 | 1.744E-04 | + |
| ZDT4 | 9.997E-01 | 2.943E-04 | 7.071E-01 | 0.000E+00 | 9.993E-01 | 8.246E-04 | 9.995E-01 | 4.602E-04 | 9.993E-01 | 6.389E-04 | + |
| ZDT6 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 9.994E-01 | 9.038E-05 | 9.996E-01 | 1.487E-04 | 9.994E-01 | 1.941E-04 | + |
| Constr | 9.110E-01 | 8.802E-02 | 9.976E-01 | 2.618E-03 | 9.995E-01 | 5.934E-04 | 9.986E-01 | 1.870E-03 | 9.973E-01 | 3.709E-03 | + |
| Srinivas | 9.810E-01 | 1.260E-02 | 9.910E-01 | 5.351E-03 | 9.938E-01 | 3.996E-03 | 9.935E-01 | 3.471E-03 | 9.924E-01 | 4.234E-03 | + |
| Osycka2 | 8.128E-01 | 1.968E-01 | 8.759E-01 | 2.947E-01 | 8.207E-01 | 3.777E-01 | 9.950E-01 | 1.854E-01 | 8.127E-01 | 2.321E-01 | + |
| Golinski | 9.684E-01 | 4.090E-02 | 9.952E-01 | 8.048E-03 | 1.000E+00 | 2.121E-03 | 9.974E-01 | 1.353E-02 | 9.855E-01 | 2.990E-02 | + |
| Tanaka | 9.665E-01 | 5.070E-02 | 9.943E-01 | 3.236E-03 | 9.982E-01 | 3.187E-04 | 9.981E-01 | 5.481E-04 | 9.976E-01 | 1.702E-03 | + |
| Binh2 | 7.695E-01 | 1.419E-04 | 8.111E-01 | 0.000E+00 | 8.111E-01 | 4.235E-10 | 8.111E-01 | 1.988E-10 | 8.111E-01 | 5.356E-10 | + |
| DTLZ1 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | - |
| DTLZ2 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 2.220E-16 | 1.000E+00 | 4.421E-06 | - |
| DTLZ3 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | - |
| DTLZ4 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 0.000E+00 | - |
| DTLZ5 | 6.104E-01 | 1.311E-02 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 1.439E-08 | 1.000E+00 | 1.998E-15 | 1.000E+00 | 1.099E-05 | + |
| DTLZ6 | 8.171E-01 | 0.000E+00 | 1.000E+00 | 1.110E-16 | 1.000E+00 | 0.000E+00 | 1.000E+00 | 1.774E-08 | 1.000E+00 | 4.589E-09 | + |
| DTLZ7 | 9.984E-01 | 7.952E-03 | 9.957E-01 | 3.671E-03 | 7.075E-01 | 4.078E-01 | 9.998E-01 | 7.590E-05 | 9.947E-01 | 1.598E-03 | + |
| Viennet1 | 9.997E-01 | 8.670E-04 | 9.999E-01 | 7.233E-04 | 1.000E+00 | 2.185E-07 | 1.000E+00 | 5.530E-05 | 9.987E-01 | 3.927E-03 | + |
| Viennet2 | 9.809E-01 | 2.253E-02 | 9.972E-01 | 5.491E-03 | 9.975E-01 | 2.324E-04 | 9.973E-01 | 9.509E-04 | 9.952E-01 | 7.998E-03 | + |
| Viennet3 | 9.980E-01 | 4.372E-03 | 9.999E-01 | 1.760E-03 | 1.000E+00 | 1.248E-05 | 1.000E+00 | 5.939E-05 | 9.996E-01 | 1.643E-03 | + |
| Viennet4 | 9.868E-01 | 1.630E-02 | 9.947E-01 | 4.593E-03 | 1.000E+00 | 7.993E-05 | 9.988E-01 | 4.537E-03 | 9.877E-01 | 1.217E-02 | + |
| Tamaki | 8.937E-01 | 4.965E-02 | 1.020E-02 | 0.000E+00 | 1.020E-02 | 1.388E-17 | 1.020E-02 | 1.388E-17 | 1.020E-02 | 1.976E-07 | + |

**Table 9.** Median and interquartile range of the hypervolume metric (50,000 function evaluations).

| Problem | IBMO | | OMOPSO | | AbYSS | | NSGA2 | | SPEA2 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | $\tilde{x}$ | $IQR$ | $\tilde{x}$ | $IQR$ | $\tilde{x}$ | $IQR$ | $\tilde{x}$ | $IQR$ | $\tilde{x}$ | $IQR$ | |
| Schaffer | 8.299E-01 | 4.796E-05 | 8.299E-01 | 4.443E-05 | 8.299E-01 | 4.373E-05 | 7.097E-01 | 8.992E-02 | 7.051E-01 | 7.526E-02 | + |
| Fonseca | 3.121E-01 | 6.856E-05 | 3.125E-01 | 2.367E-05 | 3.110E-01 | 2.776E-04 | 3.080E-01 | 5.364E-04 | 3.107E-01 | 2.337E-04 | + |
| Kursawe | 4.016E-01 | 6.182E-05 | 3.985E-01 | 4.520E-04 | 4.014E-01 | 1.378E-04 | 3.996E-01 | 3.046E-04 | 4.011E-01 | 1.879E-04 | + |
| ZDT1 | 6.620E-01 | 5.394E-05 | 6.617E-01 | 1.679E-04 | 6.620E-01 | 7.801E-05 | 6.602E-01 | 4.030E-04 | 6.616E-01 | 1.508E-04 | + |
| ZDT2 | 3.287E-01 | 4.366E-05 | 3.285E-01 | 1.297E-04 | 3.287E-01 | 8.112E-05 | 3.272E-01 | 2.817E-04 | 3.284E-01 | 1.079E-04 | + |
| ZDT3 | 5.155E-01 | 5.493E-04 | 5.154E-01 | 5.209E-04 | 5.160E-01 | 3.492E-03 | 5.154E-01 | 1.253E-04 | 5.157E-01 | 9.789E-05 | + |
| ZDT4 | 6.637E-01 | 1.138E-03 | 0.000E+00 | 0.000E+00 | 6.596E-01 | 2.659E-03 | 6.593E-01 | 1.397E-03 | 6.605E-01 | 1.398E-03 | + |
| ZDT6 | 4.014E-01 | 1.850E-05 | 4.013E-01 | 6.992E-05 | 4.007E-01 | 1.350E-04 | 3.981E-01 | 6.831E-04 | 4.004E-01 | 2.869E-04 | + |
| Constr | 7.744E-01 | 5.585E-03 | 7.767E-01 | 1.990E-04 | 7.765E-01 | 1.774E-04 | 7.742E-01 | 3.650E-04 | 7.758E-01 | 3.537E-04 | + |
| Srinivas | 5.409E-01 | 8.779E-05 | 5.408E-01 | 8.297E-05 | 5.408E-01 | 1.045E-04 | 5.380E-01 | 4.697E-04 | 5.400E-01 | 2.159E-04 | + |
| Osycka2 | 7.095E-01 | 5.144E-02 | 7.088E-01 | 1.386E-02 | 7.453E-01 | 3.462E-01 | 7.525E-01 | 7.725E-03 | 7.445E-01 | 2.284E-02 | + |
| Golinski | 9.696E-01 | 6.661E-05 | 9.683E-01 | 1.879E-04 | 9.695E-01 | 9.083E-05 | 9.691E-01 | 1.937E-04 | 9.676E-01 | 4.594E-04 | - |
| Tanaka | 3.036E-01 | 2.831E-03 | 3.073E-01 | 4.831E-04 | 3.083E-01 | 3.527E-04 | 3.078E-01 | 3.185E-04 | 3.094E-01 | 2.916E-04 | + |
| Binh2 | 7.990E-01 | 3.267E-05 | 8.008E-01 | 2.445E-05 | 8.006E-01 | 4.604E-05 | 7.994E-01 | 2.505E-04 | 7.999E-01 | 1.808E-04 | + |
| DTLZ1 | 7.402E-01 | 3.828E-02 | 0.000E+00 | 0.000E+00 | 7.952E-01 | 7.750E-03 | 7.577E-01 | 6.764E-03 | 6.431E-01 | 2.094E-01 | + |
| DTLZ2 | 4.372E-01 | 1.984E-03 | 4.222E-01 | 2.164E-03 | 4.351E-01 | 2.568E-03 | 3.738E-01 | 5.607E-03 | 4.454E-01 | 6.933E-04 | + |
| DTLZ3 | 0.000E+00 | 2.167E-02 | 0.000E+00 | 0.000E+00 | 2.966E-01 | 3.829E-01 | 3.206E-01 | 5.172E-02 | 0.000E+00 | 0.000E+00 | + |
| DTLZ4 | 4.389E-01 | 2.794E-03 | 4.041E-01 | 2.824E-02 | 4.364E-01 | 1.660E-03 | 3.762E-01 | 7.384E-03 | 4.422E-01 | 7.397E-04 | + |
| DTLZ5 | 2.119E-02 | 5.949E-03 | 9.512E-02 | 4.373E-05 | 9.538E-02 | 7.127E-06 | 9.232E-02 | 2.544E-04 | 9.498E-02 | 6.099E-05 | + |
| DTLZ6 | 2.657E-02 | 0.000E+00 | 9.536E-02 | 7.142E-06 | 5.385E-02 | 1.942E-02 | 8.111E-02 | 1.755E-02 | 0.000E+00 | 0.000E+00 | + |
| DTLZ7 | 3.224E-01 | 3.648E-03 | 3.035E-01 | 5.500E-03 | 2.724E-01 | 5.593E-02 | 2.841E-01 | 5.172E-03 | 3.082E-01 | 1.887E-03 | + |
| Viennet1 | 7.371E-01 | 4.748E-04 | 7.376E-01 | 4.059E-04 | 7.363E-01 | 6.108E-04 | 7.106E-01 | 2.796E-03 | 7.402E-01 | 1.920E-04 | + |
| Viennet2 | 9.291E-01 | 3.132E-04 | 9.294E-01 | 2.135E-04 | 9.286E-01 | 4.183E-04 | 9.198E-01 | 1.713E-03 | 9.300E-01 | 7.607E-05 | + |
| Viennet3 | 8.403E-01 | 1.211E-04 | 8.401E-01 | 7.993E-05 | 8.399E-01 | 1.229E-04 | 8.327E-01 | 9.393E-04 | 8.374E-01 | 2.602E-04 | + |
| Viennet4 | 8.685E-01 | 6.997E-04 | 8.701E-01 | 4.280E-04 | 8.682E-01 | 9.228E-04 | 8.554E-01 | 2.908E-03 | 8.727E-01 | 2.841E-04 | + |
| Tamaki | 3.192E-01 | 1.508E-02 | 4.197E-01 | 1.782E-03 | 4.184E-01 | 2.353E-03 | 3.567E-01 | 8.989E-03 | 4.346E-01 | 7.595E-04 | + |

## Appendix B

**Table 10.** Definition of Viennet family and Tamaki test problems

| Problem | D | Variable Bounds | Objective Functions |
|---|---|---|---|
| Schaffer | 1 | $[-10^3, 10^3]$ | $f_1(x) = x^2$<br>$f_2(x) = (x-2)^2$ |
| Fonseca | 3 | [ -4 , 4 ] | $f_1(x) = 1 - exp(-\sum_{i=1}^{3}(x_i - \frac{1}{\sqrt{3}})^2)$ $\quad f_2(x) = 1 - exp(-\sum_{i=1}^{3}(x_i + \frac{1}{\sqrt{3}})^2)$ |
| Kursawe | 3 | [ -5 , 5 ] | $f_1(x) = \sum_{i=1}^{n-1}(-10\,exp(-0.2\sqrt{x_i^2 + x_{i+1}^2}))$ $\quad f_2(x) = \sum_{i=1}^{n}(\lvert x_i\rvert^{0.8} + 5\sin x_i^3)$ |
| ZDT1 | 30 | [ 0 , 1 ] | $f_1(x) = x_1$ $\quad f_2(x) = g(x)(1 - \sqrt{x_1/g(x)})$<br>$g(x) = 1 + 9\sum_{i=2}^{D} x_i/(D-1)$ |
| ZDT2 | 30 | [ 0 , 1 ] | $f_1(x) = x_1$ $\quad f_2(x) = g(x)(1 - (x_1/g(x))^2)$<br>$g(x) = 1 + 9\sum_{i=2}^{D} x_i/(D-1)$ |
| ZDT3 | 30 | [ 0 , 1 ] | $f_1(x) = x_1$ $\quad f_2(x) = g(x)(1 - \sqrt{\frac{x_1}{g(x)}} -$<br>$(x_1/g(x))\,sin(10\pi x_i))$<br>$g(x) = 1 + 9\sum_{i=2}^{D} x_i/(D-1)$ |
| ZTD4 | 10 | $x_1 \in [0,1]$<br>$x_i \in [-5,5]$<br>$i = 2,\ldots,D$ | $f_1(x) = x_1$ $\quad f_2(x) = g(x)(1 - \sqrt{x_1/g(x)})$<br>$g(x) = 1 + 10\,(D-1) + \sum_{i=2}^{D}[x_i^2 - 10\,cos(4\pi x_i)]$ |
| ZDT6 | 10 | [ 0 , 1 ] | $f_1(x) = 1 - exp(-4x_1)\,sin^6(6\pi x_1)$ $\quad f_2(x) = g(x)(1 - (x_1/g(x))^2)$<br>$g(x) = 1 + 9[\sum_{i=2}^{D} x_i/(D-1)]^{0.25}$ |

**Table 11.** Definition of Viennet family and Tamaki test problems

| Problem | D | Variable Bounds | Objective Functions |
|---|---|---|---|
| Constr | 2 | $x_1 \in [0.1, 1]$<br>$x_2 \in [0, 5]$ | $f_1(x) = x_1$<br>$f_2(x) = (1 + x_2)/x_1$<br>$g_1(x) = x_2 + 9x_1 \geq 6$<br>$g_2(x) = -x_2 + 9x_1 \geq 1$ |
| Srinivas | 2 | $x_1, x_2$<br>$\in [-20, 20]$ | $f_1(x) = 2 + (x_1 - 2)^2 + (x_2 - 1)^2$<br>$f_2(x) = 9x_1 - (x_2 - 1)^2$<br>$g_1(x) = x_1^2 + x_2^2 \leq 225$<br>$g_2(x) = x_1 - 3x_2 + 10 \leq 0$ |
| Osyczka2 | 6 | $x_1, x_2 \in [0, 10]$<br>$x_3, x_5 \in [1, 5]$<br>$x_4 \in [0, 6]$<br>$x_6 \in [1, 10]$ | $f_1(x) = -[25(x_1 - 2)^2 + (x_2 - 2)^2 + (x_3 - 1)^2 + (x_4 - 4)^2 + (x_5 - 1)^2]$<br>$f_2(x) = x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2 + x_6^2$<br>$g_1(x) = x_1 + x_2 - 2 \geq 0$<br>$g_2(x) = 6 - x_1 - x_2 \geq 0$<br>$g_3(x) = 2 - x_2 + x_1 \geq 0$<br>$g_4(x) = 2 - x_1 + 3x_2 \geq 0$<br>$g_5(x) = 4 - (x_3 - 3)^2 - x_4 \geq 0$<br>$g_6(x) = (x_5 - 3)^2 + x_6 - 4 \geq 0$ |
| Golinski | 7 | $x_1 \in [2.6, 3.6]$<br>$x_2 \in [0.7, 0.8]$<br>$x_3 \in [17, 28]$<br>$x_4 \in [7.3, 8.3]$<br>$x_5 \in [7.3, 8.3]$<br>$x_6 \in [2.9, 3.9]$<br>$x_7 \in [5.0, 5.5]$ | $f_1(x) = 0.7854\, x_1 x_2{}^2 (10 x_3{}^2/3 + 14.933\, x_3 - 43.0934) -$<br>$1.508\, x_1 (x_6{}^2 + x_7{}^2) + 7.477\,(x_6{}^3 + x_7{}^3) + 0.7854\,(x_4 x_6{}^2 + x_5 x_7{}^2)$<br>$f_2(x) = \sqrt{(745\, x_4/x_2 x_3)^2 + 1.69 \times 10^7}/0.1 x_6{}^2$<br>$g_1(x) = \frac{1.0}{(x_1 x_2{}^2 x_3)} - \frac{1}{27} \leq 0$<br>$g_2(x) = \frac{1.0}{(x_1 x_2{}^2 x_3{}^2)} - \frac{1}{397.5} \leq 0$<br>$g_3(x) = \frac{x_4{}^3}{(x_2 x_3{}^2 x_6{}^4)} - \frac{1}{1.93} \leq 0$<br>$g_4(x) = \frac{x_5{}^3}{(x_2 x_3 x_7{}^4)} - \frac{1}{1.93} \leq 0$<br>$g_5(x) = x_2 x_3 - 40 \leq 0$<br>$g_6(x) = x_1/x_2 - 12 \leq 0$<br>$g_7(x) = 5 - x_1/x_2 \leq 0$<br>$g_8(x) = 1.9 - x_4 + 1.5x_6 \leq 0$<br>$g_9(x) = 1.9 - x_5 + 1.1x_7 \leq 0$<br>$g_{10}(x) = f_2(x) \leq 1300$<br>$a = 745\, x_5/x_2 x_3$<br>$b = 1.575 \times 10^8$<br>$g_{11}(x) = \frac{\sqrt{a^2 + b}}{0.1 x_7^3} \leq 1100$ |
| Tanaka | 2 | $x_1, x_2 \in [0, \pi]$ | $f_1(x) = x_1$<br>$f_2(x) = x_2$<br>$g_1(x) = x_1^2 + x_2^2 - 1 - 0.1 \cos(16 \arctan(\frac{x_1}{x_2})) \geq 0$<br>$g_2(x) = (x_1 - 0.5)^2 + (x_2 - 0.5)^2 \leq 0.5$ |
| Binh2 | 2 | $x_1$ ⊠ $[0, 5]$<br>$x_2 \in [0, 3]$ | $f_1(x) = 4x_1^2 + 4x_2^2$ $\quad f_2(x) = (x_1 - 5)^2 + (x_2 - 5)^2$<br>$g_1(x) = (x_1 - 0.5)^2 + x_2{}^2 - 25 \leq 0$<br>$g_2(x) = (x_1 - 8)^2 + (x_2 - 3)^2 - 7.7 \geq 0$ |

**Table 12.** Definition of DTLZ test problems.

| Problem | D | Variable Bounds | Objective Functions |
|---|---|---|---|
| DTLZ1 | 7 | $x_i \in [0, 1]$<br>$i = 1, \ldots, D$ | $f_1(x) = 0.5x_1x_2(1 + g(x))$<br>$f_2(x) = 0.5x_1(1 - x_2)(1 + g(x))$<br>$f_3(x) = 0.5(1 - x_1)(1 + g(x))$<br>$g(x) = 100\ [5 + \sum_{i=3}^{D}((x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5)))]$ |
| DTLZ2 | 12 | $x_i \in [0, 1]$<br>$i = 1, \ldots, D$ | $f_1(x) = (1 + g(x))\cos(0.5\pi x_1)\cos(0.5\pi x_2)$<br>$f_2(x) = (1 + g(x))\cos(0.5\pi x_1)\sin(0.5\pi x_2)$<br>$f_3(x) = (1 + g(x))\sin(0.5\pi x_1)$<br>$g(x) = \sum_{i=3}^{D}(x_i - 0.5)^2$ |
| DTLZ3 | 12 | $x_i \in [0, 1]$<br>$i = 1, \ldots, D$ | $f_1(x) = (1 + g(x))\cos(0.5\pi x_1)\cos(0.5\pi x_2)$<br>$f_2(x) = (1 + g(x))\cos(0.5\pi x_1)\sin(0.5\pi x_2)$<br>$f_3(x) = (1 + g(x))\sin(0.5\pi x_1)$<br>$g(x) = 100\ [10 + \sum_{i=3}^{D}((x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5)))]$ |
| DTLZ4 | 12 | $x_i \in [0, 1]$<br>$i = 1, \ldots, D$ | $f_1(x) = (1 + g(x))\cos(0.5\pi {x_1}^{100})\cos(0.5\pi {x_2}^{100})$<br>$f_2(x) = (1 + g(x))\cos(0.5\pi {x_1}^{100})\sin(0.5\pi {x_2}^{100})$<br>$f_3(x) = (1 + g(x))\sin(0.5\pi {x_1}^{100})$<br>$g(x) = \sum_{i=3}^{D}(x_i - 0.5)^2$ |
| DTLZ5 | 12 | $x_i \in [0, 1]$<br>$i = 1, \ldots, D$ | $f_1(x) = (1 + g(x))\cos(0.5\pi\theta_1)\cos(0.5\pi\theta_2)$<br>$f_2(x) = (1 + g(x))\cos(0.5\pi\theta_1)\sin(0.5\pi\theta_2)$<br>$f_3(x) = (1 + g(x))\sin(0.5\pi\theta_1)$<br>$g(x) = \sum_{i=3}^{D}(x_i - 0.5)^2$<br>$\theta_i = \pi(1 + 2x_i g(x))/(4(1 + g(x)))$ |
| DTLZ6 | 12 | $x_i \in [0, 1]$<br>$i = 1, \ldots, D$ | $f_1(x) = (1 + g(x))\cos(0.5\pi\theta_1)\cos(0.5\pi\theta_2)$<br>$f_2(x) = (1 + g(x))\cos(0.5\pi\theta_1)\sin(0.5\pi\theta_2)$<br>$f_3(x) = (1 + g(x))\sin(0.5\pi\theta_1)$<br>$g(x) = \sum_{i=3}^{D}{x_i}^{0.1}$<br>$\theta_i = \pi(1 + 2x_i g(x))/(4(1 + g(x)))$ |
| DTLZ7 | 22 | $x_i \in [0, 1]$<br>$i = 1, \ldots, D$ | $f_1(x) = x_1$<br>$f_2(x) = x_2$<br>$f_3(x) = (1 + g(x))\mathrm{h}(x)$<br>$g(x) = 1 + 9\sum_{i=3}^{D}x_i/20$<br>$\mathrm{h}(x) = 3 - \sum_{i=1}^{2}f_i.(1 + \sin(3\pi f_i))/(1 + g(x))$ |

**Table 13.** Definition of Viennet family and Tamaki test problems.

| Problem | D | Variable Bounds | Objective Functions |
|---|---|---|---|
| Viennet | 2 | $x_1, x_2 \in [-2, 2]$ | $f_1(x) = x_1{}^2 + (x_2 - 1)^2$<br>$f_2(x) = x_1{}^2 + (x_2 + 1)^2 + 1$<br>$f_3(x) = (x_1 - 1)^2 + x_2{}^2 + 2$ |
| Viennet2 | 2 | $x_1, x_2 \in [-4, 4]$ | $f_1(x) = (x_1 - 2)^2/2 + (x_2 + 1)^2/13 + 3$<br>$f_2(x) = (x_1 + x_2 - 3)^2/36 + (2x_2 - x_1)^2/8 - 17$<br>$f_3(x) = (x_1 + 2x_2 - 1)^2/175 + (2x_2{}^2 - x_1)^2/17 - 13$ |
| Viennet3 | 2 | $x_1, x_2 \in [-3, 3]$ | $f_1(x) = (x_1{}^2 + x_2{}^2)/2 + \sin(x_1{}^2 + x_2{}^2)$<br>$f_2(x) = (3x_1 - 2x_2 + 4)^2/8 + (x_1 - x_2 + 1)^2/27 - 15$<br>$f_3(x) = 1/(x_1{}^2 + x_2{}^2 + 1) - 1.1exp(-x_1{}^2 - x_2{}^2)$ |
| Viennet4 | 2 | $x_1, x_2 \in [-4, 4]$ | $f_1(x) = (x_1 - 2)^2/2 + (x_2 + 1)^2/13 + 3$<br>$f_2(x) = (x_1 + x_2 - 3)^2/175 + (2x_2 - x_1)^2/17 - 13$<br>$f_3(x) = (3x_1 + 2x_2 + 4)^2/8 + (x_1 - x_2 + 1)^2/27 + 15$<br>$g_1(x) = 4x_1 + x_2 - 4 < 0$<br>$g_2(x) = x_1 + 1 > 0$<br>$g_3(x) = x_1 - x_2 - 2 < 0$ |
| Tamaki | 3 | $x_1, x_2, x_3 \in [0, 1]$ | $f_1(x) = x_1$<br>$f_2(x) = x_2$<br>$f_3(x) = x_3$<br>$g(x) = x_1{}^2 + x_2{}^2 + x_3{}^2 - 1 \leq 0$ |