A Study of Swarm Topologies and Their Influence on the Performance of Multi-Objective Particle Swarm Optimizers*

Diana Cristina Valencia-Rodríguez and Carlos A. Coello Coello[0000-0002-8435-680X]

CINVESTAV-IPN (Evolutionary Computation Group)
Av. IPN 2508, San Pedro Zacatenco
Ciudad de México, 07360, MEXICO
dvalencia@computacion.cs.cinvestav.mx
ccoello@cs.cinvestav.mx

Abstract. It has been shown that swarm topologies influence the behavior of Particle Swarm Optimization (PSO). A large number of connections stimulates exploitation, while a low number of connections stimulates exploration. Furthermore, a topology with four links per particle is known to improve PSO's performance. In spite of this, there are few studies about the influence of swarm topologies in Multi-Objective Particle Swarm Optimizers (MOPSOs). We analyze the influence of star, tree, lattice, ring and wheel topologies in the performance of the Speed-constrained Multi-objective Particle Swarm Optimizer (SMPSO) when adopting a variety of multi-objective problems, including the well-known ZDT, DTLZ and WFG test suites. Our results indicate that the selection of the proper topology does indeed improve the performance in SMPSO.

Keywords: Swarm topology · Particle swarm optimization · Multi-objective particle swarm optimization · Multi-objective optimization

1 Introduction

Particle Swarm Optimization (PSO) is a metaheuristic proposed in the mid-1990s by Kennedy and Eberhart [7] that mimics the social behavior of bird flocks and schools of fish. PSO searches a solution to an optimization problem using particles that move through the search space employing their best previous position and the best position of the particles to which that particle is connected. The graph that represents these connections is called *swarm topology*. It has been empirically shown that the topology influences the behavior of a single-objective PSO [6, 8]. A topology with many connections improves the exploitative

* The first author acknowledges support from CONACyT and CINVESTAV-IPN to pursue graduate studies in Computer Science. The second author gratefully acknowledges support from CONACyT grant no. 2016-01-1920 (*Investigación en Fronteras de la Ciencia 2016*) and from a SEP-Cinvestav grant (application no. 4).

behavior of PSO, while a topology with few connections improves its explorative behavior [6].

A wide variety of Multi-Objective Particle Swarm Optimizers (MOPSOs) have been developed [10] over the years. However, unlike the case for single-objective PSO, studies on the influence of a swarm topology in the performance of a MOPSO are very scarce. Yamamoto et al. [12] studied the influence of a swarm topology for the bi-objective problems ZDT1, ZDT3, and ZDT4. They found that increasing the topology connections improves the convergence towards the true Pareto Front, and that decreasing such connections promotes diversity. On the other hand, Taormina and Chau [11] examined the effect of a swarm topology for a bi-objective problem of neural networks training. They noticed that a topology with four connections (a lattice topology) improves the performance of a MOPSO. Both studies offer relevant information about the influence of a swarm topology. However, the results of these two studies are limited to bi-objective problems having similar features. In contrast, the study presented in this paper covers a wide variety of problems with two and three objectives, taken from the Zitzler-Deb-Thiele (ZDT), the Deb-Thiele-Laumanns-Zitzler (DTLZ) and Walking-Fish-Group (WFG) test suites.

The remainder of this paper is organized as follows. In Section 2, we provide some basic concepts related to multi-objective optimization and PSO, including swarm topologies. Then, in Section 3, we describe the operation of the Speed-constrained Multi-objective Particle Swarm Optimizer (SMPSO) which is our baseline algorithm. Section 4 presents a discussion on the use of topologies in MOPSOs. Section 5 presents two schemes for handling swarm topologies in MOPSOs, as a framework for conducting the study presented herein. Our experimental results are provided in Section 6. Finally, our conclusions and some potential paths for future research are provided in Section 7.

2 Background

2.1 Multi-objective Optimization

We are interested in solving a continuous unconstrained multi-objective optimization problem that is defined as follows:

$$\underset{\mathbf{x}\in\Omega}{\text{minimize}} \quad F(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \cdots, f_m(\mathbf{x}))^T \tag{1}$$

where $\mathbf{x} = [x_1, x_2, \cdots, x_n]^T$ belongs to the decision variable space defined by Ω. And $F(\mathbf{x}) : \Omega \rightarrow \mathbb{R}^m$ consist of m objective functions $f_i(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}$ that are usually in conflict. In a multi-objective problem, we aim to find the best trade-off solutions that can be defined in terms of the notion of Pareto Optimality. We provide the following definitions to describe this concept.

Definition 1. *Let $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^m$, $\boldsymbol{u}$ is said to* ***dominates*** $\boldsymbol{v}$ *(denoted by $\boldsymbol{u} \preceq \boldsymbol{v}$), if and only if $u_i \leq v_i$ for all $i = 1, ..., m$ and $u_i < v_i$ for at least one index $j \in \{1, \cdots, m\}$.*

Definition 2. *A solution $\boldsymbol{x} \in \Omega$ is* ***Pareto Optimal*** *if it does not exist another solution $\boldsymbol{y} \in \Omega$ such that $F(\boldsymbol{y}) \preceq F(\boldsymbol{x})$.*

Definition 3. *Given a multi-objective optimization problem $(F(\boldsymbol{x}), \Omega)$, the* ***Pareto Optimal Set*** *(PS) is defined by:*

$$PS = \{\boldsymbol{x} \in \Omega \mid \boldsymbol{x} \text{ is a Pareto Optimal solution}\},$$

and its image $PF = \{F(\boldsymbol{y}) \mid \boldsymbol{y} \in PS\}$ is called ***Pareto Front****.*

2.2 Particle Swarm Optimization

PSO is a bio-inspired metaheuristic that works with a set of particles (called *swarm*) that represents potential solutions to the optimization problem. Each particle $\mathbf{x}_i \in \mathbb{R}^n$ at generation t updates its position using the following expression:

$$\mathbf{x}_i(t) = \mathbf{x}_i(t-1) + \mathbf{v}_i(t). \tag{2}$$

The factor $\mathbf{v}_i(t)$ is called *velocity* and is defined by

$$\mathbf{v}_i(t) = w\mathbf{v}_i(t-1) + C_1 r_1(\mathbf{x}_{p_i} - \mathbf{x}_i(t-1)) + C_2 r_2(\mathbf{x}_{l_i} - \mathbf{x}_i(t-1)) \tag{3}$$

where w is a positive constant known as inertia weight; C_1 and C_2 are positive constants known as *cognitive* and *social* factors, respectively; r_1 and r_2 are two random numbers with a uniform distribution in the range $[0, 1]$; x_{p_i} is the best personal position found by the i^{th} particle , and x_{l_i} is the best particle to which it is connected (called *leader*). In order to define the connections that allow us to select the leader, we need to determine the topology of the swarm.

2.3 Swarm Topology

A swarm topology (or, simply, a topology) is a graph where each vertex represents a particle, and there is an edge between two particles if they influence each other [8]. The set of particles that affect a given particle is called *neighborhood*. In the experiments reported below, we use five topologies that have been studied before in PSO:

– **Fully connected** (star or *gbest*). All the particles in this topology influence each other [6]. See Fig. 1a. Therefore, the information between particles expands quickly.
– **Ring** (*lbest*). In this topology, each particle is influenced by its two immediate neighbors [6]. See Fig. 1b. For this reason, the information transmission between particles is slow.
– **Wheel**. It consists of one central particle that influences and is influenced by the remainder particles in the swarm [6]. See Fig. 1c. The central particle acts as a filter that delays the information.
– **Lattice**. In this topology, each particle is influenced by one particle above, one below and two on each side [6]. See Fig. 1d.
– **Tree**. The swarm in this topology is organized as a binary tree where each node represents a particle. See Fig. 1e.

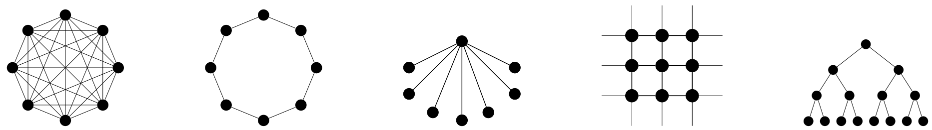


Fig. 1. Swarm topologies

3 SMPSO

In contrast to single-objective PSO, a MOPSO's particle could have more than one leader due to the nature of multi-objective problems. Therefore, a large number of MOPSOs usually store their leaders in an *external archive*, which retains the non-dominated solutions found so far [10]. For this reason, we assume in this paper that a MOPSO works with an external archive and selects the leaders from it. Accordingly, we selected for the experimental analysis a standard Pareto-based MOPSO that works in this manner: the Speed-constrained Multi-objective Particle Swarm Optimizer (SMPSO) [9]. The core idea behind SMPSO is to control the particles' velocity employing a constriction coefficient χ defined by:

$$\chi = 2/(2 - \varphi - \sqrt{\varphi^2 - 4\varphi}) \tag{4}$$

where $\varphi = 1$ if $C_1 + C_2$ is less or equal than four. Otherwise, $\varphi = C_1 + C_2$. Besides the constriction coefficient, SMPSO bounds the j^{th} velocity component of each i^{th} particle, denoted by $v_{i,j}(t)$, using the equation:

$$v_{i,j}(t) = \begin{cases} \delta_j & \text{if } v_{i,j}(t) > \delta_j \\ -\delta_j & \text{if } v_{i,j}(t) \leq -\delta_j \\ v_{i,j}(t) & \text{otherwise} \end{cases} \tag{5}$$

where $\delta_j = (upper\_limit_j - lower\_limit_j)/2$, and the upper and lower limits of the j^{th} decision variable are $upper\_limit_j$ and $lower\_limit_j$ respectively.

In summary, for computing the velocity, SMPSO selects the leader by randomly taking two solutions from the external archive and chooses the one with the largest crowding distance, which measures how isolated a particle is from the others. After that, the velocity is estimated with the selected leader using equation (3). Then, the result is multiplied by the constriction factor defined in equation (4) and bounded using the rule defined in equation (5).

SMPSO works in the following way. First, the swarm is randomly initialized, and the external archive is constructed with the non-dominated solutions currently available. During a certain (pre-defined) number of iterations, the velocity and position of each particle is computed. Then, polynomial-based mutation [1] is applied to the resulting individual, using a mutation rate p_m, and the new particle is evaluated. Finally, the particles' personal best and the external archive are updated. If the archive exceeds a pre-defined limit, the solution with the lowest crowding distance is removed. The pseudocode of SMPSO is shown in Algorithm 1.

Algorithm 1 Pseudocode of SMPSO

```
Initialize the swarm with random values
Initialize the external archive with the non-dominated solutions of the swarm
while the maximum number of iterations is not reached do
    for each particle p_i in the swarm do
        Randomly take two solutions from the external archive and select the one
        with the largest crowding distance as the leader x_{l_i}
        Compute the velocity using equation (3) and multiply it by equation (4)
        Constrain the velocity using equation (5)
        Compute the particle's position with equation (2)
        Apply polynomial-based mutation
        Evaluate the new particle
    end for
    Update the particle's memory and the external archive
    if the size of the external archive exceeds its limit then
        Remove from the external archive the particle with the lowest crowding
        distance
    end if
end while
```

4 Handling Topologies in Multi-Objective Particle Swarm Optimizers

In PSO, each particle updates its best personal position by comparing both the current and the previous positions and selecting the best one. Furthermore, the leader of each particle is selected by examining the best personal position of the particles to which it is connected. In multi-objective problems, however, we cannot select just one solution as the best. Therefore many MOPSOs store the best position found by particles in an external archive and select the leaders from it. This leader selection scheme does not allow MOPSOs to use distinct topologies because the neighborhood of a particle is not examined to select its leader. Moreover, many MOPSOs use a fully connected topology because each particle takes into consideration the positions found by the whole swarm. For this reason, it is necessary to design a leader selection scheme to handle swarm topologies in MOPSOs.

Yamamoto *et al.* [12] introduced a topology handling scheme where each particle had a sub-archive that was updated by the particle and its neighbors. Accordingly, each particle selected its leader from its sub-archive and the sub-archives of its neighbors. One advantage of this scheme is that it promotes diversity because a solution from a sub-archive could dominate a solution from another one. Furthermore, this scheme allows us to manipulate directly the best position found by the particles. On the other hand, one disadvantage of this scheme is that the space and time complexity of the MOPSO increase due to the use of sub-archives, and they get worse when the population size is increased.

Taormina and Chau [11] proposed another topology handling scheme where the leaders are added to the swarm. Each leader will influence four particles, but the particles will not influence the leaders, so they will not move. Taormina and Chau mentioned that these leaders are instances of non-dominated solutions found by the particles, but they do not provide any further information.

Due to the disadvantages of these two previously described schemes, we propose here two topology handling schemes which are described next.

5 The Proposed Topology Handling Schemes

In order to analyze the influence of the topology in MOPSOs, we propose two topology handling schemes and implement them in SMPSO. Both schemes differ only in the place from which the leader is taken: either the particle's memory or the external archive:

5.1 Scheme 1

The idea of scheme 1 is to emulate the leader selection scheme from a single-objective PSO. Therefore, it selects the leader of each particle by examining the personal best positions of the particles in the neighborhood and selecting the best from them. In order to implement scheme one in SMPSO (we called this algorithm SMPSO-E1), we modified line 5 of Algorithm 1. Thus, SMPSO-E1 obtains the particle's neighborhood and saves it in N_i. Next, it selects as a leader, the particle whose personal best position dominates most of the others in N_i. After that, the particle's position and its velocity are computed as in the original SMPSO.

5.2 Scheme 2

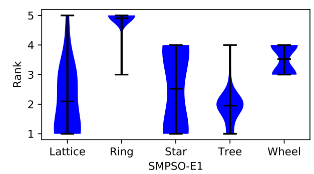
Under scheme 2, we associate each element of the external archive to each particle in the swarm, i.e., the i^{th} element of the external archive is associated with the i^{th} particle in the swarm. If the archive size is smaller than the swarm size, the archive elements are assigned again. Furthermore, the swarm size is restricted to be larger or equal to the archive size. Afterwards, a particle will select its leader by exploring the external archive components that are assigned to the particle's neighbors. The idea of scheme 2 is to use each external archive element as an alternative memory, in order to operate with the global best positions as leaders. In order to implement scheme 2 in SMPSO (we named this algorithm SMPSO-E2), we modified Algorithm 1. First, before computing the new positions of the particles, we assign the external archive elements to each particle. Then, for each particle, we randomly take two elements in the neighborhood and select as leader the one with the largest crowding distance. After that, we compute the particle's distance as in the original SMPSO.

6 Experiments and Analysis

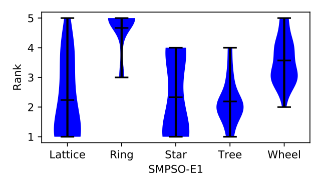
In this work, we compare five state-of-the-art topologies: star, ring, lattice, wheel, and tree. The influence of each topology is evaluated both using SMPSO-E1 and SMPSO-E2. We also contrast the performance of SMPSO-E1, SMPSO-E2, and the original version of SMPSO. In order to analyze the impact of a particular topology in the performance of a MOPSO, we adopted several test problems: the Zitzler-Deb-Thiele (ZDT) [14], the Deb-Thiele-Laumanns-Zitzler (DTLZ) [2], and the Walking Fish Group (WFG) [4] test suites. From the ZDT test suite, we excluded ZDT5 due to its discrete nature. We use 3-objective instances of DTLZ and WFG problems. The number of variables is 30 for ZDT1 to ZDT3, and 10 for ZDT4 and ZDT6. In the case of the DTLZ problems, the number of variables is $n = 3 + k - 1$, where $k = 5$ for DTLZ1, $k = 10$ for DTLZ2 to DTLZ6, and $k = 20$ for DTLZ7. Finally, we use 24 variables for the WFG problems.

For assessing performance, we selected three performance indicators: the hypervolume (HV) [13], the Modified Inverted Generational Distance (IGD$^+$) [5], and the s-energy [3]. The two first indicators assess both the convergence and the spread of the approximation set, while the third indicator measures only the diversity of the approximation set. The reference points used for the hypervolume, per problem, are the worst values found of the objective functions multiplied by 1.1.

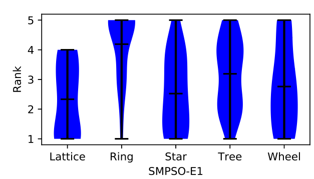
To ensure a fair comparison, we defined the same set of parameters for each MOPSO. We set the swarm and archive size to 100 for the ZDT problems and to 91 for the WFG and DTLZ test problems. The mutation probability was set to $p_m = 1/n$, and the inertia weight was set to $w = 0.1$. Moreover, the MOPSOs stop after performing 2500 iterations.



(a) Hypervolume



(b) IGD$^+$



(c) S-energy

Fig. 2. Distribution of ranks of SMPSO-E1 for each topology where rank 1 is the best and rank 5 is the worst.

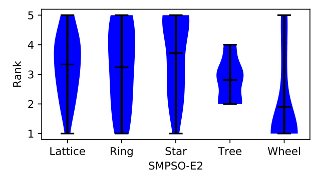
6.1 Methodology

We performed 30 independent runs of each MOPSO and normalized the resulting Pareto Front approximations. Then, we computed the indicators, normalized their values, and computed the corresponding means and standard deviations. Since we are dealing with stochastic algorithms, we also applied the Wilcoxon signed-rank test with a significance level of 5% to validate the statistical confidence of our results. We used the *mannwhitneyu* function from the `SciPy` `Python` library for this purpose.

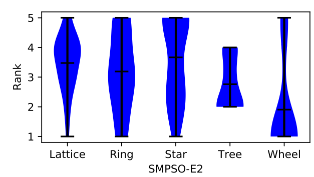
6.2 Experimental Results

Here, we present the comparison of SMPSO, SMPSO-E1, and SMPSO-E2 for each of the 5 topologies considered. Tables 1, 2, and 3 summarize the results for each indicator where the best values have a gray background, and the "*" symbol means that this result is statistically significant. Figs. 2 and 3 show the rank distribution among the topologies of SMPSO-E1 and SMPSO-E2, respectively. In this case, rank 1 is better than rank 5. In Fig. 2, the SMPSO-E1 with lattice, star, and tree topologies rank more frequently in the first places regarding the hypervolume and IGD$^+$ indicators. In contrast, the ring and wheel topologies rank more regularly in the last positions. This indicates that topologies with more connections promote the convergence of SMPSO-E1. Regarding the s-energy indicator, the lattice topology ranks more frequently in the best places, while the ring topology commonly ranks in the worst. Therefore, the lattice topology offers the best trade-off between convergence and diversity for SMPSO-E1.

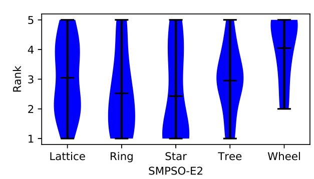
In the case of SMPSO-E2, we can see in Fig. 3 that the wheel topology ranks more frequently in the best places with respect to the hypervolume and IGD$^+$, followed by the tree topology, followed by the ring and lattice topologies, and finally, by the star topology. It is worth noting that topologies with fewer connections have better values in the convergence indicators. Regarding the s-energy indicator, the star topology ranks more frequently in the first places, followed by the ring topology, and then the tree and the lattice topologies. Ultimately, the wheel topology ranks more often in the worst places. In this case, we cannot define a topology that provides the best possible trade-off between convergence and diversity. Fig. 4 compares the performance of the MOPSOs in each problem. The blue and red connected lines denote the behavior of SMPSO-E1 and SMPSO-E2, respectively, for each topology. Furthermore, the green line represents the original SMPSO. We can see that in Figs. 4b and 4c, most of the blue lines are above the green and red lines. Conversely, in Fig. 4a, all the



(a) Hypervolume



(b) IGD$^+$

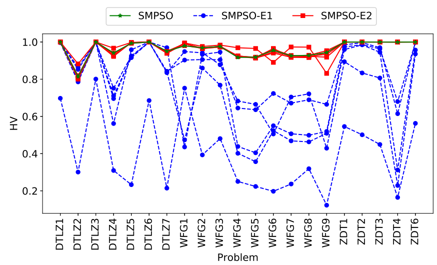


(c) S-energy

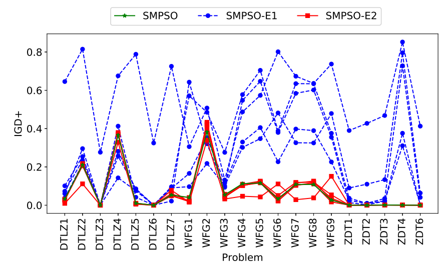
Fig. 3. Distribution of ranks of SMPSO-E2 for each topology where rank 1 is the best and rank 5 is the worst.

lines are below the green and red lines. Therefore, it is clear that SMPSO-E1 performs worse than SMPSO-E2 and SMPSO.

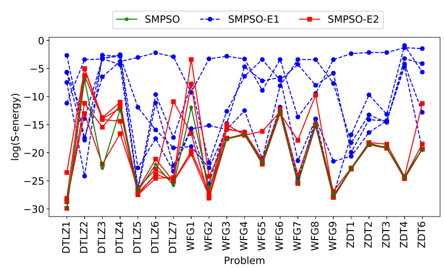
Finally, in Tables 1 and 2, we can see that SMPSO-E2 with a wheel topology has the best performance with respect to IGD$^+$ and the hypervolume. Besides, in Table 3, SMPSO-E2 with a star topology performs better with respect to s-energy, but the difference is not statistically significant.



(a) Hipervolumen



(b) IGD$^+$



(c) S-energy

Fig. 4. Indicator values of SMPSO, SMPSO-E1, and SMPSO-E2 for each problem. Lower values are preferred for s-energy and IGD$^+$, while higher values are preferred for the hypervolume.

7 Conclusions and Future Work

In this work, we proposed two topology handling schemes that differ in the place from which the leader is taken, and we implemented them in SMPSO. Moreover, using the resulting MOPSOs (SMPSO-E1 and SMPSO-E2), we performed an experimental analysis of the influence of the topology in the performance of a MOPSO.

The experiments show that a scheme that uses information from the external archive perform better than a scheme that uses information from the swarm. Furthermore, the same topology will influence the performance of a MOPSO in a different manner if the topology handling scheme is changed. On the other hand, our experiments also indicate that the fewer topology connections SMPSO-E2 has, the better its convergence is. This effect could be because the particles in a topology with many connections could try to go in multiple directions due to the existence of multiple optimal solutions, causing the MOPSO to converge. Conversely, if the topology has few connections, the information flows slowly and the particles move to specific optimal solutions. Furthermore, the wheel topology in SMPSO-E2 performs better than SMPSO and SMPSO-E1.

Therefore, the right selection of a topology can indeed improve the performance of a MOPSO. In SMPSO-E1 and SMPSO-E2, the swarm topology had little influence in the distribution of solutions. Thus, a topology handling scheme that focuses on this topic could be worth developing.

Table 1. Mean and standard deviation of the HV indicator for SMPSO, SMPSO-E1, and SMPSO-E2. The best values are highlighted in gray, and "*" indicates that the results are statistically significant

	SMPSO	SMPSO-E1					SMPSO-E2				
		Lattice	Ring	Star	Tree	Wheel	Lattice	Ring	Star	Tree	Wheel
DTLZ1	0.99999 (0.000)	0.99697 (0.003)	0.69748 (0.303)	0.99966 (0.001)	0.99967 (0.001)	0.99633 (0.014)	0.99999 (0.000)	0.99999 (0.000)	0.99999 (0.000)	0.99999 (0.000)	1.00000 (0.000)*
DTLZ2	0.81405 (0.038)	0.79588 (0.053)	0.30091 (0.135)	0.86375 (0.093)	0.85269 (0.069)	0.78544 (0.086)	0.81054 (0.041)	0.80032 (0.051)	0.80880 (0.037)	0.81901 (0.037)	0.88217 (0.064)
DTLZ3	1.00000 (0.000)	1.00000 (0.000)	0.80084 (0.254)	0.99999 (0.000)	1.00000 (0.000)	0.99999 (0.000)	1.00000 (0.000)	1.00000 (0.000)	1.00000 (0.000)	1.00000 (0.000)	1.00000 (0.000)
DTLZ4	0.94139 (0.019)	0.56177 (0.095)	0.30938 (0.150)	0.75153 (0.117)	0.71058 (0.119)	0.69577 (0.100)	0.93544 (0.016)	0.92248 (0.019)	0.93991 (0.017)	0.92746 (0.015)	0.96688 (0.015)*
DTLZ5	0.99326 (0.002)	0.95838 (0.006)	0.23308 (0.104)	0.92515 (0.024)	0.91583 (0.014)	0.91969 (0.020)	0.99601 (0.002)	0.99702 (0.001)	0.99270 (0.003)	0.99633 (0.001)	0.99760 (0.001)*
DTLZ6	0.99997 (0.000)	0.99993 (0.000)	0.68536 (0.277)	0.99955 (0.001)	0.99971 (0.000)	0.99920 (0.001)	0.99997 (0.000)	0.99997 (0.000)	0.99997 (0.000)	0.99997 (0.000)	0.99997 (0.000)
DTLZ7	0.95036 (0.027)	0.96958 (0.021)*	0.21446 (0.132)	0.83502 (0.055)	0.84259 (0.052)	0.83981 (0.065)	0.93912 (0.019)	0.95026 (0.027)	0.94900 (0.027)	0.94681 (0.018)	0.94085 (0.026)
WFG1	0.97934 (0.005)	0.43578 (0.217)	0.75196 (0.070)	0.94859 (0.015)	0.90301 (0.038)	0.47389 (0.217)	0.99078 (0.003)	0.99114 (0.002)	0.98043 (0.003)	0.99183 (0.002)	0.99495 (0.004)*
WFG2	0.96900 (0.010)	0.93333 (0.018)	0.39314 (0.170)	0.94195 (0.026)	0.90560 (0.027)	0.85999 (0.052)	0.96614 (0.010)	0.96408 (0.012)	0.96622 (0.012)	0.96427 (0.010)	0.97596 (0.014)*
WFG3	0.97586 (0.009)	0.94573 (0.036)	0.48072 (0.155)	0.87778 (0.048)	0.90471 (0.031)	0.76816 (0.144)	0.97749 (0.011)	0.97382 (0.012)	0.97226 (0.012)	0.97692 (0.009)	0.98341 (0.011)*
WFG4	0.91697 (0.020)	0.43842 (0.111)	0.25039 (0.110)	0.68273 (0.097)	0.64494 (0.065)	0.40172 (0.168)	0.92541 (0.016)	0.92125 (0.018)	0.92499 (0.020)	0.92351 (0.016)	0.96908 (0.020)*
WFG5	0.91891 (0.018)	0.40458 (0.127)	0.22354 (0.100)	0.66482 (0.070)	0.63603 (0.072)	0.35667 (0.153)	0.91564 (0.017)	0.91365 (0.019)	0.91434 (0.015)	0.91843 (0.016)	0.96516 (0.020)*
WFG6	0.95530 (0.059)	0.54999 (0.083)	0.19753 (0.085)	0.50558 (0.107)	0.72334 (0.067)	0.52365 (0.089)	0.96534 (0.017)	0.94079 (0.073)	0.94778 (0.060)	0.96042 (0.018)	0.89054 (0.100)
WFG7	0.92619 (0.017)	0.50668 (0.103)	0.23671 (0.101)	0.70426 (0.092)	0.67143 (0.059)	0.46856 (0.130)	0.91768 (0.015)	0.91782 (0.021)	0.92472 (0.018)	0.92754 (0.012)	0.97322 (0.015)*
WFG8	0.92835 (0.017)	0.49882 (0.101)	0.31892 (0.109)	0.72133 (0.082)	0.68940 (0.059)	0.46319 (0.111)	0.91636 (0.020)	0.92307 (0.015)	0.92857 (0.013)	0.92391 (0.017)	0.97240 (0.016)*
WFG9	0.94363 (0.057)	0.51928 (0.074)	0.12279 (0.063)	0.42907 (0.118)	0.66481 (0.086)	0.50926 (0.113)	0.94353 (0.054)	0.91859 (0.089)	0.95324 (0.016)	0.93372 (0.067)	0.83133 (0.114)
ZDT1	0.99974 (0.000)	0.98460 (0.006)	0.54580 (0.190)	0.89412 (0.037)	0.97277 (0.011)	0.95933 (0.035)	0.99980 (0.000)	0.99983 (0.000)	0.99972 (0.000)	0.99981 (0.000)	0.99989 (0.000)*
ZDT2	0.99977 (0.000)	0.99640 (0.001)	0.50140 (0.189)	0.83360 (0.072)	0.98613 (0.006)	0.98471 (0.010)	0.99982 (0.000)	0.99983 (0.000)	0.99976 (0.000)	0.99981 (0.000)	0.99982 (0.000)
ZDT3	0.99973 (0.000)	0.96979 (0.012)	0.44926 (0.190)	0.80671 (0.083)	0.96309 (0.016)	0.94488 (0.039)	0.99980 (0.000)	0.99982 (0.000)	0.99973 (0.000)	0.99982 (0.000)	0.99988 (0.000)*
ZDT4	0.99946 (0.000)	0.61466 (0.213)	0.16460 (0.109)	0.22890 (0.148)	0.67996 (0.198)	0.31222 (0.140)	0.99964 (0.000)	0.99970 (0.000)	0.99946 (0.000)	0.99970 (0.000)	0.99982 (0.000)*
ZDT6	0.99984 (0.000)	0.99970 (0.000)	0.56279 (0.222)	0.93559 (0.192)	0.99724 (0.003)	0.95678 (0.048)	0.99988 (0.000)	0.99992 (0.000)	0.99983 (0.000)	0.99990 (0.000)	0.99983 (0.000)

Table 2. Mean and standard deviation of the IGD$^+$ indicator for SMPSO, SMPSO-E1, and SMPSO-E2. The best values are highlighted in gray, and "*" represents that the results are statistically significant

	SMPSO	SMPSO-E1					SMPSO-E2				
		Lattice	Ring	Star	Tree	Wheel	Lattice	Ring	Star	Tree	Wheel
DTLZ1	0.03301	0.10113	0.64555	0.03768	0.07282	0.06023	0.02695	0.02469	0.03161	0.02813	0.00964
	(0.008)	(0.034)	(0.235)	(0.019)	(0.028)	(0.034)	(0.006)	(0.007)	(0.008)	(0.008)	(0.007)*
DTLZ2	0.21076	0.25408	0.81479	0.21706	0.23453	0.29559	0.21078	0.21764	0.20805	0.20568	0.11160
	(0.034)	(0.063)	(0.101)	(0.094)	(0.069)	(0.088)	(0.041)	(0.049)	(0.038)	(0.043)	(0.053)*
DTLZ3	0.00010	0.00025	0.27655	0.00039	0.00063	0.00050	0.00009	0.00009	0.00011	0.00008	0.00006
	(0.000)	(0.000)	(0.272)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)*
DTLZ4	0.36628	0.41288	0.67536	0.14286	0.25582	0.28146	0.36892	0.37913	0.36749	0.37046	0.32834
	(0.058)	(0.122)	(0.157)	(0.101)*	(0.117)	(0.098)	(0.036)	(0.045)	(0.050)	(0.032)	(0.045)
DTLZ5	0.01056	0.04157	0.78828	0.07523	0.08581	0.08063	0.00810	0.00711	0.01042	0.00645	0.00549
	(0.004)	(0.007)	(0.116)	(0.029)	(0.017)	(0.019)	(0.003)	(0.002)	(0.003)	(0.003)	(0.003)
DTLZ6	0.00011	0.00015	0.32484	0.00056	0.00041	0.00092	0.00011	0.00011	0.00011	0.00011	0.00011
	(0.000)	(0.000)	(0.307)	(0.001)	(0.000)	(0.001)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)
DTLZ7	0.05341	0.02154	0.72537	0.09052	0.08785	0.09626	0.06084	0.04887	0.05003	0.05035	0.07792
	(0.028)	(0.011)*	(0.168)	(0.048)	(0.041)	(0.056)	(0.017)	(0.027)	(0.020)	(0.017)	(0.025)
WFG1	0.04042	0.64259	0.30569	0.09692	0.16581	0.57108	0.02178	0.02101	0.04016	0.01833	0.01592
	(0.009)	(0.198)	(0.067)	(0.035)	(0.056)	(0.219)	(0.008)	(0.008)	(0.007)	(0.006)	(0.011)
WFG2	0.38268	0.32075	0.50751	0.21807	0.35269	0.48516	0.43306	0.42087	0.40543	0.42631	0.33968
	(0.110)	(0.065)	(0.148)	(0.115)*	(0.098)	(0.206)	(0.115)	(0.129)	(0.135)	(0.117)	(0.138)
WFG3	0.05400	0.09340	0.13147	0.10034	0.11614	0.27577	0.04774	0.04590	0.05720	0.04256	0.03254
	(0.023)	(0.072)	(0.059)	(0.095)	(0.070)	(0.196)	(0.020)	(0.022)	(0.025)	(0.023)	(0.018)*
WFG4	0.11260	0.54752	0.48730	0.30173	0.33005	0.57747	0.10234	0.10968	0.10192	0.10363	0.04687
	(0.030)	(0.124)	(0.102)	(0.091)	(0.071)	(0.220)	(0.030)	(0.029)	(0.029)	(0.030)	(0.034)*
WFG5	0.11549	0.64706	0.57343	0.34744	0.40533	0.70434	0.12136	0.12583	0.12115	0.11680	0.04322
	(0.020)	(0.138)	(0.070)	(0.072)	(0.069)	(0.165)	(0.023)	(0.023)	(0.019)	(0.023)	(0.025)*
WFG6	0.03305	0.37975	0.80127	0.48182	0.22680	0.38804	0.02238	0.05078	0.03640	0.02242	0.11099
	(0.063)	(0.076)	(0.107)	(0.105)	(0.067)	(0.081)	(0.008)	(0.087)	(0.062)	(0.010)	(0.120)
WFG7	0.10877	0.58401	0.67342	0.32610	0.39782	0.63484	0.11558	0.11913	0.10563	0.10341	0.02883
	(0.023)	(0.128)	(0.125)	(0.106)	(0.067)	(0.167)	(0.021)	(0.029)	(0.020)	(0.019)	(0.019)*
WFG8	0.10817	0.60173	0.63615	0.32510	0.38945	0.63409	0.12590	0.12015	0.11112	0.11776	0.03815
	(0.025)	(0.119)	(0.108)	(0.092)	(0.065)	(0.133)	(0.027)	(0.022)	(0.018)	(0.025)	(0.019)*
WFG9	0.02673	0.35521	0.73704	0.47878	0.22603	0.37548	0.02877	0.05392	0.01700	0.03652	0.15095
	(0.049)	(0.050)	(0.082)	(0.084)	(0.086)	(0.080)	(0.054)	(0.095)	(0.007)	(0.069)	(0.121)
ZDT1	0.00039	0.01323	0.39003	0.09011	0.02464	0.03646	0.00035	0.00033	0.00041	0.00031	0.00025
	(0.000)	(0.005)	(0.203)	(0.030)	(0.010)	(0.032)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)
ZDT2	0.00021	0.00251	0.42733	0.10997	0.00930	0.01021	0.00018	0.00015	0.00021	0.00018	0.00009
	(0.000)	(0.001)	(0.213)	(0.046)	(0.004)	(0.007)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)*
ZDT3	0.00103	0.02096	0.46837	0.13350	0.02279	0.03485	0.00093	0.00085	0.00106	0.00098	0.00078
	(0.000)	(0.007)	(0.214)	(0.057)	(0.008)	(0.026)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)
ZDT4	0.00063	0.37585	0.85215	0.79590	0.31050	0.72769	0.00052	0.00048	0.00063	0.00047	0.00043
	(0.000)	(0.234)	(0.130)	(0.174)	(0.215)	(0.158)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)
ZDT6	0.00019	0.00035	0.41284	0.06345	0.00269	0.04083	0.00014	0.00010	0.00017	0.00012	0.00020
	(0.000)	(0.000)	(0.224)	(0.192)	(0.003)	(0.044)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)

Table 3. Mean and standard deviation of the *s*-energy indicator for SMPSO, SMPSO-E1, and SMPSO-E2. The best values are highlighted in gray, and "•" represents that the results are statistically significant

	SMPSO	SMPSO-E1					SMPSO-E2				
		Lattice	Ring	Star	Tree	Wheel	Lattice	Ring	Star	Tree	Wheel
DTLZ1	1.141e-13 (0.000)	3.506e-03 (0.018)	7.015e-02 (0.239)	3.376e-03 (0.018)	5.690e-04 (0.002)	1.401e-05 (0.000)	3.346e-13 (0.000)	6.078e-13 (0.000)	4.183e-13 (0.000)	1.031e-13 (0.000)	6.109e-11 (0.000)
DTLZ2	9.007e-04 (0.005)	9.139e-07 (0.000)	2.869e-08 (0.000)	1.957e-08 (0.000)	3.223e-11 (0.000)	3.337e-02 (0.179)	1.391e-05 (0.000)	6.640e-03 (0.011)	2.229e-06 (0.000)	1.896e-03 (0.007)	6.427e-03 (0.011)
DTLZ3	1.355e-10 (0.000)	3.907e-02 (0.180)	1.544e-03 (0.008)	4.636e-02 (0.162)	7.300e-02 (0.110)	3.635e-02 (0.122)	1.965e-07 (0.000)	1.093e-06 (0.000)	2.665e-10 (0.000)	9.056e-07 (0.000)	7.648e-07 (0.000)
DTLZ4	4.563e-06 (0.000)	1.350e-02 (0.037)	2.343e-02 (0.053)	5.861e-02 (0.162)	5.994e-02 (0.192)	8.029e-02 (0.203)	9.265e-06 (0.000)	1.679e-05 (0.000)	6.177e-08 (0.000)	5.557e-06 (0.000)	5.653e-07 (0.000)
DTLZ5	2.277e-12 (0.000)	2.626e-12 (0.000)	4.879e-02 (0.188)	6.785e-06 (0.000)	4.160e-12 (0.000)	1.408e-10 (0.000)	2.918e-12 (0.000)	1.302e-12 (0.000)	3.706e-12 (0.000)	1.162e-12 (0.000)	1.657e-12 (0.000)
DTLZ6	2.565e-10 (0.000)	1.483e-05 (0.000)	1.115e-01 (0.290)	1.209e-07 (0.000)	6.510e-05 (0.000)	2.566e-08 (0.000)	6.661e-10 (0.000)	4.564e-11 (0.000)	1.127e-10 (0.000)	2.188e-11 (0.000)	3.971e-11 (0.000)
DTLZ7	6.239e-12 (0.000)	2.101e-10 (0.000)	5.481e-02 (0.205)	4.958e-09 (0.000)	3.105e-08 (0.000)	7.612e-11 (0.000)	1.291e-11 (0.000)	1.447e-11 (0.000)	1.245e-11 (0.000)	2.506e-11 (0.000)	1.847e-05 (0.000)
WFG1	6.707e-06 (0.000)	1.523e-07 (0.000)	9.602e-05 (0.000)	6.198e-09 (0.000)	3.914e-04 (0.002)	1.454e-07 (0.000)	2.714e-09 (0.000)	4.275e-04 (0.002)	1.621e-09 (0.000)	3.333e-02 (0.180)	6.297e-08 (0.000)*
WFG2	3.857e-12 (0.000)	7.653e-12 (0.000)	3.792e-02 (0.179)	1.370e-10 (0.000)	3.513e-10 (0.000)	2.660e-07 (0.000)	1.913e-12 (0.000)	6.343e-13 (0.000)	9.577e-13 (0.000)	3.085e-11 (0.000)	7.659e-13 (0.000)
WFG3	2.375e-08 (0.000)	3.806e-07 (0.000)	6.042e-02 (0.212)	1.434e-07 (0.000)	3.310e-06 (0.000)	1.295e-07 (0.000)	3.176e-07 (0.000)	2.546e-08 (0.000)	2.581e-08 (0.000)	2.816e-08 (0.000)	1.365e-07 (0.000)
WFG4	5.093e-08 (0.000)	9.281e-03 (0.050)	3.680e-02 (0.180)	8.530e-08 (0.000)	1.686e-03 (0.009)	3.754e-06 (0.000)	4.755e-08 (0.000)	4.844e-08 (0.000)	6.139e-08 (0.000)	4.854e-08 (0.000)	8.009e-08 (0.000)
WFG5	3.552e-10 (0.000)	7.542e-04 (0.004)	1.361e-04 (0.001)	5.059e-10 (0.000)	3.333e-02 (0.180)	7.000e-10 (0.000)	9.346e-08 (0.000)	3.491e-10 (0.000)	2.681e-10 (0.000)	5.375e-10 (0.000)	5.671e-10 (0.000)
WFG6	3.291e-06 (0.000)	1.376e-03 (0.007)	3.336e-02 (0.180)	7.085e-06 (0.000)	8.563e-04 (0.004)	3.146e-04 (0.002)	3.014e-06 (0.000)	2.469e-06 (0.000)	2.242e-06 (0.000)	2.407e-06 (0.000)	4.877e-06 (0.000)
WFG7	1.089e-11 (0.000)	4.965e-10 (0.000)	1.204e-06 (0.000)	2.080e-11 (0.000)	1.406e-02 (0.076)	3.333e-02 (0.180)	1.054e-11 (0.000)	8.564e-12 (0.000)	1.898e-08 (0.000)	1.033e-11 (0.000)	4.042e-11 (0.000)
WFG8	2.987e-07 (0.000)	8.504e-07 (0.000)	7.951e-05 (0.000)	8.527e-07 (0.000)	3.385e-04 (0.002)	3.336e-02 (0.180)	2.860e-07 (0.000)	2.381e-07 (0.000)	5.952e-05 (0.000)	2.796e-07 (0.000)	3.430e-07 (0.000)
WFG9	1.001e-12 (0.000)	4.561e-10 (0.000)	3.333e-02 (0.180)	1.009e-12 (0.000)	2.981e-03 (0.016)	4.756e-04 (0.003)	7.260e-13 (0.000)	7.276e-13 (0.000)	9.823e-13 (0.000)	7.737e-13 (0.000)	2.156e-12 (0.000)
ZDT1	1.222e-10 (0.000)	1.104e-09 (0.000)	9.597e-02 (0.208)	5.123e-08 (0.000)	2.253e-09 (0.000)	1.240e-08 (0.000)	1.165e-10 (0.000)	1.175e-10 (0.000)	1.130e-10 (0.000)	1.215e-10 (0.000)	1.374e-10 (0.000)
ZDT2	9.719e-09 (0.000)	7.584e-08 (0.000)	1.170e-01 (0.293)	6.223e-05 (0.000)	7.512e-07 (0.000)	1.764e-06 (0.000)	1.144e-08 (0.000)	9.838e-09 (0.000)	8.591e-09 (0.000)	9.851e-09 (0.000)	1.269e-08 (0.000)
ZDT3	5.414e-09 (0.000)	6.296e-07 (0.000)	1.171e-01 (0.221)	1.945e-06 (0.000)	5.979e-07 (0.000)	4.632e-07 (0.000)	4.847e-09 (0.000)	4.776e-09 (0.000)	4.701e-09 (0.000)	4.816e-09 (0.000)	9.529e-09 (0.000)
ZDT4	2.786e-11 (0.000)	9.603e-03 (0.027)	2.743e-01 (0.179)	4.082e-01 (0.249)	1.501e-02 (0.079)	3.949e-02 (0.108)	2.426e-11 (0.000)	2.049e-11 (0.000)	2.770e-11 (0.000)	2.624e-11 (0.000)	2.999e-11 (0.000)
ZDT6	4.016e-09 (0.000)	3.719e-09 (0.000)	2.354e-01 (0.291)	3.585e-03 (0.015)	2.762e-06 (0.000)	1.615e-02 (0.087)	4.053e-09 (0.000)	3.969e-09 (0.000)	3.975e-09 (0.000)	9.804e-09 (0.000)	1.334e-05 (0.000)

References

1. Deb, K., Agrawal, R.B.: Simulated Binary Crossover for Continuous Search Space. Complex Systems **9**, 115–148 (1995)
2. Deb, K., Thiele, L., Laumanns, M., Zitzler, E.: Scalable Test Problems for Evolutionary Multiobjective Optimization. In: Abraham, A., Jain, L., Goldberg, R. (eds.) Evolutionary Multiobjective Optimization: Theoretical Advances and Applications, pp. 105–145. Springer London, London (2005)
3. Hardin, D., Saff, E.: Discretizing Manifolds via Minimum Energy Points. Notices of the American Mathematical Society **51**(10), 1186–1194 (2004)
4. Huband, S., Barone, L., While, L., Hingston, P.: A Scalable Multi-objective Test Problem Toolkit. In: Evolutionary Multi-Criterion Optimization. pp. 280–295. Springer Berlin Heidelberg, Berlin, Heidelberg (2005)
5. Ishibuchi, H., Masuda, H., Tanigaki, Y., Nojima, Y.: Difficulties in Specifying Reference Points to Calculate the Inverted Generational Distance for Many-objective

Optimization Problems. In: 2014 IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making (MCDM'2014). pp. 170–177 (Dec 2014)
6. Kennedy, J.: Small Worlds and Mega-Minds: Effects of Neighborhood Topology on Particle Swarm Performance. In: Proceedings of the 1999 Congress on Evolutionary Computation (CEC'1999). vol. 3, pp. 1931–1938 (Jul 1999)
7. Kennedy, J., Eberhart, R.: Particle Swarm Optimization. In: Proceedings of the 1995 IEEE International Conference on Neural Networks (ICNN'1995). vol. 4, pp. 1942–1948 (1995)
8. Mendes, R.: Population Topologies and Their Influence in Particle Swarm Performance. Ph.D. thesis, Departamento de Informtica, Escola de Engenharia, Universidade do Minho (Apr 2004)
9. Nebro, A.J., Durillo, J.J., Garca-Nieto, J., Coello Coello, C.A., Luna, F., Alba, E.: SMPSO: A new PSO-based Metaheuristic for Multi-objective Optimization . In: 2009 IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making (MCDM'2009). pp. 66–73 (Mar 2009)
10. Parsopoulos, K.E., Vrahatis, M.N.: Multi-objective Particles Swarm Optimization Approaches . In: Multi-objective optimization in computational intelligence: Theory and practice, pp. 20 – 42. IGI Global (2008)
11. Taormina, R., Chau, K.: Neural Network River Forecasting with Multi-objective Fully Informed Particle Swarm Optimization. Journal of Hydroinformatics **17**(1), 99–113 (2014)
12. Yamamoto, M., Uchitane, T., Hatanaka, T.: An Experimental Study for Multi-objective Optimization by Particle Swarm with Graph Based Archive. In: Proceedings of SICE Annual Conference (SICE'2012). pp. 89–94 (Aug 2012)
13. Zitzler, E.: Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications. Ph.D. thesis, Swiss Federal Institute of Technology (ETH), Zurich, Suiza (Nov 1999)
14. Zitzler, E., Deb, K., Thiele, L.: Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. Evolutionary Computation **8**(2), 173–195 (2000)