Alexandria Engineering Journal (2018) 57, 1619–1629

HOSTED BY ELSEVIER

Alexandria University

Alexandria Engineering Journal

www.elsevier.com/locate/aej
www.sciencedirect.com

ORIGINAL ARTICLE

# Multi-swarm multi-objective optimization based on a hybrid strategy

**Shery Sedarous** *, **Sherin M. El-Gokhy**, **Elsayed Sallam**

*Computer and Control Engineering Department, Faculty of Engineering, Tanta University, Egypt*

Received 26 January 2017; revised 10 May 2017; accepted 19 June 2017
Available online 31 July 2017

**KEYWORDS**

Multi-objective;
Multi-swarm;
Decomposition;
Dominance

**Abstract** Multi-objective optimization is a very competitive issue that emerges naturally in most real world problems. It is concerned with the optimization of conflicting objectives in multi-objective problems. The multi-objective problem treats with tradeoff solutions in order to satisfy all objectives. An extensive variety of algorithms has been developed to solve multi-objective optimization problems. In this paper, we presents a multi-swarm multi-objective intelligence-based algorithm enhanced with a hybrid strategy between decomposition and dominance (**MSMO/2D**) to improve convergence and diversity by splitting the primary swarm into a number of sub-swarms. The proposed algorithm is applied to fourteen standard problems and compared with two of the most familiar multi-objective optimization algorithms MOEA/D and D$^2$MOPSO. The experimental results give evidence that the multi-swarm armed by the hybrid strategy constitutes a better alternative for multi-objective optimization problems.



## 1. Introduction

Multi-objective optimization is a real challenge that attracts the attention of scientific researchers in different disciplines especially in engineering and science. Many population-based metaheuristics algorithms have been proposed to solve the multi-objective problems (**MOP**). These algorithms are included in two main sets; evolutionary-based optimization algorithms besides swarm intelligence-based optimization algorithms.

Multi-objective evolutionary-based optimization algorithms adopt the evolutionary computation strategy that simulates the biological evolutions, such as cloning, mutation, mating, and selection. Aiming to solve multi-objective optimization problems, a considerable variety of evolutionary optimization algorithms were elaborated [1–6].

While multi-objective swarm intelligence-based optimization algorithms follow the computational intelligence paradigm in imitating the swarm behavior by adjusting position and velocity for each particle as well as its neighbors. Several intelligence-based algorithms have been investigated in multi-objective optimization [7–10].

* Corresponding author.
E-mail addresses: shery@f-eng.tanta.edu.eg (S. Sedarous), sherin.elgokhy@f-eng.tanta.edu.eg (S.M. El-Gokhy), sallam@f-eng.tanta.edu.eg (E. Sallam).

Peer review under responsibility of Faculty of Engineering, Alexandria University.

http://dx.doi.org/10.1016/j.aej.2017.06.017
1110-0168 

Particle swarm optimization (**PSO**) is considered as the most familiar intelligence-based algorithm utilized to solve MOP [11–13].

All the mentioned algorithms have been applied to test problems as well as real world problems in order to be evaluated. In spite of the fact that they succeed to achieve the Pareto front (**PF**) approximation for the multi-objective optimization problems, they suffer from highly computational complexity. Therefore, the decomposition concept is considered in order to improve the complexity by decomposing a multi-objective optimization problem into multiple scalar optimization sub-problems. Then, those sub-problems are optimized simultaneously [14–17].

Another reinforcement has been added to multi-objective algorithms; namely, dominance. The dominance-based approaches consider the concept of dominance to scalarize the objective vector in the fitness function in order to treat all objectives correlatively to find the Pareto optimal solution. Dominance guarantees the impossibility to find a solution that improves an objective without degrading at least another one [18–22].

Recently, Al Moubayed et al. have proposed a new technique that hybridizes decomposition and dominance approaches [23].

In this paper, we propose MSMO/2D: a multi-swarm optimization algorithm enhanced by decomposition and dominance. It performs the optimization using multiple sub-swarms rather than one normal swarm. Our proposed algorithm gives a significant performance improvement comparing to MOEA/D and D$^2$MOPSO. It achieves at least two advantages: the potential integration with other search techniques and the augmentation of the population diversity.

This paper is arranged into six sections. In addition to the present section (introduction), Section 2 provides essential concepts that are required to understand MOP along with a literature review of the related work. Section 3 illustrates approaches that are adopted in our proposed algorithm. Then, Section 4 explains our proposed MSMO/2D algorithm. After that, Section 5 presents the performance evaluation of the proposed algorithm through experiments. Finally, conclusion and trends for pertinent future work are given in Section 6.

## 2. Background and related work

This section aims to give a birds-eye view on the methods and concepts required to understand the proposed algorithm. It also provides a detailed review of the related work.

### 2.1. Multi-objective optimization

The multi-objective optimization is interested in solving problems that have various objective functions to be optimized concurrently to the same set of solutions. Such problems are interested not only in a mono optimal solution but in the tradeoffs between the different objectives. These problems can be expressed as [12]:

$$\text{Min. or Max.}|x \in \Omega\{F(x) \in \Delta\} \tag{1}$$

where $F(x) = (f_1(x), f_2(x), \ldots, f_m(x))$, $\Omega$ and $\Delta$ are the solution and objective spaces respectively and F is an objective function vector.

Multi-objective optimization cares in the set of non-dominated Pareto optimal solutions.

As $x$ dominates $y$ if and only if $f_i(x) \leq f_i(y)$ for all $i = \{1, 2, \ldots, m\}$ and there is at least one $j$ for which $f_j(x) < f_j(y)$. Therefore, $x$ is considered as a Pareto optimal solution if there is no other solution $s \in \Omega$ such that $s$ dominates $x^*$.

### 2.2. Multi-swarm optimization

Multi-swarm optimization is derived from PSO. It uses multiple sub-swarms rather than one standard swarm. This multi-swarm framework is the most appropriate framework for optimizing MOP [24].

PSO is a metaheuristic algorithm based on population that yields competitive solutions in many application domains. Each particle in the swarm offers a possible solution in the solution space and is specified by its position and velocity vector. Where, the position specifies the particle's location in the solution space and the velocity represents the positional change. PSO utilizes the following formula to update the position $x$ and velocity $v$ of the particles:

$$v_{i(k+1)} = wv_{i(k)} + c_1r_1(x_{pbesti} - x_{i(k)}) + c_2r_2(x_{lbesti} - x_{i(k)}) \tag{2}$$

$$x_{i(k+1)} = x_{i(k)} + v_{i(k+1)} \tag{3}$$

where $p_{besti}$ and $l_{besti}$ are the personal best and local best performance respectively for the $i$th particle, $c_1$ and $c_2$ are nonnegative constants; namely, acceleration coefficients, $r_1$, and $r_2$ are two random numbers within [0, 1], and $w$ is an inertia weight.

### 2.3. Overview of the related work

Multi-Objective Particle Swarm Optimization (**MOPSO**) was firstly presented by Moore and Chapman as unpublished manuscript in 1999 [13]. Since that date, several MOPSO methods have been developed and applied to various real-life problems and standard benchmarks [12,25–29]. Whilst, the real outset of MOPSO was proposed in 2004 by Coello et al. [21].

Zhang and Li developed an algorithm for solving MOP by employing the evolutionary algorithm depending on the decomposition approach for the first time and called it (MOEA/D) in 2007 [14]. Their algorithm relies on decomposing MOP to a number of sub-problems and optimizing them simultaneously as scalar values; optimizing each sub-problem only through notifications from its neighboring sub-problems. Then in 2008, W. Peng and Q. Zhang set PSO in lieu of genetic algorithm in MOEA/D titled (**MOPSO/D**) [15]. Unfortunately, it was exposed to fall in local optima because of applying mutation.

Al Moubayed et al. in 2010 proposed smart particle swarm optimization algorithm based on decomposition (**SDMOPSO**) [16]. They tried to avoid falling in local optima by permitting the update of the particle position if it leads to a better aggregation value.

In May 2011, Yong Zhang et al. presented multi-swarm cooperative multi-objective particle swarm optimizer

(**MC-MOPSO**) adopted multi-swarm by dividing swarm to several slave swarms and a single master swarm, where each slave swarm optimizes an objective and the master swarm uses the MOPSO to cover gaps between the nondominated solutions to conserve Pareto front distribution. However, their optimizer suffered from the limitation of setting the radius of the species that sensitively affective in the algorithm efficiency [24].

Martínez and Coello in July 2011, decided to use the decomposition approach in updating the leaders' archive, selecting the swarm leaders and reinitializing the particles that became non-updatable after a specific number of iterations and exceeds the age. They called it decomposition-based multi-objective particle swarm optimizer (**dMOPSO**) [17]. However, reinitializing particles lead to losing the experience gained during the exploration process. Moreover, using decomposition in lieu of the dominance might not success in covering complicated Pareto front.

Later in 2011, Peng Hu et al. introduced a multi-swarm approach relying on both MOPSO and decomposition (**MOPSO_MS**) by constructing a new update strategy for velocity [30]. In 2014, Al Moubayed et al. developed a new MOPSO approach based on decomposition enriched with dominance (**D$^2$MOPSO**) by submitting an archiving technique to get better diversity of the obtained solutions and more coverage in both solution space and also in objective space [23,31].

In 2015, Lin et al. suggested to apply several search strategies rather than one strategy in updating the velocity of each particle in the MOPSO technique naming it (**MMOPSO**) [32]. Recently in 2016, A Dıaz-Manrıquez et al. coupled R2 indicator with PSO (**R2-MOPSO**) instead of the Pareto dominance and the external archive [33]. Later in 2016, Li Li et al. adopt a new mechanism that selects the optimal guides in MOPSO considering the ranking dominance and integrates into MOPSO [34].

## 3. Methodology

The strategies adopted in our algorithm are explained in details below.

### 3.1. Decomposition strategy

In decomposition strategy, each MOP is divided into a number of mono-objective optimization sub-problems. There are several approaches for constructing aggregation functions of decomposition (e.g. weighted sum, Tchebycheff, weighted Tchebycheff and Penalty based boundary intersection (**PBI**)). PBI uses a coefficient weighted vector $V$ and a penalty value $\theta$ to decrease both the distance to the ideal vector $d_1$ and the direction error to the weighted vector $d_2$ (see Fig. 1).

In case of minimization of Eq. (1):

$$\text{Minimize}: \ g(x|V, z^*) = d_1 + \theta d_2 \tag{4}$$

where

$$d_1 = \|(F(x) - z^*)^T * V\| / \|V\|$$

$$d_2 = \|(F(x) - z^*) - d_1 * V/\|V\|\|$$

and $z^*$ is a reference point.

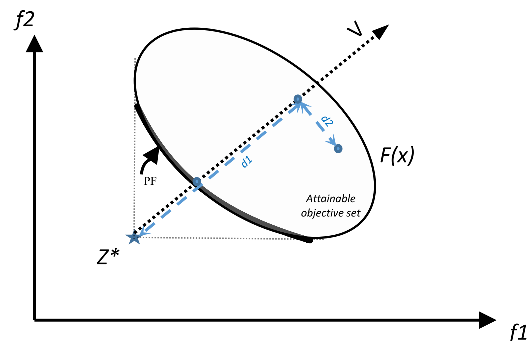


**Figure 1** Penalty boundary intersection.

This paper adopts the PBI approach as it generates optimal solutions more uniformly distributed. PBI avoids getting an aggregated function such that $g\ (x|V, z^*) = g\ (y|V, z^*)$ [14].

### 3.2. Dominance strategy

The dominance-based approaches; namely, Pareto approaches depend on both dominance and Pareto optimality in the fitness assignment evaluation, contrary to the other approaches that use a scalarization function and treat the various objectives separately. The advantages of dominance strategies are that they do not transform MOP to one objective problem and they are capable of generating a diverse set of Pareto optimal solutions. Most of Pareto approaches use the evolutionary-based algorithms in optimizing MOP (e.g. NSGA-II and SPEA2) [35].

The majority of dominance based MOPSO algorithms use a restricted archive to back up the trade-off solutions that are received over the optimization procedure [36].

There are several density estimator methods for sustaining the archive and picking up leaders (e.g. kernel or crowding distance [37], nearest-neighbor [38], adaptive grid [39], niche count [40], and $\epsilon$-dominance [41]). In this paper, we utilize the crowding distance estimator as it is a nonparametric approach that constructs a reasonable density fit for both solution and objective spaces.

### 3.3. Hybrid strategy

This strategy is proposed recently in D$^2$MOPSO [23]. It makes an integration between dominance and decomposition. D$^2$MOPSO uses dominance-based ranking for the non-dominated solutions of the leaders' archive using the crowding distance values in both solution and objective spaces to store only non-dominated particles. While, updating the personal best values, and the leaders are selected using the decomposition's aggregation function.

$$KD_{(P_i)} = \left( \sum_{j=1}^{Cs} \|P_i, P_j\|_{\Omega}, \sum_{j=1}^{Cs} \|F(P_i), F(P_j)\|_{\Delta} \right) \tag{5}$$

where $KD_{(pi)}$ is a vector of the crowding distances of particle $P_i$ in both solution and objective spaces, $Cs$ is the archive size, $P_i$ and $P_j$ are the decision variable vectors of particles $i$, $j$ (see Fig. 2).

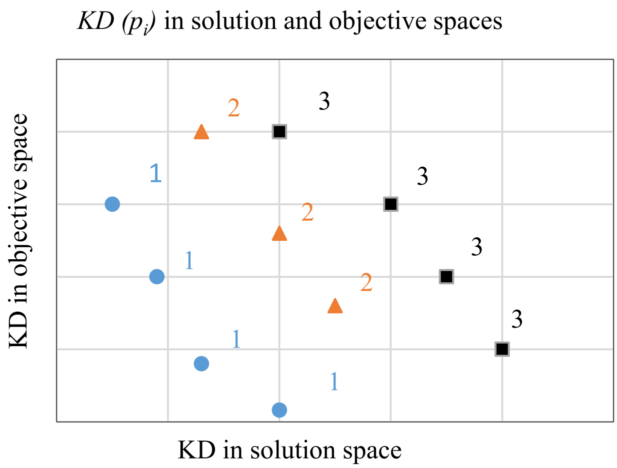


**Figure 2** Non-dominated solution ranking in the KD ($p_i$).

## 4. The proposed algorithm

We propose MSMO/2D in order to solve MOP. It is a multi-swarm based on PSO algorithm and enhanced with a hybrid strategy between decomposition and dominance. The MSMO/2D algorithm relies on the hybrid strategy that integrates both decomposition and dominance strategies as illustrated in the hybrid strategy.

MSMO/2D aims to minimize the gap between the solutions in the leaders' archive (the approximated PF) and the true Pareto front, as well as, maximize the diversity between solutions. The MSMO/2D algorithm works as follows:

*Input:* MOP (1);
  Swarm_size: number of the swarm particles;
  No_subswarms: number of subswarms;
*Step1:* Calculate Subswarm size = Swarm_size/No_subswarms;
*Step 2: For* subswarm = 1 to No_subswarms *do*
  *For* $t$ = 1 to Max_iterations do
  Apply PSO algorithm as in Eqs. (2) and (3);
  *If* Size (leaders_archive) > = Max_size then
    Find $KD_{(Pi)}$ as in Eq. (5);
    Replace the particle with the worst rank;
  *End If*
  Update leaders archive;
  Update external archive;
  *End For*
  Return final result in the external archive;
  Append the result to the results file;
*End For*

## 5. Experiments and results

This section defines test problems, as well as, the experiment setup. Then, the indicators that are used to validate the

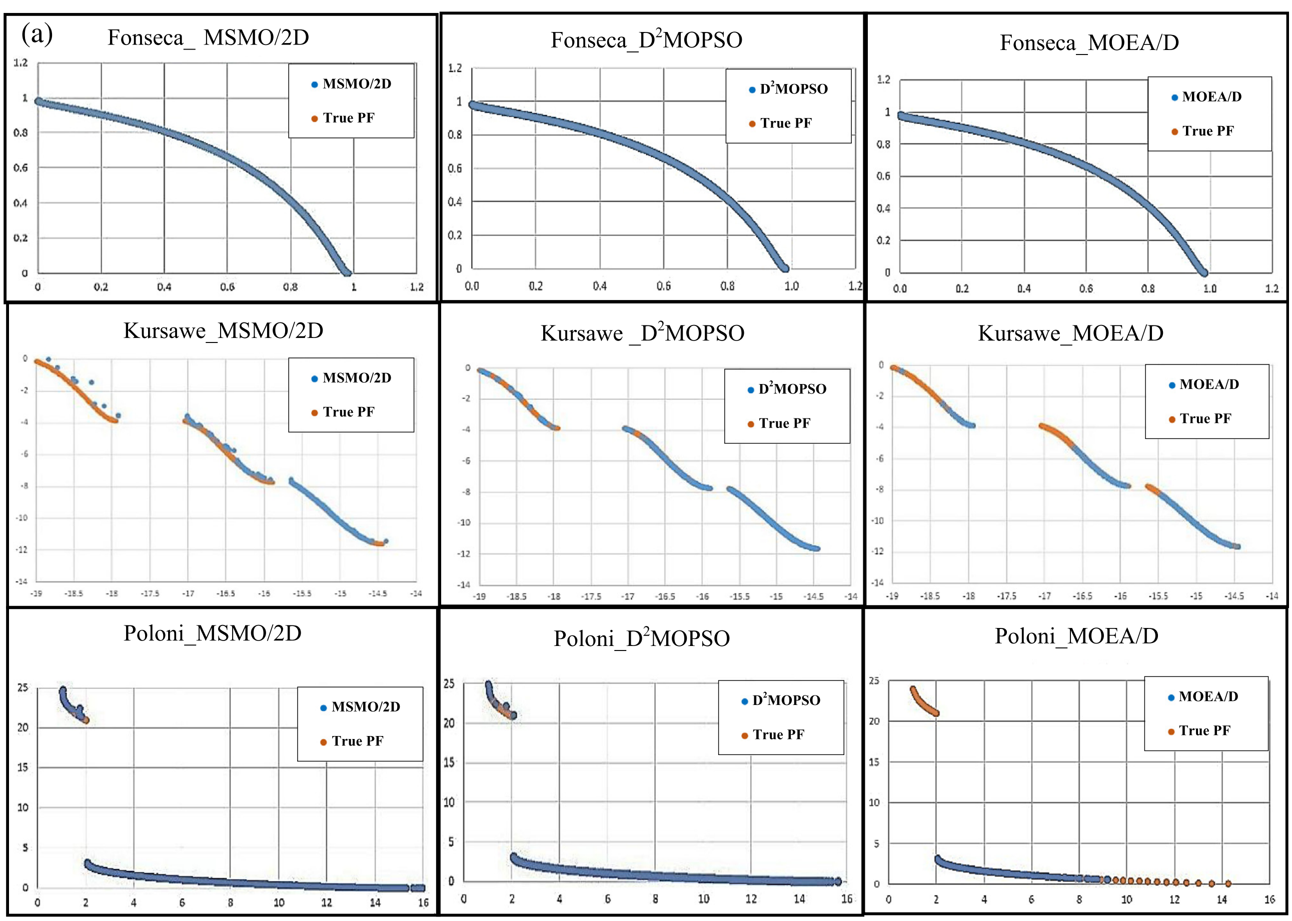


**Figure 3** Plot of two-objective problems using the three considered algorithms compared with its relative true PF approximation.

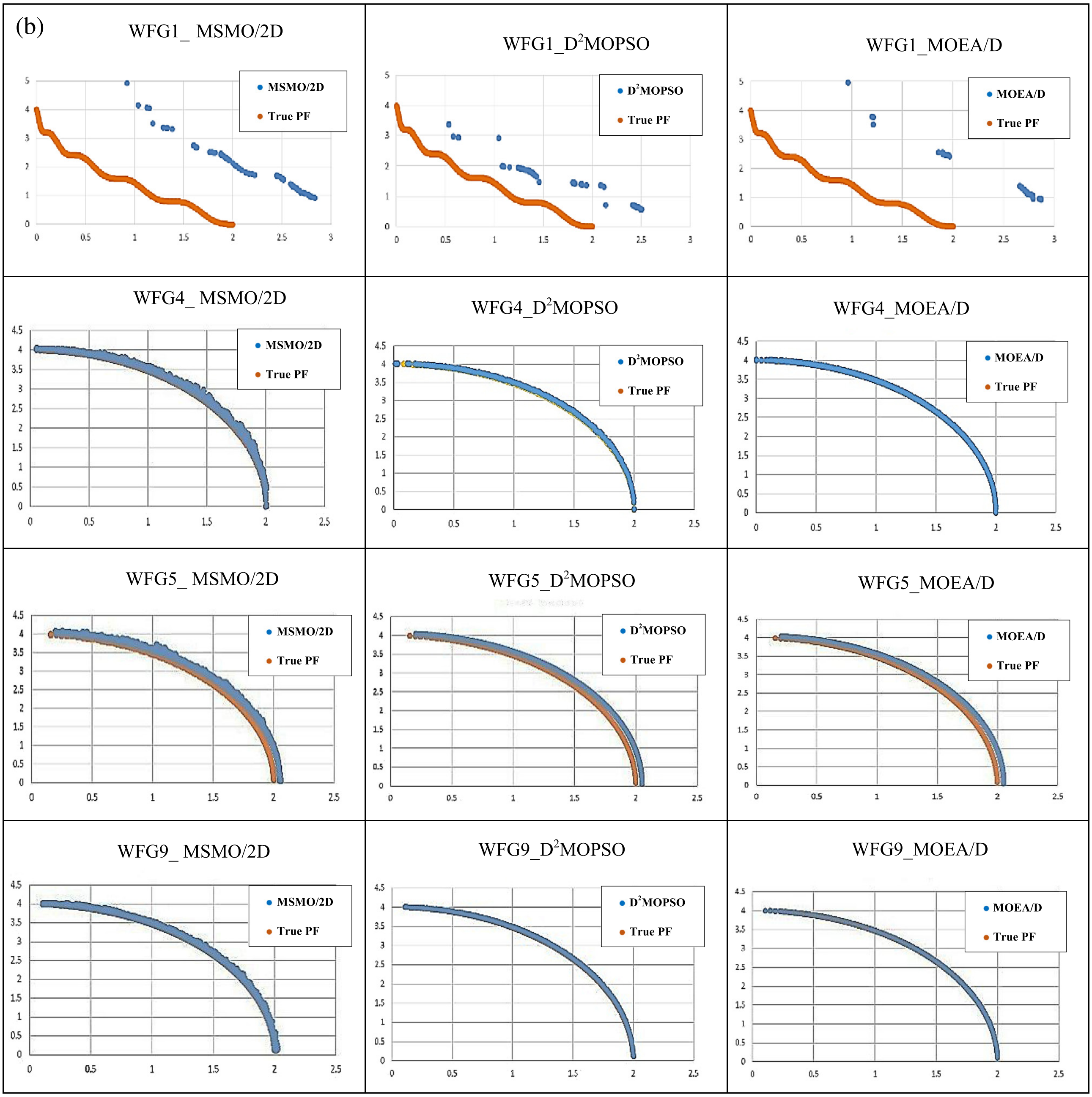


**Fig. 3** (*continued*)

proposed algorithm are illustrated. Finally, the obtained results are discussed in details.

### 5.1. Experiment setup

The proposed algorithm is applied to fourteen standard benchmark problems [28,35]. Problems with two-objective (Poloni, Fonseca, Kursawe, WFG1, 4, 5, 9), and problems with three-objective (DTLZ1 to DTLZ7) [42–46]. The swarm is divided into 4 sub-swarms. The performance of our algorithm MSMO/2D is compared with both D$^2$MOPSO and MOEA/D algorithms. They are considered as the most familiar algorithms that are used in solving MOP. The algorithm is run 30 times with 100 iterations and 100 particles for the two-objective problems. While, it is run 30 times with 300 iterations and 600 particles for the three-objective problems.

The PBI aggregation function with $\theta = 5$ is applied for all algorithms in all problems.

### 5.2. Performance indicators

Performance indicators can be classified based on their performance goals into three categories:

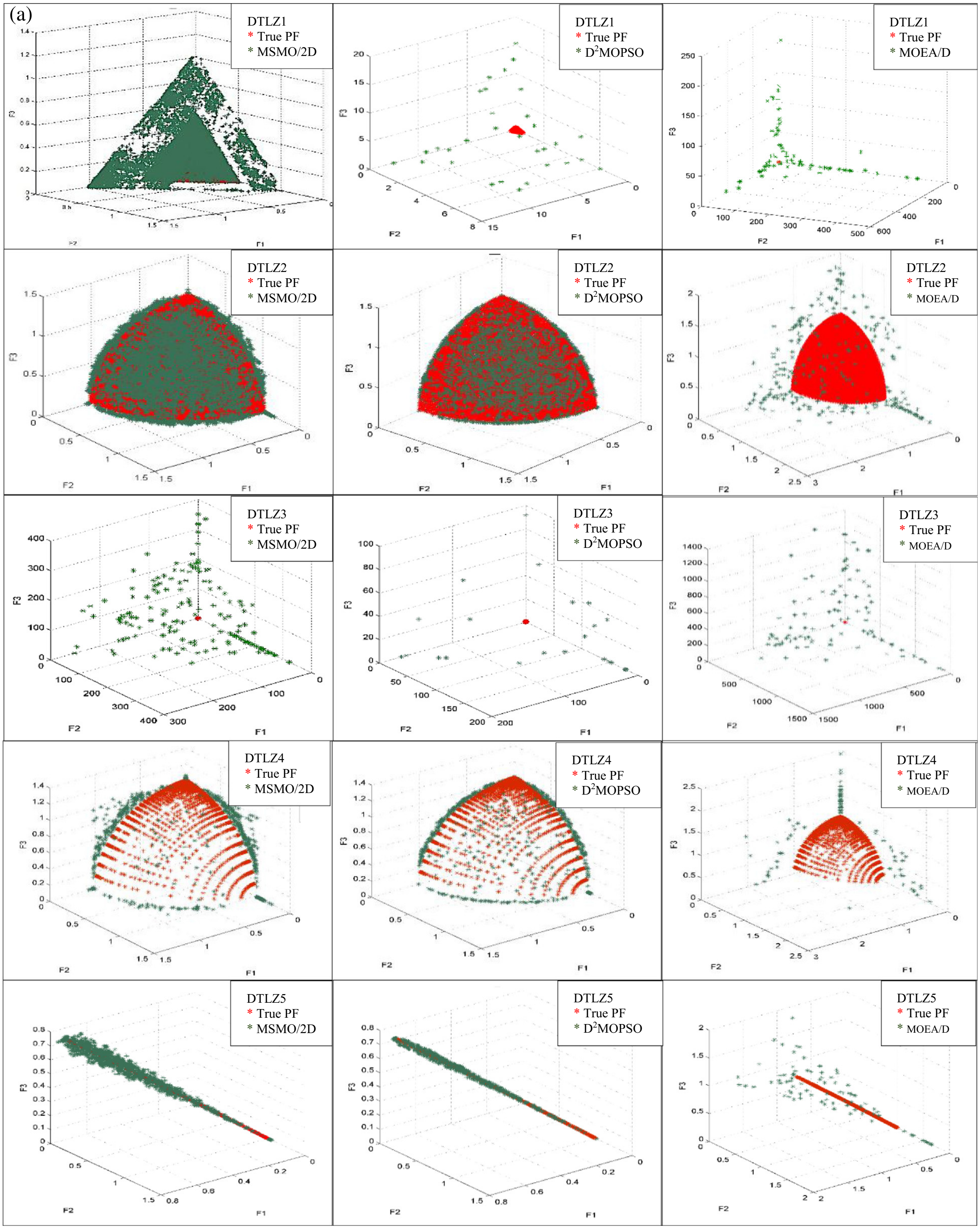


**Figure 4** Plot of three-objective problems using the three considered algorithms compared with its relative true PF approximation.

*5.2.1. Convergence-Based indicators*

Convergence-based indicators measure the convergence of the optimal PF like:

– Generational Distance ($I_{GD}$) - Contribution - Cardinality measure
– Inverted Generational Distance ($I_{IGD}$) - $\epsilon$-Indicator ($I_{\epsilon}$) - Distance measure

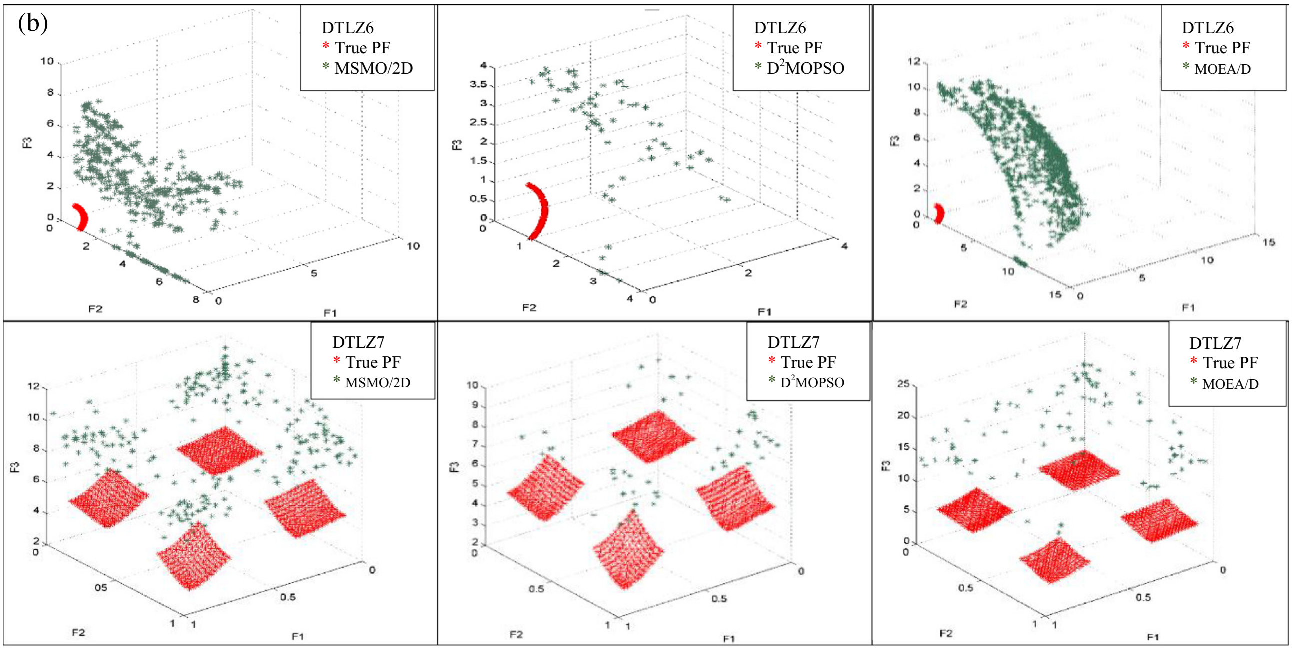


**Fig. 4** (*continued*)

**Table 1** $I_{Hv}$ results of the three algorithms.

| Problem | | MSMO/2D | D$^2$MOPSO | MOEA/D | Problem | | MSMO/2D | D$^2$MOPSO | MOEA/D |
|---|---|---|---|---|---|---|---|---|---|
| Fonseca | Md. | *3.17E−01* | 3.16E−01 | 3.12E−01 | DTLZ1 | Md. | 5.07E−01 | 5.11E−01 | *9.84E−01* |
| | Av. | *3.17E−01* | 3.16E−01 | 3.12E−01 | | Av. | 4.60E−01 | 4.72E−01 | *9.83E−01* |
| Poloni | Md. | *9.18E−01* | 9.06E−01 | 7.07E−01 | DTLZ2 | Md. | 4.49E−01 | 4.03E−01 | *7.56E−01* |
| | Ave. | *9.19E−01* | 9.04E−01 | 7.07E−01 | | Av. | 4.49E−01 | 4.04E−01 | *7.61E−01* |
| Kursawe | Md. | *4.05E−01* | 4.03E−01 | 1.76E−01 | DTLZ3 | Md. | 7.05E−01 | 2.09E−01 | *8.92E−01* |
| | Av. | *4.05E−01* | 4.03E−01 | 2.22E−01 | | Av. | 7.07E−01 | 2.51E−01 | *8.92E−01* |
| WFG1 | Md. | *5.62E−01* | 3.94E−01 | 4.56E−01 | DTLZ4 | Md. | 3.08E−01 | 2.74E−01 | *6.38E−01* |
| | Av. | *5.74E−01* | 4.14E−01 | 4.38E−01 | | Av. | 3.04E−01 | 2.68E−01 | *6.40E−01* |
| WFG4 | Md. | 1.95E−01 | 2.04E−01 | *2.06E−01* | DTLZ5 | Md. | 1.21E−01 | 9.00E−02 | *5.63E−01* |
| | Av. | 1.95E−01 | 2.05E−01 | *2.06E−01* | | Av. | 1.14E−01 | 8.77E−02 | *5.61E−01* |
| WFG5 | Md. | *2.34E−01* | 2.30E−01 | 2.25E−01 | DTLZ6 | Md. | *7.74E−01* | 4.23E−01 | 4.90E−01 |
| | Av. | *2.33E−01* | 2.30E−01 | 2.25E−01 | | Av. | *7.78E−01* | 4.09E−01 | 4.93E−01 |
| WFG9 | Md. | *2.29E−01* | 2.26E−01 | 2.24E−01 | DTLZ7 | Md. | 2.38E−01 | 9.53E−02 | *6.45E−01* |
| | Av. | *2.29E−01* | 2.26E−01 | 2.24E−01 | | Av. | 2.61E−01 | 1.13E−01 | *6.50E−01* |

*5.2.2. Diversity-Based indicators*

In general, diversity-based indicators depend on cardinality, distance, or volume measurement. It calculates the uniformity of distribution of the obtained solutions through dispersion and extension as:

– Spread - Extent - Entropy

*5.2.3. Hybrid indicators*

The hybrid indicators merge convergence and diversity measurements in one indicator such as:

– Hypervolume ($I_{Hv}$) - R-metrics ($I_R$).

To compare the proposed algorithm with the other chosen algorithms, we consider the generational distance indicator ($I_{IGD}$) and the ϵ-Indicator ($I_\epsilon$) to measure convergence and the spread indicator ($I_{Spread}$) to measure diversity. In addition to, hypervolume ($I_{Hv}$) as a hybrid indicator. These indicators are ranked as the most used indicators [47].

*5.3. Discussion of results*

The results of the two-objective problems Fonseca, Poloni, Kursawe, and WFG1, 4, 5, 9 are displayed in Fig. 3a and b,

**Table 2** $I_{Spread}$ results of the three algorithms.

| Problem | | MSMO/2D | $D^2$MOPSO | MOEA/D | Problem | | MSMO/2D | $D^2$MOPSO | MOEA/D |
|---|---|---|---|---|---|---|---|---|---|
| Fonseca | Md. | 1.28E+00 | *8.21E−01* | 1.46E−01 | DTLZ1 | Md. | *1.10E+00* | 6.53E−01 | 4.97E−01 |
| | Av. | 1.29E+00 | *8.19E−01* | 1.46E−01 | | Av. | *1.12E+00* | 6.47E−01 | 4.95E−01 |
| Poloni | Md. | *1.34E+00* | 1.59E+00 | 5.26E−01 | DTLZ2 | Md. | 6.28E−01 | *6.37E−01* | 5.16E−01 |
| | Av. | *1.43E+00* | 1.54E+00 | 5.20E−01 | | Av. | 6.19E−01 | *6.40E−01* | 5.15E−01 |
| Kursawe | Md. | 1.40E+00 | *1.15E+00* | 6.02E−01 | DTLZ3 | Md. | 6.53E−01 | *6.57E−01* | 4.75E−01 |
| | Ave. | 1.39E+00 | *1.14E+00* | 5.97E−01 | | Av. | *6.67E−01* | 6.55E−01 | 4.74E−01 |
| WFG1 | Md. | *1.02E+00* | 1.20E+00 | 1.26E+00 | DTLZ4 | Md. | 1.04E+00 | *1.00E+00* | 1.63E+00 |
| | Ave. | *1.01E+00* | 1.21E+00 | 1.27E+00 | | Ave. | 1.01E+00 | *1.01E+00* | 1.63E+00 |
| WFG4 | Md. | *8.67E−01* | 7.93E−01 | 3.80E−01 | DTLZ5 | Md. | 6.89E−01 | *6.92E−01* | 5.24E−01 |
| | Av. | *8.64E−01* | 8.21E−01 | 3.95E−01 | | Ave. | 6.99E−01 | *7.10E−01* | 5.25E−01 |
| WFG5 | Md. | *9.73E−01* | 6.86E−01 | 4.45E−01 | DTLZ6 | Md. | *8.40E−01* | 8.33E−01 | 6.60E−01 |
| | Av. | *1.01E+00* | 6.97E−01 | 4.47E−01 | | Av. | *8.46E−01* | 8.34E−01 | 6.60E−01 |
| WFG9 | Md. | *7.85E−01* | 6.56E−01 | 3.72E−01 | DTLZ7 | Md. | *7.14E−01* | 6.78E−01 | 4.53E−01 |
| | Av. | *7.91E−01* | 6.69E−01 | 3.73E−01 | | Av. | *7.13E−01* | 6.93E−01 | 4.53E−01 |

**Table 3** $I_{IGD}$ results of the three algorithms.

| Problem | | MSMO/2D | $D^2$MOPSO | MOEA/D | Problem | | MSMO/2D | $D^2$MOPSO | MOEA/D |
|---|---|---|---|---|---|---|---|---|---|
| Fonseca | Md. | 1.28E−05 | 8.13E−06 | *5.17E−06* | DTLZ1 | Md. | *1.56E−03* | 5.81E−02 | 7.28E−03 |
| | Av. | 1.98E−05 | 7.89E−06 | *5.14E−06* | | Av. | *3.80E−03* | 6.87E−02 | 6.95E−03 |
| Poloni | Md. | *1.61E−05* | 1.69E−05 | 9.04E−05 | DTLZ2 | Md. | *2.76E−04* | 9.76E−04 | 6.36E−03 |
| | Av. | *2.35E−05* | 2.52E−05 | 1.03E−04 | | Av. | *3.61E−04* | 9.88E−04 | 6.10E−03 |
| Kursawe | Md. | *1.89E−05* | 3.28E−05 | 1.05E−01 | DTLZ3 | Md. | *4.99E−03* | 6.71E−02 | 1.18E−02 |
| | Av. | *2.03E−05* | 3.33E−05 | 1.06E−01 | | Av. | *6.01E−03* | 6.48E−02 | 1.14E−02 |
| WFG1 | Md. | *1.54E−03* | 1.01E−02 | 1.16E−02 | DTLZ4 | Md. | *2.03E−03* | 8.42E−03 | 1.33E−02 |
| | Av. | *1.79E−03* | 1.08E−02 | 1.15E−02 | | Av. | *2.37E−03* | 8.38E−03 | 1.26E−02 |
| WFG4 | Md. | *1.21E−04* | 4.13E−04 | 2.87E−04 | DTLZ5 | Md. | *3.52E−04* | 1.18E−03 | 8.52E−03 |
| | Av. | *1.57E−04* | 4.00E−04 | 2.86E−04 | | Av. | *5.18E−04* | 1.48E−03 | 8.45E−03 |
| WFG5 | Md. | *2.40E−05* | 2.65E−05 | 1.29E−04 | DTLZ6 | Md. | 2.43E−03 | 1.76E−02 | *1.93E−03* |
| | Av. | 3.98E−05 | *2.67E−05* | 1.31E−04 | | Av. | 3.00E−02 | 1.86E−02 | *1.89E−03* |
| WFG9 | Md. | *3.80E−05* | 1.19E−04 | 1.17E−04 | DTLZ7 | Md. | *6.78E−03* | 3.70E−02 | 1.55E−02 |
| | Av. | *4.96E−05* | 1.14E−04 | 1.16E−04 | | Av. | *8.16E−03* | 4.18E−02 | 1.49E−02 |

while Fig. 4a and b displays the results of the selected three-objective problems DTLZ1 to DTLZ7. All problems are optimized using our proposed algorithm MSMO/2D, $D^2$-MOPSO, and MOEA/D algorithms. The results illustrate that the three algorithms are typically relative to the true PF curve in the majority of the testing problems except in WFG1 problem in which the obtained results of the three algorithms diver apparently from the true PF. However, in WFG5 problem our algorithm deviates slightly from the true PF unlike $D^2$-MOPSO and MOEA/D that record a clear deviation from the true PF curve while in Poloni problem, our algorithm covers the major part of the true curve. Fig. 4a and b confirms that our algorithm is the most intensive and the closest approximation to the true PF. Evidently, the proposed algorithm outperforms the other two algorithms especially in problems DTLZ1, 2, and 7.

Tables 1–4 compare the performance of our algorithm with both $D^2$MOPSO and MOEA/D based on the chosen performance indicators $I_{Hv}$, $I_{Spread}$, $I_{IGD}$ and $I_{\epsilon}$ respectively. Indicators are represented using the statistics values median (Md.) and average (Av.). The italics and underlined numbers represent the winner in the corresponding performance metric. The comparison results demonstrate that our proposed algorithm MSMO/2D exceeds the other algorithms relative to three of the chosen indicators in five problems (Poloni, WFG1, WFG5, WFG9, and DTLZ1). Also, it outperforms the other

**Table 4** $I_\epsilon$ results of the three algorithms.

| Problem | | MSMO/2D | $D^2$MOPSO | MOEA/D | Problem | | MSMO/2D | $D^2$MOPSO | MOEA/D |
|---|---|---|---|---|---|---|---|---|---|
| Fonseca | Md. | 1.88E−02 | *3.46E−03* | 5.47E−01 | DTLZ1 | Md. | *3.77E−01* | 5.84E+00 | 2.21E+01 |
| | Av. | 2.22E−02 | *4.18E−03* | 4.98E−01 | | Av. | *5.36E−01* | 5.92E+00 | 2.29E+01 |
| Poloni | Md. | 1.04E+00 | 9.56E−01 | *2.17E−02* | DTLZ2 | Md. | 1.04E−01 | *6.71E−02* | 2.84E−01 |
| | Av. | 9.44E−01 | 8.00E−01 | *2.41E−02* | | Av. | 1.03E−01 | *6.40E−02* | 2.81E−01 |
| Kursawe | Md. | 5.12E−01 | *8.67E−02* | 3.40E+00 | DTLZ3 | Md. | 1.09E+02 | *8.44E+01* | 2.24E+02 |
| | Av. | 5.39E−01 | *9.51E−02* | 3.24E+00 | | Av. | 1.07E+02 | *8.19E+01* | 2.09E+02 |
| WFG1 | Md. | 3.16E−01 | 3.48E−01 | *2.59E−01* | DTLZ4 | Md. | 2.79E−01 | *1.83E−01* | 5.43E−01 |
| | Av. | 3.14E−01 | 3.69E−01 | *2.51E−01* | | Av. | 2.61E−01 | *1.79E−01* | 5.64E−01 |
| WFG4 | Md. | 8.34E−02 | 4.47E−02 | *3.00E−02* | DTLZ5 | Md. | 5.31E−02 | *2.79E−02* | 2.10E−01 |
| | Av. | 9.17E−02 | 4.66E−02 | *3.00E−02* | | Av. | 5.73E−02 | *3.41E−02* | 2.06E−01 |
| WFG5 | Md. | 5.38E−02 | *1.30E−02* | 2.19E−02 | DTLZ6 | Md. | 1.91E+00 | 1.10E+00 | *9.34E−01* |
| | Av. | 6.50E−02 | *1.32E−02* | 1.94E−02 | | Av. | 1.86E+00 | 1.14E+00 | *9.02E−01* |
| WFG9 | Md. | 4.30E−02 | 1.65E−02 | *1.38E−02* | DTLZ7 | Md. | 2.24E+00 | *1.07E+00* | 1.60E+00 |
| | Av. | 6.97E−02 | 1.62E−02 | *1.29E−02* | | Av. | 2.36E+00 | *1.21E+00* | 1.57E+00 |

**Table 5** overview of the performance indicators of the three algorithms.

| Problem | MSMO/2D | $D^2$MOPSO | MOEA/D | Problem | MSMO/2D | $D^2$MOPSO | MOEA/D |
|---|---|---|---|---|---|---|---|
| Fonseca | $I_{Hv}$ | $I_{Spread}$, $I_\epsilon$ | $I_{IGD}$ | DTLZ1 | $I_{IGD}$, $I_{Spread}$, $I_\epsilon$ | – | $I_{Hv}$ |
| Poloni | $I_{Hv}$, $I_{IGD}$, $I_{Spread}$ | – | $I_\epsilon$ | DTLZ2 | $I_{IGD}$ | $I_{Spread}$, $I_\epsilon$ | $I_{Hv}$ |
| Kursawe | $I_{Hv}$, $I_{IGD}$ | $I_{Spread}$, $I_{Spread}$ | – | DTLZ3 | $I_{IGD}$, $I_{Spread}$($Av.$) | $I_{Spread}$ ($Md.$), $I_\epsilon$ | $I_{Hv}$ |
| WFG1 | $I_{Hv}$, $I_{IGD}$, $I_{Spread}$ | – | $I_\epsilon$ | DTLZ4 | $I_{IGD}$ | $I_{Spread}$, $I_\epsilon$ | $I_{Hv}$ |
| WFG4 | $I_{IGD}$, $I_{Spread}$ | – | $I_{Hv}$, $I_\epsilon$ | DTLZ5 | $I_{IGD}$ | $I_{Spread}$, $I_\epsilon$ | $I_{Hv}$ |
| WFG5 | $I_{Hv}$, $I_{IGD}$ ($Md.$), $I_{Spread}$ | $I_{IGD}$ ($Av.$), $I_\epsilon$ | – | DTLZ6 | $I_{Hv}$, $I_{Spread}$ | – | $I_{IGD}$, $I_\epsilon$ |
| WFG9 | $I_{Hv}$, $I_{IGD}$, $I_{Spread}$ | – | $I_\epsilon$ | DTLZ7 | $I_{IGD}$, $I_{Spread}$ | $I_\epsilon$ | $I_{Hv}$ |

algorithms for at least two indicators in five problems (DTLZ3, 6, 7, Kursawe, and WFG4). In DTLZ2, 4, 5, and Fonseca problems, our proposed algorithm MSMO/2D is competing the other algorithms in at least one indicator. Table 5 summarizes the performance indicators results of the considered algorithms.

## 6. Conclusion

This paper utilizes the multi-swarm optimization armed by the hybrid strategy to optimize MOP. Multi-swarm optimization gives a significant improvement in the population diversity by applying multiple sub-swarms rather than one standard swarm. Our proposed MSMO/2D algorithm is enhanced by adopting both decomposition and dominance strategies.

The proposed MSMO/2D algorithm outperforms $D^2$MOPSO and MOEA/D in most of the considered problems, concerning hypervolume, spread, IGD, and $\epsilon$ performance indicators. The results illustrate that the proposed algorithm MSMO/2D is very competitive especially in Poloni, WFG1, WFG9, and DTLZ1, DTLZ7 problems. The results obtained in this paper are very encouraging for future research. Therefore, we intend to test the proposed algorithm on a variety of different real-life problems and apply the hybrid strategy on other metaheuristics either evolutionary or swarm intelligence algorithms.

## References

[1] E. Zitzler, K. Deb, L. Thiele, C. Coello, D. Corne, Proceedings of the First Evolutionary Multi Criterion Optimization, (EMO-01) Conference Lecture Notes in Computer Science (LNCS) 1993, Springer, Heidelberg, 2001.

[2] C. Coello, D.A. VanVeldhuizen, G. Lamont, Evolutionary Algorithms for Solving Multi-Objective Problems, Kluwer, Boston, MA, 2002.

[3] A. Osyczka, Evolutionary algorithms for single and multicriteria design optimization, Physica-Verlag, Heidelberg, 2002.

[4] C. Fonseca, P. Fleming, E. Zitzler, K. Deb, L. Thiele, Proceedings of the Second Evolutionary MultiCriterion Optimization, (EMO-03) Conference, Lecture Notes in Computer Science (LNCS) 2632, Springer, Heidelberg, 2003.

[5] C. Coello, A.H. Aguirre, E. Zitzler, Evolutionary Multi-Criterion Optimization, Third International Conference (LNCS 3410), Springer, Berlin, Germany, 2005.

[6] S. Obayashi, K. Deb, C. Poloni, T. Hiroyasu, T. Murata, Evolutionary Multi-Criterion Optimization, 4th International Conference, EMO 2007, Matsushima, Japan, March 5-8, 2007, Proceedings, Lecture Notes in Computer Science, vol. 4403 of. Springer, 2007.

[7] I. Alaya, C. Solnon, K. Ghedira, Ant Colony Optimization for Multi-Objective Optimization Problems, Proceeding ICTAI

2007, in: 19th IEEE International Conference, vol. 01, 2007, pp. 450–457.

[8] J. Falcón-Cardona, C. Coello, A new indicator-based many-objective ant colony optimizer for continuous search spaces, Swarm Intell. 11 (1) (2017) 71–100.

[9] J. Bai, H. Liu, Multi-objective Artificial Bee Algorithm Based on Decomposition by PBI Method, Springer, Science + Business Media, New York, 2016.

[10] C. Coello, N. Cortes, Solving Multiobjective Optimization Problems Using an Artificial Immune System, CINVESTAV-IPN, Evolutionary Computation Group, Depto. de Ingeniería Electrica, Sección de Computación, Av. Instituto Politecnico Nacional No. 2508, Col. San Pedro Zacatenco, México, D. F. 07300, Mexico 2005.

[11] Kaveh, Particle Swarm Optimization, Advances in Metaheuristic Algorithms for Optimal Design of Structures, Springer, pp 11–43, 2016.

[12] M. Sierra, C. Coello, Multi-objective particle swarm optimizers: a survey of the state-of-the-art, Int. J. Comput. Intell. Res. 2 (3) (2006) 287–308.

[13] J. Moore, R. Chapman, Application of Particle Swarm to Multiobjective Optimization, Auburn University Department of Computer Science and Software Engineering, 1999.

[14] Q. Zhang, H. Li, MOEA/D: A multiobjective evolutionary algorithm based on decomposition, IEEE Trans. Evol. Comput. 11 (6) (2007) 712–731.

[15] W. Peng, Q. Zhang, A decomposition-based multi-objective particle swarm optimization algorithm for continuous optimization problems, in: IEEE, International Conference on Granular Computing, Hangzhou, 2008.

[16] N. Al Moubayed, A. Petrovski, J. McCall, A novel smart particle swarm optimisation using decomposition, in: PPSN XI, Lecture Notes in Computer Science, Springer, 2010.

[17] S. Martínez, C. Coello, A multi-objective particle swarm optimizer based on decomposition, in: Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation, GECCO '11, ACM, 2011.

[18] S. Mostaghim, J. Teich, Strategies for finding good local guides in multi-objective particle swarm optimization (MOPSO), in: Proceedings of the 2003 IEEE Swarm Intelligence SymposDebium, IEEE Service Center, pages 26–33, Indianapolis, Indiana, USA, April 2003.

[19] D. Srinivasan, T. Seow, Particle swarm inspired evolutionary algorithm (PS-EA) for multiobjective optimization problem, in: Congress on Evolutionary Computation (CEC'2003), IEEE Press, volume 3: 2292–2297, Canberra, Australia, December 2003.

[20] S. Mostaghim, J. Teich, The role of $\epsilon$- dominance in multi objective particle swarm optimization methods, in: Congress on Evolutionary Computation (CEC'2003), IEEE Press, volume 3, Canberra, Australia, December 2003, pp. 1764–1771.

[21] C. Coello, G. Pulido, M. Lechuga, Handling multiple objectives with particle swarm optimization, IEEE Trans. Evol. Comput. 8 (3) (June 2004) 256–279.

[22] M. Sierra, C. Coello, Improving PSO-based multi-objective optimization using crowding, mutation and $\epsilon$-dominance, in: Third International Conference on Evolutionary Multi-Criterion Optimization, EMO 2005, Springer-Verlag, LNCS 3410, Guanajuato, Mexico, 2005, pp. 505–519.

[23] N. Al Moubayed, A. Petrovski, J. McCall, D2MOPSO: MOPSO based on decomposition and dominance with archiving using crowding distance in objective and solution spaces, J. Evol. Comput. Archive 22 (1) (2014) 47–77, MIT Press Cambridge, MA, USA, ACM.

[24] Y. Zhang, D. Gong, Z. Ding, Handling multi-objective optimization problems with a multi-swarm cooperative particle swarm optimizer, Expert Systems with Applications 38 (2011) 13933–13941.

[25] B. Mishra, S. Dehuri, S. Cho, Swarm Intelligence in Multiple and Many Objectives Optimization: A Survey and Topical Study on EEG Signal Analysis, Springer, Studies in Computational Intelligence, vol. 592, 2015, pp. 27–73.

[26] A. Haeri, K. Rezaie, S. Hatefi, Developing an approach for decision making units integration using multi objective particle swarm optimization, Int. J. Eng. Technol; Dubai 4 (1) (2015) 48–53.

[27] Z. Miljković, M. Petrović, Application of modified multi-objective particle swarm optimisation algorithm for flexible process planning problem, Int. J. Comput. Int. Manuf. 30 (2–3) (February 2017) 271–291.

[28] R. Cheng, Y. Jin, M. Olhofer, Test problems for large-scale multiobjective and many-objective optimization, IEEE Trans. Cybern. 99 (2016) 1–14.

[29] Y. Zhang, D. Gong, J. Cheng, Multi-objective particle swarm optimization approach for cost-based feature selection in classification, in: IEEE/ACM transactions on computational biology and bioinformatics/IEEE, ACM, vol. 14(1), 2015, pp. 64–75.

[30] P. Hu, L. Rong, C. Liang-lin, L. Li-xian, Multiple swarms multi-objective particle swarm optimization based on decomposition, Proc. Eng. 15 (2011) 3371–3375.

[31] N. Al Moubayed, A. Petrovski, J. McCall, D 2 MOPSO: multi-objective particle swarm optimizer based on decomposition and dominance, Evolutionary Computation in Combinatorial Optimization, Springer Lecture, Notes in Computer Science, vol. 7245, 2012, pp. 75–86.

[32] Q. Lin, J. Li, Z. Du, J. Chen, Z. Ming, A novel multi-objective particle swarm optimization with multiple search strategies, Eur. J. Oper. Res. 247 (2015) (2015) 732–744.

[33] Díaz-Manríquez, G. Toscano, J. Barron-Zambrano, E. Tello-Leal, R2-Based Multi/Many-Objective Particle Swarm Optimization, Hindawi Publishing Corporation, Computational Intelligence and Neuroscience, 2016.

[34] L. Li, W. Wang, W. Lia, X. Xua, Y. Zhaob, A novel ranking-based optimal guides selection strategy in MOPSO, Proc. Comput. Sci. 91 (2016) 1001–1010.

[35] Talbi, Metaheuristics - From Design to Implementation, Wiley, 2009.

[36] C. Coello, G. Lamont, D. Veldhuizen, Evolutionary Algorithms for Solving Multi-Objective Problems, Genetic and Evolutionary Computation, 2nd ed., Springer, New York, 2007.

[37] C. Fonseca, P. Fleming, Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization, in: fifth international conference on genetic algorithms, San Mateo, CA. Morgan Kaufmann, vol. 1, 1993, pp. 416–423.

[38] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA–II, IEEE Trans. Evol. Comput. 6 (2) (2002) 182–197.

[39] J. Knowles, D. Corne, Approximating the nondominated front using the Pareto archived evolution strategy, Evol. Comput. 8 (2000) 149–172.

[40] K. Deb, D. Goldberg, An Investigation of Niche and Species Formation in Genetic Function Optimization, 3rd Int. Conf. Gen. Alg., Morgan Kaufmann Publishers Inc, San Francisco, CA, USA, 1989.

[41] M. Laumanns, L. Thiele, K. Deb, E. Zitzler, Combining convergence and diversit in evolutionary multiobjective optimization, Evol. Comput 10 (3) (2002) 263–282.

[42] Poloni, Hybrid GA for multi objective aerodynamic shape optimization, Genetic algorithms in engineering and computer science, vol. 33, 1995, pp. 397–415.

[43] Fonseca, P. Flemming, Multiobjective optimization and multiple constraint handling with evolutionary algorithms - part ii: application example, IEEE Trans. Syst., Man, Cybern. 28 (1) (1998) 38–47.

[44] F. Kursawe, A variant of evolution strategies for vector optimization, in: H.P. Schwefel, R. Manner (Eds.), Parallel Problem Solving for Nature, Springer-Verlag, Berlin, Germany, 1990, pp. 193–197.

[45] S. Huband, P. Hingston, L. Barone, L. While, A review of multi-objective test problems and a scalable test problem toolkit, IEEE Trans. Evol. Comput. 10 (5) (2007) 477–506, IEEE.

[46] K. Deb, L. Thiele, M. Laumanns, E. Zitzler, Scalable Test Problems for Evolutionary Multi-Objective Optimization, Zurich, Switzerland, Tech. Rep. 112, 2001.

[47] N. Riquelme, C. Lücken, B. Baran, Performance metrics in multi-objective optimization, Computing Conference (CLEI), Latin American, 19-23 Oct. 2015.