Applied and Computational Mechanics 18 (2024) 77–110

Applied and Computational Mechanics

An improved vibrating particles system method for many-criteria engineering design applications

M. Nejlaoui[a,b*]

[a] *Department of Mechanical Engineering, College of Engineering, Qassim University, Buraydah 52571, Saudi Arabia*
[b] *Laboratory of Mechanical Engineering, National School of Engineers, University of Monastir, Tunisia*

Received 5 May 2023; accepted 24 June 2024

Abstract

Optimization is getting more and more important due to its application in real engineering problems. Recently, the vibrating particles system algorithm has been developed as an efficient method for mono-objective optimization. However, in multi- and many-objective design problems, the vibrating particles system method is unable to handle simultaneously the conflicting objectives. The second drawback of the vibrating particles system algorithm is the variability of the obtained results at each independent test, due to its inability to balance exploitation and exploration capabilities. To address these issues, this paper proposes an enhanced vibrating particles system algorithm called the many-objective vibrating particles system algorithm. The proposed many-objective vibrating particles system algorithm uses the Pareto principle to store the non-dominated solutions of multiple conflicting functions. Moreover, the implementation of the particle position enhancement mechanism to boost this algorithm's exploitation and exploration capabilities is another distinctive aspect of the suggested method. A variety of high-dimensional test functions and engineering design problems are used to evaluate the efficiency of the many-objective vibrating particles system algorithm. The obtained results show that the proposed algorithm outperforms other popular methods in terms of convergence characteristics and global search ability.

Keywords: metaheuristic, optimization problems, learning algorithms, optimization method, testing

1. Introduction

Optimizing multiples objectives is an issue encountered in most scientific and engineering applications [3]. These optimization problems are called multi-objective when there are two or three objectives, whereas they are called many-objective when there are more than three objectives. Several methodologies have been proposed to solve multi- and many-objective optimization problems over the last two decades. These methods are based on laws of nature, such as those found in the biological evolution, animal behavior, socio-political field and physical phenomena. Evolutionary algorithms utilize biological evolution-inspired ideas like crossover, mutation, and selection. The Pareto archived evolution strategy [23], the non-dominated sorting genetic algorithm III [40], a set-based genetic algorithm [19] and the improved differential evolutionary algorithm [39] are well-known evolutionary methods used in solving engineering problems.

Numerous animal behaviors, such as fishing and bird flock movements, are examined and mimicked as clever principles to create a variety of effective algorithms. The lifespan of water strider insects served as the model for the population-based algorithm known as the water strider algorithm [32]. This algorithm imitates water strider order, clever rippling interaction, breeding

*Corresponding author. Tel.: +966 565 696 267, e-mail: m.nejlaoui@qu.edu.sa.
https://doi.org/10.24132/acm.2024.835

Nomenclature			
BP	bad particle	MOMVO	multi-objective multi-verse optimization
DTLZ	Deb-Thiele-Laumanns-Zitzler test functions	MOSHO	spotted hyena optimizer
GP	good particle	MOVPS	multi-objective VPS
HB	best situation attained in the population	OHB	historically the best position in the population
HMOSHSSA	hybrid multi-objective spotted hyena salp swarm algorithm	S	non-dominated solution set
HV	hypervolume metric	SAMO	surrogate assisted multi-objective optimization
IBGA	incentives based grant allocation algorithm	SD	standard deviation
IGD	inverted generational distance metric	SP	spread metric
$Iter$	current iteration number	SPA	spacing metric
$Iter_{\max}$	total number of iterations	UF	unconstrained test functions
MaF	many-objective test functions	VPS	vibrating particles system
MOABHH	multi-objective agent-based hyper-heuristic algorithm	WFG	walking fish group test functions
		ZDT	Zitzler-Deb-Thiele's test functions

manner, and territorial actions. Moreover, the particle swarm optimization is a well-known method inspired from animal behavior and used in multi and many optimization engineering design [44]. This algorithm was motivated by fish schools or flocks of birds. The ant colony optimizer [40] is inspired by the method adopted by the colony of ants to reach their food. The imperialist competitive algorithm is a socio-political behavior of countries divided into imperialists and their colonies for expanding power or authority. The well-known versions of imperialist competitive methods are the multi-objective imperialist competitive algorithm [4] and the enhanced colonial method [46].

Natural physical rules are usually replicated by algorithms that are based on physical phenomena. The thermal exchange optimization algorithm [31] was based on the Newton's law of cooling indicating that a body rate of heat loss is directly related to the temperature differential between the body and its environment. The Vibrating Particles System (VPS) is a newly designed algorithm that is based on physical principles [36]. With viscous damping, this method simulates the unconstrained vibration of single degree of freedom systems. The theoretical structure of VPS is simple and its performance has been demonstrated through several engineering design problems [36]. VPS has been used in hard clustering problems [30], reinforced concrete shear walls [51], reservoir system operation [49], photovoltaic system [18], steel curved roof frames [34], buckling restrained brace frames [50] as well as other engineering design issues [38].

Despite its efficiency in mono-objective optimization and reduced calculation time [37], VPS has some drawbacks. The major drawback of VPS is the variability of the result at each independent test. This proves that the ability to balance exploitation (convergence capability) and exploration (explore the global regions of the search space) in VPS is not always successful [18]. Adding a mechanism for improving exploitation and exploration abilities can improve the

performances of the algorithm to reach the global optimum position in the search space. On the other hand, VPS cannot handle simultaneously multiple conflicting objectives in multi- and many-objective design problems [35].

In [33], Kaveh et al. used VPS to solve a multi-objective optimization problem by converting it to a mono-objective issue using the weighted sum of objectives. In fact, they used VPS to identify deterioration of truss systems. In this work, the selection of weighting factors is arbitrary and rarely justified. The ideal solution changes when these weighting parameters are changed [50]. Other authors suggested combining VPS with another method to address multi- and many-objective issues. For truss structures optimization, Kaveh and Ilchi [35] established the MDVC-VPS method which combines VPS and the Multi-Design Variable Configuration (MDVC).

Few researchers have concentrated on developing a multi-objective VPS version. In order to address various structural design issues, Kaveh and Ilchi [37] established a multi-objective variant of VPS that combines the VPS algorithm with the maximal strategy concept. To our knowledge, the literature research seems to be missing multi-objective versions of VPS. Moreover, no prior studies have proposed a mechanism to improve the VPS exploitation and exploration abilities.

The aim of this study is to develop a novel Multi-Objective Vibrating Particles System (MOVPS) algorithm based on the Pareto front concept. Moreover, in order to improve the algorithm's exploitation and exploration abilities, an enhancement mechanism of the particle position has been implemented at the end of the evaluation phase. In fact, the equations of the VPS algorithm have been updated and new parameters were added. The performances of MOVPS are tested through several high dimension test functions and a set of multi- and many-objective engineering design problems. The obtained results show that the proposed approach is efficient and is able to yield a wide spread of solutions with good convergence, diversity and robustness.

2. The proposed multi-objective vibrating particles system algorithm

2.1. The original VPS

This section describes the VPS algorithm. First, a brief overview of the free vibration of single degree of freedom systems with viscous damping is provided and then the proposed method is presented.

2.1.1. Brief concepts of VPS

Vibration is a mechanical phenomenon that causes oscillations around a fixed point. A mass spring damper system is used to analyze the damped vibrating motion, as shown in Fig. 1.

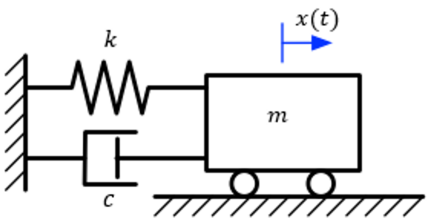


Fig. 1. Free vibration of a system with damping

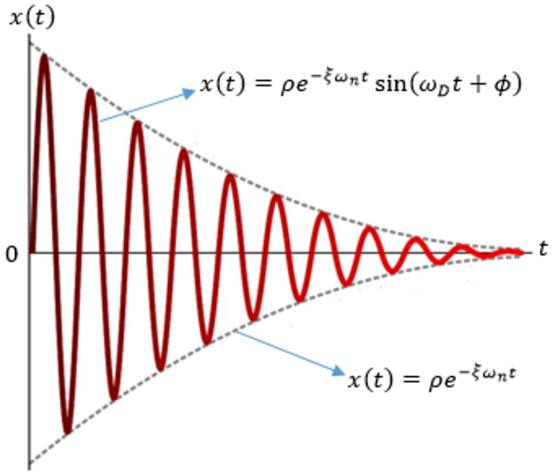


Fig. 2. Vibrating motion of an underdamped system

If the block of mass m is moved by distance x from its equilibrium point, the equation of motion can be expressed as

$$m\ddot{x} + c\dot{x} + kx = 0, \tag{1}$$

where c is the coefficient of viscous damping, and k is the spring stiffness. The solution to this equation is [48]

$$x(t) = \rho e^{-\xi\omega_n t} \sin\left(\omega_D t + \phi\right), \tag{2}$$

where ρ and ϕ are constants determined by the initial conditions of the problem, $\omega_n = \sqrt{k/m}$, $\omega_D = \omega_n\sqrt{(1-\xi^2)}$, and $\xi = c/(2m\omega_n)$ are the natural circular frequency, the damped natural frequency, and the damping ratio, respectively. The vibrating motion of the underdamped system (2) is presented in Fig. 2. An underdamped single degree of freedom system oscillates and returns to its equilibrium position. This kind of motion serves as inspiration for VPS.

2.1.2. VPS algorithm

The VPS method has been developed recently by Kaveh and Ilchi in [36]. Fig. 3 illustrates the VPS flowchart. The particles system is represented by a number of possible solutions in VPS. In the search space, the particles are randomly initiated and eventually approach their equilibrium positions. The different phases of VPS are presented in the following sections.

Phase 1 – Initialize VPS: The particle initial positions are produced at random

$$x_i^j = x_{\min} + rand\left(x_{\max} - x_{\min}\right), \tag{3}$$

where x_i^j is the j-th variable of particle i, and $rand$ is a random number uniformly distributed in the range of $[0, 1]$. The lowest and maximum permitted variable vectors are $x_{\min}$ and $x_{\max}$, respectively.

Phase 2 – Candidate solution evaluation: In this step, the value of the objective function for every particle is calculated.

Phase 3 – Update the particle location: For each particle, three equilibrium positions with differing weights are defined: (1) the historically best (HB) position of the entire population, (2) a good particle (GP), and (3) a bad particle (BP). The current population is sorted in ascending order according to the values of their objective functions in order to select GP and BP for

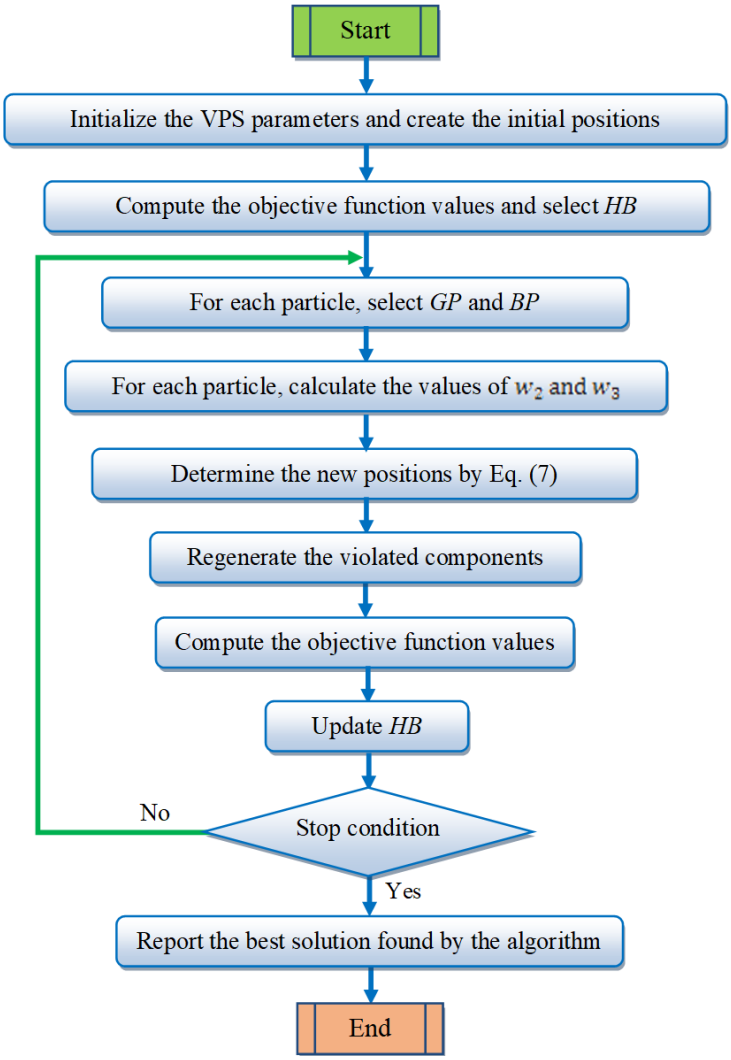


Fig. 3. The vibrating particles system flowchart

each candidate solution. Then, GP and BP are chosen randomly from the first and second half, respectively. The locations are changed to reflect the previously described ideas

$$x_i^j = w_1 \left(DA\,rand1 + \mathrm{HB}^j\right) + w_2 \left(DA\,rand2 + \mathrm{GP}^j\right) + w_3 \left(DA\,rand3 + \mathrm{BP}^j\right), \quad (4)$$

where $D = (Iter/Iter_{\max})^{-\alpha}$, $Iter$ is the current iteration number, $Iter_{\max}$ is the total number of iterations for the optimization process, and

$$A = \left[w_1 \left(\mathrm{HB}^j - x_i^j\right)\right] + \left[w_2 \left(\mathrm{GP}^j - x_i^j\right)\right] + \left[w_3 \left(\mathrm{BP}^j - x_i^j\right)\right], \quad (5)$$

where w_1, w_2, and w_3 $(w_1 + w_2 + w_3 = 1)$ are three parameters to measure the relative importance of HB, GP, and BP, respectively, $rand1$, $rand2$, and $rand3$ are random numbers uniformly distributed in the range of $[0, 1]$, and α is a constant value. Picking a value of 0.05 is advised [36]. The effect of A and D parameters in (4) is similar to that of ρ and $e^{-\xi\omega_n t}$ in (2), respectively. Also, the value of $\sin(\omega_D t + \phi)$ is considered unity in (4) $(x(t) = \rho e^{-\xi\omega_n t}$ are shown in Fig. 2 by interrupted lines).

A parameter like p within $(0, 1)$ is defined and it is specified that the effect of BP is sometimes ignored in an updating position. For each particle, p is compared with $rand$ and if $p < rand$, then $w_3 = 0$ and $w_2 = 1 - w_1$.

This algorithm takes into account three important concepts: competition, cooperation, and self-adaptation. Self-adaptation happens as a particle runs in the direction of HB. Cooperation between particles is offered in VPS, where every particle has the capacity to influence the new location of other particles. The competition is provided as the p parameter indicates that the influence of GP is greater than the influence of BP.

Phase 4 – Constraints handling: During the optimization process, the particle explores the search space in order to obtain a more accurate result. If any candidate solution violates the constraints, it must be regenerated by a penalty function approach [20]. Using this technique, unfeasible solutions are converted into a poor function (6) value. A constrained optimization problem can be turned into an unconstrained one by including a penalized term in the objective function

$$f(X) = \begin{cases} f(X) & \text{if } X \in F, \\ f(X) + penalty & \text{otherwise}, \end{cases} \tag{6}$$

where X represents the design vector of the system and F is the feasible search space. The penalty value was empirically chosen to be far larger than the objective function values generated by the problems under consideration. In fact, when the constraint is violated, the *penalty* is a high value number (here given as 10^{11}) that penalizes the objective function.

Phase 5 – Controlling the final criterion: Steps 2 to 4 are repeated until a termination criterion is satisfied. In this study, the optimization process is stopped after achieving a predetermined number of iterations.

2.2. The proposed MOVPS algorithm

In what follows, the Multi-Objective Vibrating Particles System algorithm (MOVPS, based on the original VPS) is proposed, see Fig. 4. Two main changes are performed on the original VPS:

- Incorporation of the particle position enhancement mechanism to improve the exploitation and exploration abilities of the algorithm.
- The fast non-dominated sorting strategy has been integrated into the algorithm to handle many conflicting objective functions.

The following paragraphs give details of the steps added to the original VPS.

2.2.1. The particle position enhancement mechanism

After the evaluation of candidate solutions, the particle position undergoes an enhancement phase. This enhancement mechanism improves convergence speed and algorithm search capability. "Memory" and "OHB" are two new parameters introduced in this step. Memory works in the same way as HB, only it saves the number of the historically best locations in the entire population. OHB (one of the historically greatest positions in the entire population) is one row of Memory that is randomly picked. In the MOVPS algorithm, HB is substituted by Memory. Another change in the VPS algorithm is that (4) should be replaced with $(7)_1$. In (7), one of the equations (a)–(c) is applied with the possibility of w_1, w_2 or w_3, respectively, i.e.,

$$x_i^j = \begin{cases} DA\,rand1 + \text{OHB}^j, & \text{(a)} \\ DA\,rand2 + \text{GP}^j, & \text{(b)} \\ DA\,rand3 + \text{BP}^j, & \text{(c)} \end{cases} \qquad A = \begin{cases} (\pm 1)\left(\text{OHB}^j - x_i^j\right), & \text{(d)} \\ (\pm 1)\left(\text{GP}^j - x_i^j\right), & \text{(e)} \\ (\pm 1)\left(\text{BP}^j - x_i^j\right), & \text{(f)} \end{cases} \tag{7}$$

where (± 1) is randomly used.

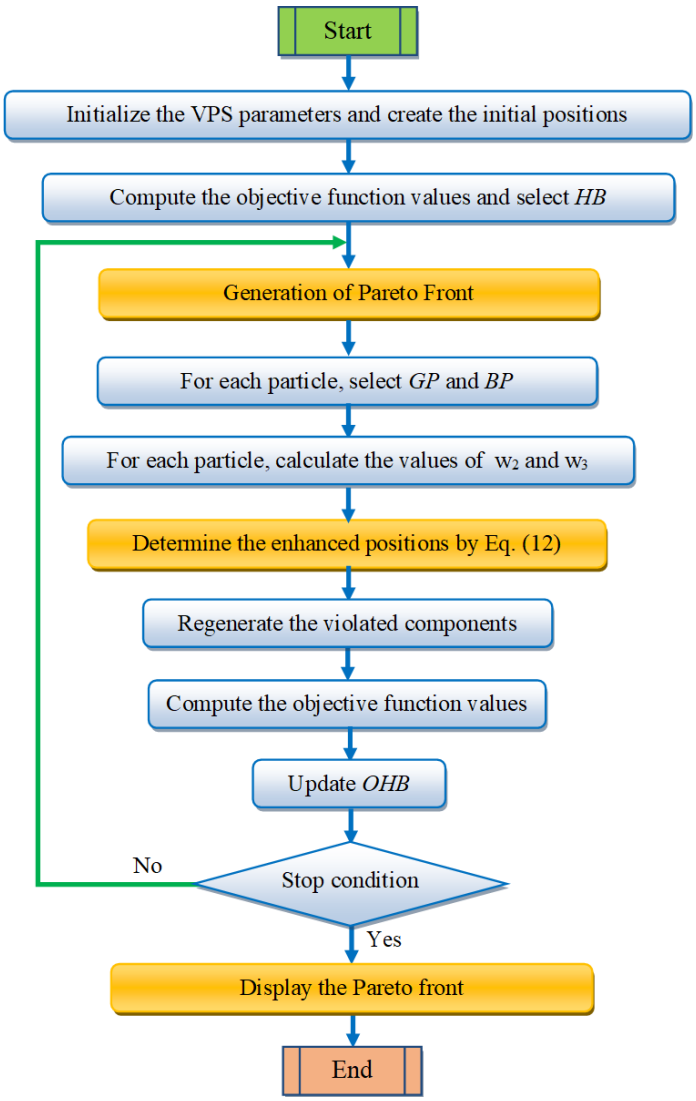


Fig. 4. The multi-objective vibrating particles system flowchart

2.2.2. *Generation of the Pareto front*

This phase is used to keep the non-dominated solutions, reflecting the Pareto front, in an archive by using the fast non-dominated solution approach [14]. During the initial iteration, N particles are chosen. Each particle i is contrasted with the remaining particles j, as illustrated in Fig. 5. The particle i counter variable (n_{Pi}) is augmented anytime it is dominated. The particle that has $n_{\mathrm{Pi}} = 0$ is kept in the archive. This process is done until the Pareto front is complete. In the remaining iterations, particles will be compared with those representing the precedent Pareto front (Fig. 5).

3. Experiments and results

3.1. *Performance metrics*

To evaluate the performances of MOVPS, five performance metrics will be used. These performance metrics are defined as follows [28, 47]:

- Inverted Generational Distance (IGD): The distance between the non-dominated solutions set

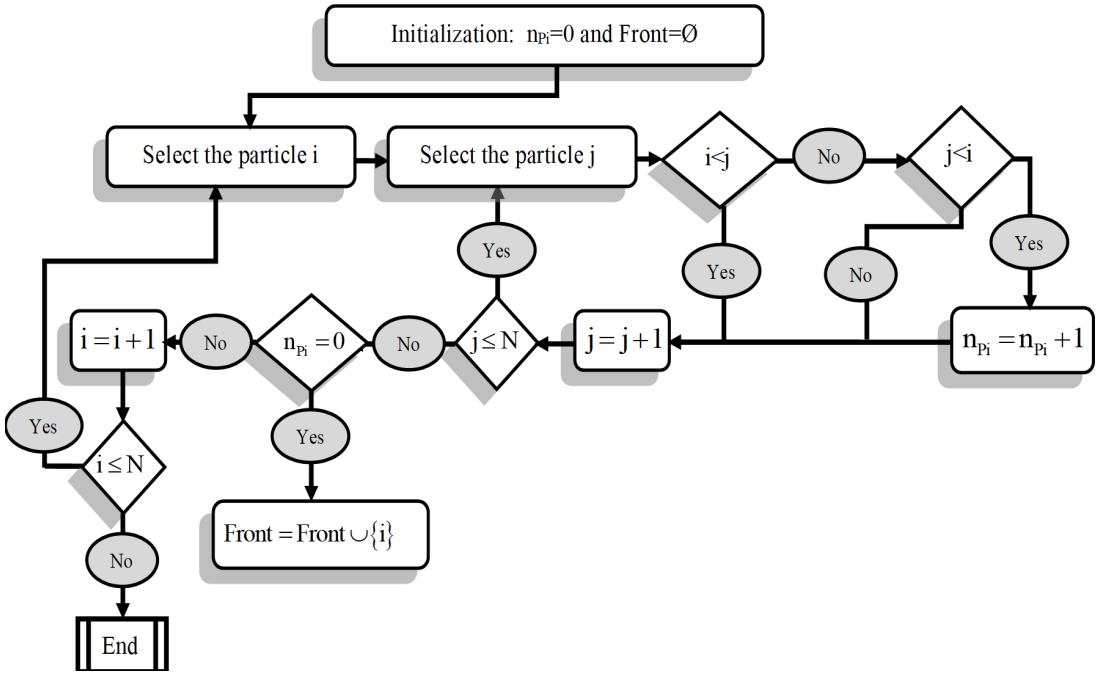


Fig. 5. Generation of the Pareto front

S and the distributed solutions over the true Pareto front P^* is measured by this metric, which can simultaneously show the convergence and the distribution of S along P^*. This metric is defined as [59]

$$\text{IGD}\,(S, P^*) = \frac{1}{|P^*|}\sum_{i=1}^{|P^*|}\min_{k=1}^{|S|} dist\left(p^i, s^k\right), \tag{8}$$

where $|P^*|$and $|S|$ are, respectively, the number of solutions in P^* and S given by the corresponding test functions, $dist\left(p^i, s^k\right)$ is the Euclidean distance between the solutions $p^i \in P^*$ and $s^k \in S$. It is noted that the set S with a low IGD value indicates a high approximation to the entire true Pareto front.

- Hypervolume (HV): This measures the volume surrounded by the obtained non-dominated solutions set S with a reference point r meeting $s^k < r,\ \forall k = 1, \ldots, |S|$. This metric is defined as [59]

$$\text{HV}(S, r) = volume\left[\bigcup_{k=1}^{N} HyperRectangle\left(s^k, r\right)\right], \tag{9}$$

where $HyperRectangle\left(s^k, r\right)$ designates a hyper-rectangle designed by the point r and the solution s^k, as illustrated in Fig. 6. The reference point $\boldsymbol{r} = (r, r, \ldots, r)$ is specified in the normalized objective space as follows [31]

$$r = 1 + \frac{1}{|S| - 1}, \tag{10}$$

where $|S|$ is the number of non-dominated solutions.

- Standard Deviation (SD): In order to assess the robustness of MOVPS for each test function, 10 independent runs (R) were conducted. The standard deviation, over the 10 runs, of the

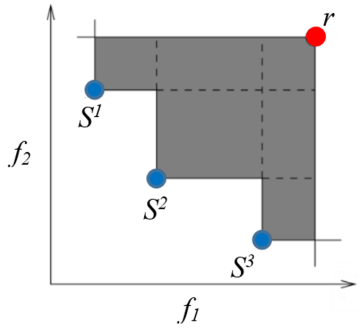


Fig. 6. $HyperRectangle\left(s^k, r\right)$ for two-objective optimization

obtained finding (Y) is calculated as follows

$$\mathrm{SD}(Y) = \sqrt{\frac{1}{R}\sum_{l=1}^{R}\left(Y_l - \overline{Y}\right)^2}, \tag{11}$$

where $\overline{Y}$ is the average of the obtained results during the 10 runs.

- Spacing (SPA): This metric indicates the evenness of the solutions dispersed along the Pareto front set [27]

$$\mathrm{SPA} = \left[\frac{1}{|S|}\sum_{k=1}^{|S|}\left(d_k - \overline{d}\right)^2\right]^{\frac{1}{2}}, \tag{12}$$

where d_k is the Euclidean distance in the Pareto front between the solution k and its closest subsequent solution, and $\overline{d}$ is the average of all distances between the closest solutions in S. The obtained non-dominated solutions have a more uniform distribution when SPA is lower.

- Spread (SP): This metric is used to calculate the extent of spread by the obtained solution set S

$$\mathrm{SP} = \frac{\sum_{k=1}^{|S|} d\left(s_{ex}^k, S\right) + \sum_{k=1}^{|S|}\left(d_k - \overline{d}\right)}{\sum_{k=1}^{|S|} d\left(s_{ex}^k.S\right) + |S|\,\overline{d}}, \tag{13}$$

where s_{ex}^k are the extreme non-dominated solutions in S and $d\left(s_{ex}^k, S\right)$ is the minimum Euclidean distance between s_{ex}^k and the points in S.

3.2. Test suites

Various well-known test suites were used to assess the performance of the MOVPS algorithm. These tests functions considered bi-objective (Zitzler–Deb–Thiele's and the unconstrained functions), tri-objective (Deb-Thiele-Laumanns-Zitzler and the Walking Fish Group), and many-objective (many-objective test functions) optimization problems of different natures.

The Zitzler-Deb-Thiele's (ZDT) test suite [14] offers two main advantages: The Pareto optimal fronts of its problems are well-defined and the test results from a variety of other research papers are commonly available, which facilitates comparisons with new algorithms.

The unconstrained functions (UF) [55] allow the discussion of a number of characteristics like multi-modality, bias, and Pareto front irregularity that add to the problems complexity. The

Deb-Thiele-Laumanns-Zitzler (DTLZ) test suites [15] are frequently used as a starting point for evaluating how well multi-objective algorithms work. DTLZ1 exhibits both multimodal and linear properties. The Pareto fronts of DTLZ2–6 are concave. The features of DTLZ7 are biased, discontinuous and mixed. The Walking Fish Group (WFG) test functions [25] are scalable issues in terms of both the number of objectives and the treated variables, as well as non-separable problems, deceiving problems, genuinely degenerative problems, and mixed form Pareto front problems. The many-objective functions (MaF) [12] are test suites for many-objective optimization that was recently proposed. Among the many characteristics of the MaF issues are their multimodality, degeneracy and bias. Verifying the efficacy of MOVPS on many-objective optimization problems with a range of real-world scenarios, is the goal of employing MaF.

3.3. Obtained results

3.3.1. Results of bi-objective test functions

The MOVPS performances are assessed using a series of bi-objective test functions: Zitzler-Deb-Thiele's and unconstrained test functions [14, 55]. The obtained results by MOVPS for the bi-objective test functions are illustrated in Fig. 18 (see Appendix). Tables 4–5 (see Appendix) provide a comparison between the MOVPS results and those obtained by other well-known algorithms.

It is noted that in every test function, the Pareto fronts produced by the proposed MOVPS closely mimic the true Pareto front solutions (Fig. 18). These results validate the MOVPS convergence efficiency. In particular, the effectiveness of MOVPS becomes apparent in ZDT3 and UF6 test function results, where the solutions in the Pareto front exhibit discontinuities. Moreover, for all bi-objective test functions, MOVPS provides smaller values of IGD, SPA and SP and higher values of the HV metric than those given in the literature (Tables 4–5). The results demonstrate that MOVPS outperforms other algorithms in terms of convergence and solution diversity. Indeed, the variance of the results yielded by MOVPS is smaller compared to the variances obtained from other algorithms (see Tables 4–5). These findings demonstrate the robustness of MOVPS.

3.3.2. Results of tri-objective test functions

The effectiveness of MOVPS is also evaluated in terms of managing issues with three conflicting objective functions: Deb-Thiele-Laumanns-Zitzler and Walking Fish Group (WFG) test functions [15, 25]. Nine common test functions with known Pareto fronts were chosen. The solutions for these test functions can vary in linear, spherical, curvilinear, or discontinuous forms.

Fig. 19 (see Appendix) displays the MOVPS results of tri-objective test functions as well as the true Pareto fronts. It can be noted that the Pareto fronts generated by the proposed MOVPS align with the true Pareto fronts in all tri-objective test functions. These results demonstrate the convergence efficiency of MOVPS when optimizing three conflicting functions.

The algorithm performances were also assessed by comparing them with the results from existing studies in the literature (Tables 6–7, see Appendix). It can be noted that the MOVPS algorithm has achieved better results in all metrics compared to the other well-known algorithms.

3.3.3. Results of many-objective test functions

The MOVPS performances were evaluated through a set of many-objective test functions (MaF) [12], as presented in Fig. 20 (see Appendix). The high convergence and solution diversity

abilities of the proposed MOVPS are demonstrated in all test functions, see Fig. 20. In fact, the non-dominated solutions are very close to the true Pareto front.

The algorithm effectiveness is also evaluated by comparing it to findings from previous research outlined in Table 8, see Appendix. It is noted that the proposed MOVPS surpasses the other algorithms in all metrics, demonstrating its high convergence, diversity and robustness for all many-objective test functions.

4. Engineering applications

In this section, the MOVPS algorithm will be used to solve three engineering design problems. These problems are the speed reducer design problem, the tri-objective vehicle crashworthiness design and the many-objective machining problem. After that, the obtained results are compared to those presented in the literature.

4.1. Speed reducer design problem

As shown in Fig. 7, the design of the speed reducer is considered having the face width (b), the module of the teeth (m), the number of teeth on pinion (z), the length of the first shaft between bearings (L_1), the length of the second shaft between bearings (L_2), diameter of the first shaft (da_1), and the diameter of the second shaft (da_2) [16, 29, 45]. The objective is to minimize the total weight of the speed reducer while satisfying eleven constraints. The constraints include the limits on the bending stress of the gear teeth, surface stress, transverse deflections of shafts 1 and 2 due to the transmitted force, and stresses in shafts 1 and 2.

The mathematical programming model of the speed reducer problem considered in this study is expressed as follows:

$$\text{Minimize} \begin{cases} F_1 = 0.7854\, bm^2 \left(\frac{10z^2}{3} + 14.933z - 43.0934\right) - 1.508b\left(da_1^2 + da_2^2\right) \\ \qquad +7.477\left(da_1^3 + da_2^3\right) + 0.7854\left(L_1 da_1^2 + L_2 da_2^2\right), \\ F_2 = \frac{1}{da_1^3}\sqrt{\left(745\frac{L_1}{mz}\right)^2 + 1.69\times 10^7}. \end{cases} \tag{14}$$

$$\text{Subject to} \begin{cases} g_1 = \frac{1}{bm^2 z} - \frac{1}{27} \le 0, & g_2 = \frac{1}{bm^2 z^2} - \frac{1}{397.5} \le 0, \\ g_3 = \frac{L_1^3}{mz\, da_1^4} - \frac{1}{1.93} \le 0, & g_4 = \frac{L_2^3}{mz\, da_2^4} - \frac{1}{1.93} \le 0, \\ g_5 = mz - 40 \le 0, & g_6 = \frac{b}{m} - 12 \le 0, \\ g_7 = 5 - \frac{b}{m} \le 0, & g_8 = 1.9 - L_1 + 1.5da_1 \le 0, \\ g_9 = 1.9 - L_2 + 1.1da_2 \le 0, & \\ g_{10} = \frac{1}{0.1da_1^3}\sqrt{\left(745\frac{L_1}{mz}\right)^2 + 1.69\times 10^7} - 1\,300 \le 0, & \\ g_{11} = \frac{1}{0.1da_2^3}\sqrt{\left(745\frac{L_2}{mz}\right)^2 + 1.575\times 10^8} - 1\,100 \le 0, & \end{cases} \tag{15}$$

where the search domains of the different design parameters are

$$\begin{array}{llll} 2.6 \le b \le 3.6, & 0.7 \le m \le 0.8, & 17 \le z \le 28, & 7.3 \le L_1 \le 8.3, \\ 7.3 \le L_2 \le 8.3, & 2.9 \le da_1 \le 3.9, & 5 \le da_2 \le 5.5. & \end{array} \tag{16}$$

Fig. 8 shows the optimization results obtained by the proposed MOVPS algorithm. According to this figure, one can note that the MOVPS solutions are very close to the true Pareto front.

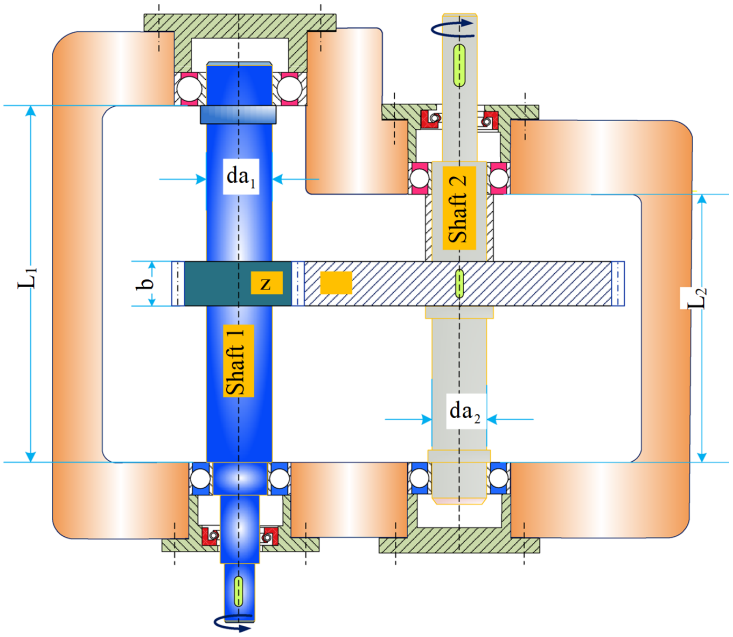


Fig. 7. Speed reducer problem design

This illustrates that the MOVPS approach shows effective convergence towards the true Pareto front while maintaining a high level of diversity in the obtained solutions.

The MOVPS results are compared to those given by the literature [2, 16, 29, 43, 45], as illustrated in Table 1. One can note that MOVPS yields better results than those given by the HMOSHSSA, MOSHO and MOMVO methods. In fact, MOVPS provides smaller values of IGD and higher values of HV.

4.2. Tri-objective vehicle crashworthiness design problem

This problem can be formulated as structural optimization of the frontal structure of a vehicle for crash-worthiness [2,5,29,43]. It contains three objectives which need to be minimized: mass of the vehicle (f_1), deceleration during the full frontal crash (f_2) and the toe board intrusion

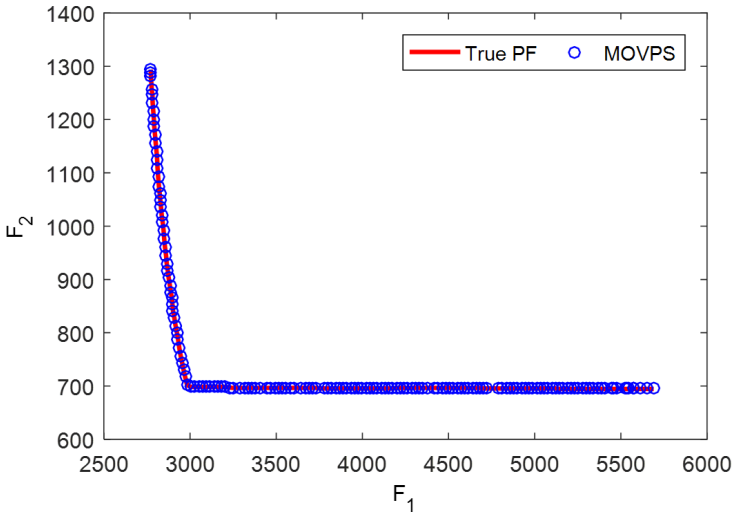


Fig. 8. Speed reducer optimization results

Table 1. IGD and HV metrics findings

Methods	IGD		HV	
	Mean	SD	Mean	SD
HMOSHSSA [52]	0.00312	0.00156	0.86200	0.22100
MOSHO [7]	0.00413	0.00157	0.32200	0.22000
MOMVO [6]	0.43067	0.04121	–	–
MOVPS	0.00271	0.00116	0.89715	0.00914

in the offset-frontal crash (f_3). The design variables are the thickness values of five reinforced members around the frontal structure of the vehicle (t_1, t_2, t_3, t_4, and t_5), as presented in Fig. 9. The design problem can be formulated as follows [5, 45]:

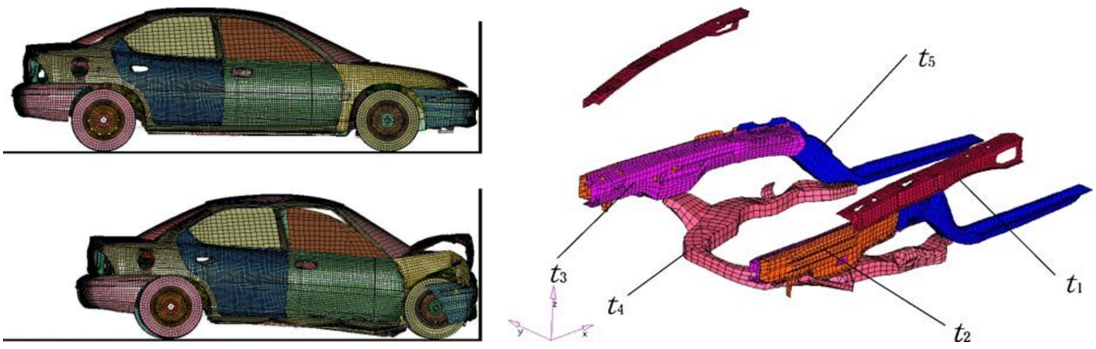
$$\text{Minimize} \begin{cases} f_1 = 1\,640.2823 + 2.3573285t_1 + 2.3220035t_2 + 4.5688768t_3 \\ \quad +7.7213633t_4 + 4.4559504t_5, \\ f_2 = 6.5856 + 1.15t_1 - 1.0427t_2 + 0.9738t_3 + 0.8364t_4 - 0.3695t_1t_4 \\ \quad -0.0861t_1t_5 + 0.3628t_2t_4 - 0.1106t_1^2 - 0.3437t_3^2 - 0.1764t_4^2, \\ f_3 = -0.0551 + 0.0181t_1 + 0.1024t_2 + 0.0421t_3 - 0.0073t_1t_2 \\ \quad +0.024t_2t_3 - 0.0118t_2t_4 - 0.0204t_3t_4 - 0.008t_3t_5 \\ \quad -0.0241t_2^2 + 0.0109t_4^2. \end{cases} \tag{17}$$

MOVPS is used to solve the multi-objective optimization problem of the car crashworthiness design and the obtained results are presented in Fig. 10. It is noted that MOVPS offers a range of solutions that are very close to the true Pareto front. These results validate the MOVPS capacity to deliver precise and reliable outcomes.

The performances of the proposed MOVPS algorithm is compared to the existing results from the literature [2,5,45], see Table 2. It is noted that the MOVPS algorithm has outperformed the other literature methods in terms of IGD and HV metrics.

4.3. Many-objective machining problem

The machining problem [8] formulates machining recommendations under multiple criteria. This problem considered tests on aluminum cut with VC-3 carbide cutting tools, see Fig. 11,



(a) A frontal crash (b) Design variables

Fig. 9. Crashworthiness design

Table 2. Vehicle crashworthiness design results

Methods	IGD		HV	
	Mean	SD	Mean	SD
d-NSGA-II [5]	0.03528	0.00470	1.02700	0.002500
SAMO [2]	0.00580	0.00030	0.01040	0.000800
MOMVO [45]	0.01608	0.00474	0.01057	0.000015
MOVPS	0.00297	0.00019	1.04050	0.000011

as a basis to test the approach, which has significant automotive industry applications [8]. The problem has three decision variables and four objective functions. The speed (ϑ), feed (j), and cut depth (D_c) are attributes considered in the definition of three decision variables: $m_1 = \ln \vartheta$, $m_1 = \ln(1000j)$, and $m_1 = \ln(1000D_c)$. The four objectives are considered in (18): (i) minimizing the surface roughness $f_1(m)$, (ii) maximizing the surface integrity $f_2(m)$, (iii) maximizing the tool life $f_3(m)$, and (iv) maximizing the metal removal rate $f_4(m)$. The machining optimization problem can be formulated as [8]:

$$\text{Minimize} \begin{cases} f_1 = -7.49 + 0.44m_1 - 1.16m_2 + 0.61m_3, \\ f_2 = -4.31 + 0.92m_1 - 0.16m_2 + 0.43m_3, \\ f_3 = 21.90 - 1.94m_1 - 0.3m_2 - 1.04m_3, \\ f_4 = -11.331 + m_1 + m_2 + m_3. \end{cases} \tag{18}$$

$$\text{Subject to} \begin{cases} G_1 = -0.44m_1 + 1.16m_2 - 0.61m_3 \leq -3.17, \\ G_2 = -0.92m_1 + 0.16m_2 - 0.43m_3 \leq -8.04, \\ G_3 = 1.94m_1 + 0.3m_2 + 1.04m_3 \leq 18.5, \\ 6.4 \leq m_1 \leq 7.09,\ 0.69 \leq m_2 \leq 2.89,\ 3.91 \leq m_3 \leq 4.61. \end{cases} \tag{19}$$

Fig. 12 presents the parallel coordinate plot solutions corresponding to the many-objective machining problem and obtained by using the MOVPS algorithm. In Table 3, we provide a

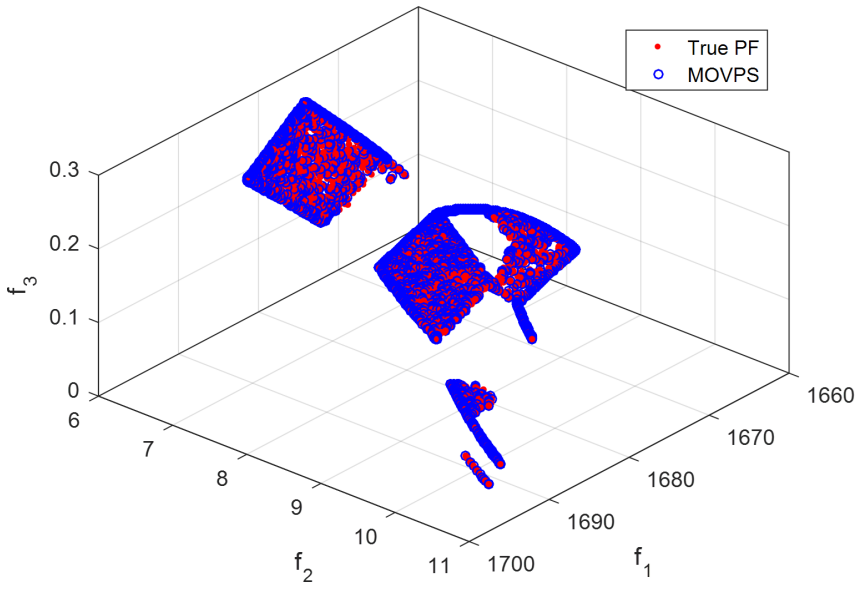


Fig. 10. Obtained results for vehicle crashworthiness problem

Fig. 11. Machining operation

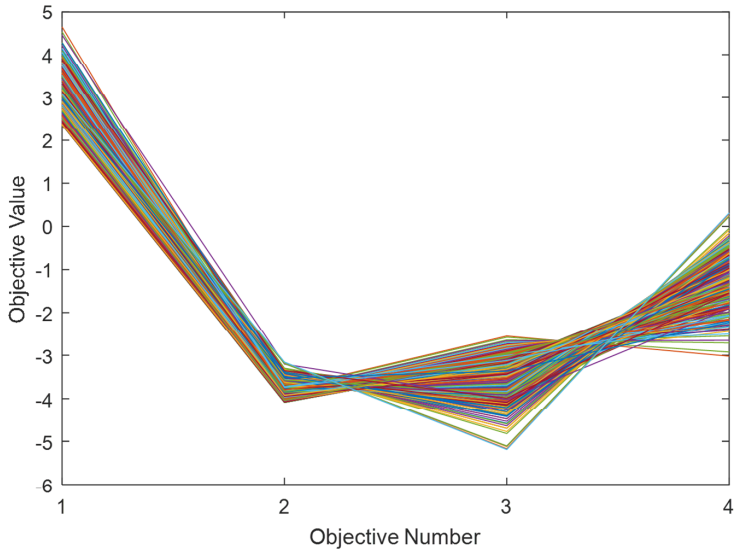


Fig. 12. Parallel coordinate plot solutions corresponding to machining problem

comparison of the MOVPS results with those reported in the existing literature. According to Table 3, MOVPS surpasses the other algorithms in both measures (IGD and HV), demonstrating its ability to achieve high convergence and diversity for the complex many-objective machining problem.

Table 3. Results of the machining problem

Methods	IGD		HV	
	Mean	SD	Mean	SD
MOABHH [8]	5.0530e-04	–	2.7118e-01	–
Kemeny-Young [7]	–	–	2.8194e-01	–
IBEA [8]	5.1369e-04	–	2.7348e-01	–
MOVPS	3.0521e-04	0.00009	5.7946e-01	0.00081

5. High dimensional problems

To evaluate the performance of MOVPS for high-dimensional problems, a set of benchmark functions is employed. These test functions can be divided into unimodal, multimodal, fixed dimension multimodal, and composite benchmark functions [1, 17, 22, 54]. The unimodal functions are appropriate for benchmarking exploitation. The multimodal functions, in contrast to the unimodal ones, have a large number of local optima, which increases exponentially with dimension. As a result, they are ideal for evaluating an algorithm exploration ability. The composite functions can be used to benchmark both exploration and exploitation at the same time [22,54]. The graphs illustrating the surface plots of these test functions are given in Fig. 13 as well as in Figs. 21 and 22, see Appendix.

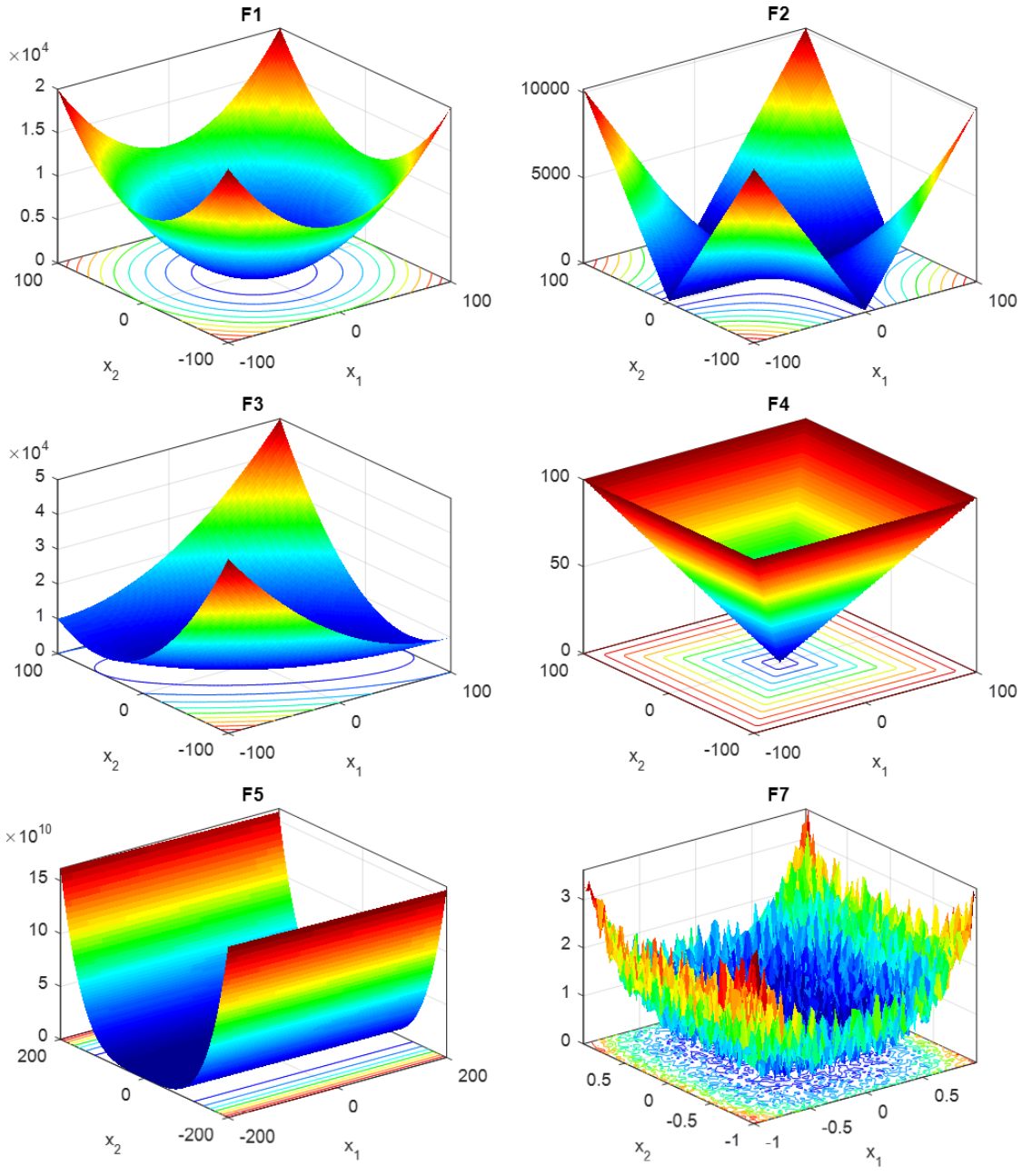


Fig. 13. Unimodal issues

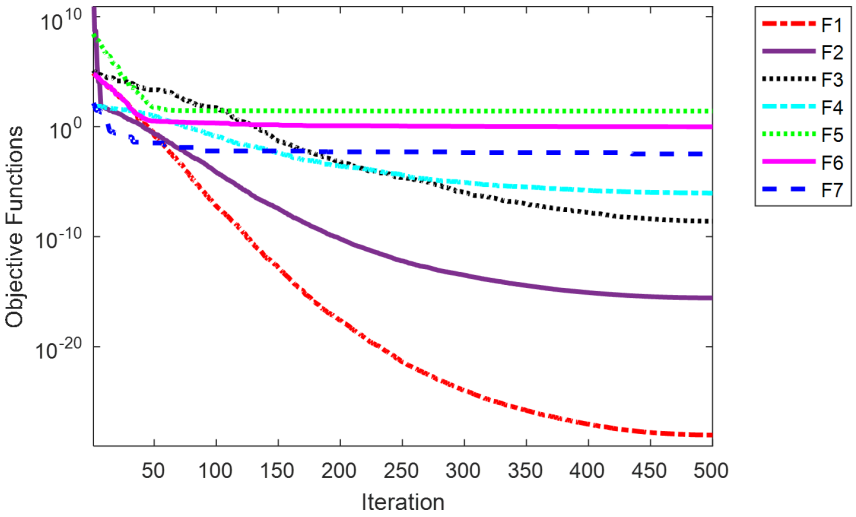


Fig. 14. Unimodal issues findings

Ten runs were performed for each benchmark function. The mean and the standard deviation of the obtained results were then recorded. During each iteration, the objective function was plotted. The results are displayed in logarithmic and linear scales in Figs. 14–17. The MOVPS results are compared, in Tables 9–10 (see Appendix), to the literature findings. Table 9 (Unimodal) shows that MOVPS presents a very competitive performance. In fact, MOVPS outperforms the literature methods in all unimodal test functions. These results demonstrate the MOVPS ability to reach the global optimum.

Furthermore, Table 9 (Multimodal) and Table 10 (Fixed dimension multimodal) show that MOVPS produces extremely competitive results for multimodal test functions. In fact, for the majority of multimodal functions, MOVPS dominates existing published methods. These findings prove that the MOVPS method outperforms the other algorithms in terms of exploration. MOVPS also presents very competitive performance for all composite benchmark functions, as shown in Table 10 (Composite).

Thus, it proves that MOVPS is able to balance successfully the exploration and exploitation

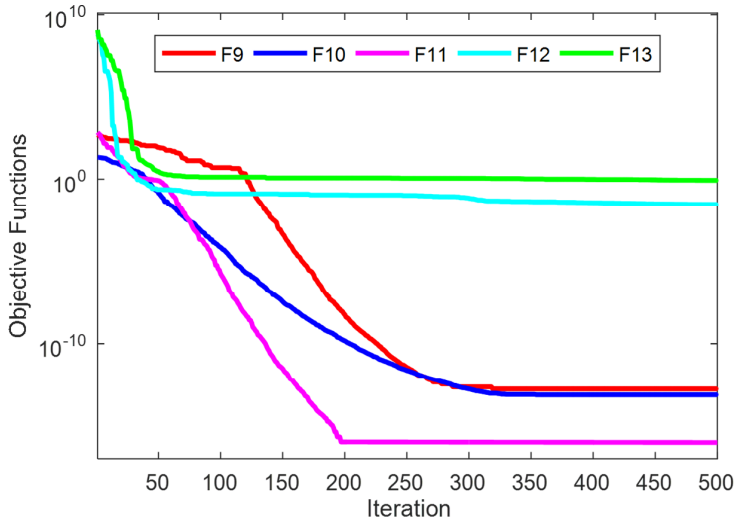


Fig. 15. Multimodal issues findings

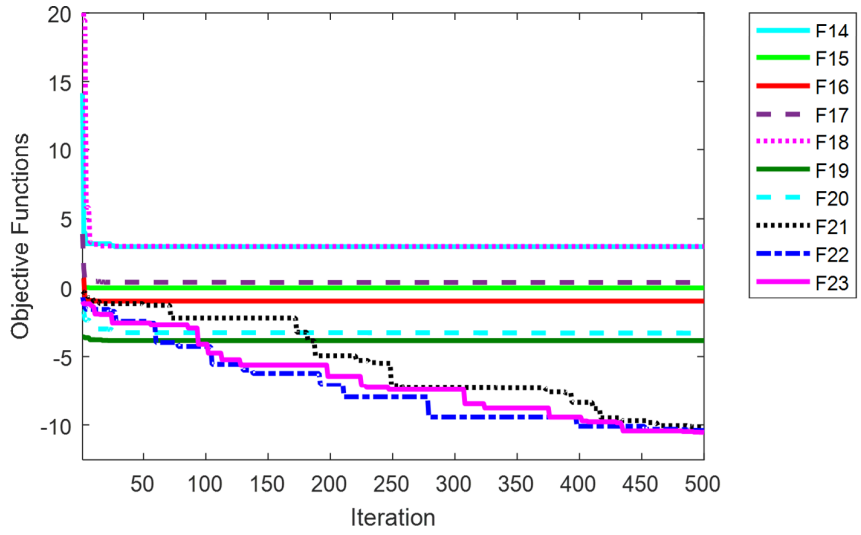


Fig. 16. Convergence curve for fixed multimodal test functions

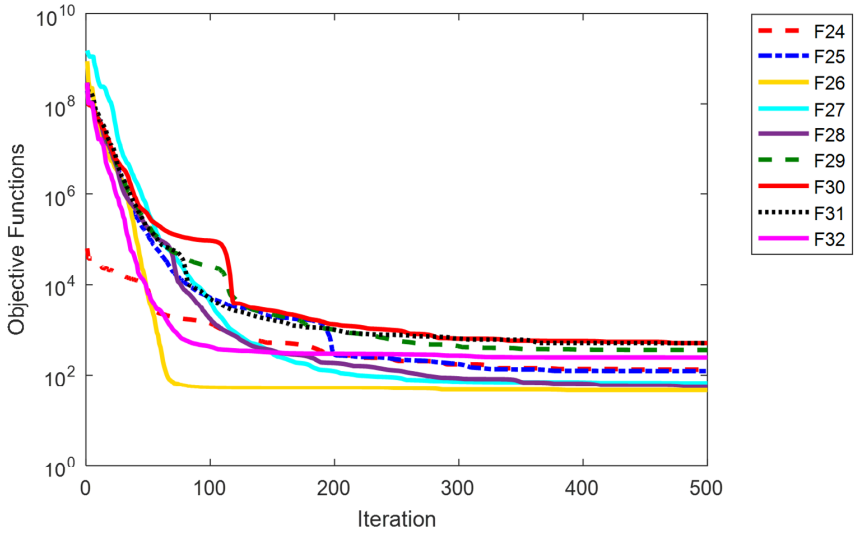


Fig. 17. Convergence curve for composite test functions

capabilities. This is due to the particle position enhancement mechanism implemented in the algorithm and used during the search for the optimal solution.

6. Conclusions

In this study, the Vibrating Particles System (VPS) algorithm was improved to address multi- and many-objective optimization problems with high convergence and diversity performances. The proposed method is called the Multi-Objective Vibrating Particles System (MOVPS) algorithm. For this purpose:

- The original VPS was modified by implementing the Pareto concept in order to deal with the many conflicts objective functions.
- A particle position enhancement mechanism was introduced in the algorithm to enhance the quality of obtained solutions and ensure the best compromise between exploitation and exploration skills.

- Several test functions were investigated to validate the proposed MOVPS.
- The efficiency of the proposed method was tested on multi- and many-objective engineering issues as well as unimodal, multimodal and composite test functions.
- An experimental study was conducted to compare the performance of MOVPS to other methods suggested in the literature.
- The performance of the proposed algorithm was assessed using inverted generational distance, hypervolume, robustness, spacing and spread measures, which are comprehensive indices of convergence and solution diversity.
- It was demonstrated that MOVPS produced superior outcomes than those found in the literature, even for high dimensional problems.
- In order to assess the MOVPS robustness, we looked at the variance of findings as a function of the number of runs in this study.
- The influence of the setup parameters on the algorithm performance will be examined in a future study.

References

[1] Ahmed, A. M., Rashid, T. A., Saeed, S. A. M., Cat swarm optimization algorithm: A survey and performance evaluation, Computational Intelligence and Neuroscience 2020 (2020) No. 4854895. https://doi.org/10.1155/2020/4854895

[2] Bhattacharjee, K. S., Singh, H. K., Ray, T., Multi-objective optimization with multiple spatially distributed surrogates, Journal of Mechanical Design 138 (9) (2016) No. 091401. https://doi.org/10.1115/1.4034035

[3] Bilel, N., Mohamed, N., Zouhaier, A., Lotfi, R., An efficient evolutionary algorithm for engineering design problems, Soft Computing 23 (2019) 6 197–6 213. https://doi.org/10.1007/s00500-018-3273-z

[4] Bilel, N., Mohamed, N., Zouhaier, A., Lotfi, R., An improved imperialist competitive algorithm for multi-objective optimization, Engineering Optimization 48 (11) (2016) 1 823–1 844. https://doi.org/10.1080/0305215X.2016.1141204

[5] Cai, X., Sun, H., Fan, Z., A diversity indicator based on reference vectors for many-objective optimization, Information Sciences 430–431 (2018) 467–486. https://doi.org/10.1016/j.ins.2017.11.051

[6] Cai, X., Sun, H., Zhu, C., Li, Z., Zhang, Q., Locating the boundaries of Pareto fronts: A many-objective evolutionary algorithm based on corner solution search, 2018, arXiv:1806.02967.

[7] Carvalho, V. R., Larson, K., Brandão, A. A. F., Sichman, J. S., Applying social choice theory to solve engineering multi-objective optimization problems, Journal of Control, Automation and Electrical Systems 31 (2020) 119–128. https://doi.org/10.1007/s40313-019-00526-2

[8] Carvalho, V. R., Sichman, J. S., Solving real-world multi-objective engineering optimization problems with an election-based hyper-heuristic, Proceedings of the International Workshop on Optimisation in Multiagent Systems (OptMAS-18), Stockholm, 2018, pp. 1–15.

[9] Chakri, A., Khelif, R., Benouaret, M., Yang, X.-S., New directional bat algorithm for continuous optimization problems, Expert Systems with Applications 69 (2017) 159–175. https://doi.org/10.1016/j.eswa.2016.10.050

[10] Chen, F., Wu, S., Liu, F., Ji, J., Lin, Q., A novel angular-guided particle swarm optimizer for many-objective optimization problems, Complexity 2020 (2020) No. 623820. https://doi.org/10.1155/2020/6238206

[11] Chen, J., Li, J. Xin, B., DMOEA-εC: Decomposition-based multiobjective evolutionary algorithm with the ε-constraint framework, IEEE Transactions on Evolutionary Computation 21 (5) (2017) 714–730. https://doi.org/10.1109/TEVC.2017.2671462

[12] Cheng, R., Li, M., Tian, Y., Zhang, X., Yang, S., Jin, Y., Yao, X., A benchmark test suite for evolutionary many-objective optimization, Complex & Intelligent Systems 3 (2017) 67–81. https://doi.org/10.1007/s40747-017-0039-7

[13] Cortés-Toro, E. M., Crawford, B., Gómez-Pulido, J. A., Soto, R., Lanza-Gutiérrez, J. M., A new metaheuristic inspired by the vapour-liquid equilibrium for continuous optimization, Applied Science 8 (11) (2018) No. 2080. https://doi.org/10.3390/app8112080

[14] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T., A fast and elitist multi-objective genetic algorithm: NSGA-II, IEEE Transactions on Evolutionary Computation 6 (2) (2002) 182–197. https://doi.org/10.1109/4235.996017

[15] Deb, K., Thiele, L., Laumanns, M., Zitzler, E., Scalable test problems for evolutionary multiobjective optimization, In: Evolutionary multi-objective optimization – Theoretical advances and applications, (editors) A. Abraham, L. Jain, R. Goldberg, Springer, Berlin, 2005, pp. 105–145. https://doi.org/10.1007/1-84628-137-7_6

[16] Dhiman, G., Kumar, V., Multi-objective spotted hyena optimizer: A multi-objective optimization algorithm for engineering problems, Knowledge-Based Systems (150) (2018) 175–197. https://doi.org/10.1016/j.knosys.2018.03.011

[17] Elaziz, M. A., Mirjalili, S., A hyper-heuristic for improving the initial population of whale optimization algorithm, Knowledge-Based Systems 172 (2019) 42–63. https://doi.org/10.1016/j.knosys.2019.02.010

[18] Gnetchejo, P. J., Essiane, S. N., Ele, P., Wamkeue, R., Wapet, D. M., Ngoffe, S. P., Enhanced vibrating particles system algorithm for parameters estimation of photovoltaic system, Journal of Power and Energy Engineering 7 (8) (2019) 1–26. https://doi.org/10.4236/jpee.2019.78001

[19] Gong, D., Sun, J., Miao, Z., A set-based genetic algorithm for interval many-objective optimization problems, IEEE Transactions on Evolutionary Computation 22 (1) (2016) 47–60. https://doi.org/10.1109/TEVC.2016.2634625

[20] Guedria, N. B., Improved accelerated PSO algorithm for mechanical engineering optimization problems, Applied Soft Computing 40 (2016) 455–467. https://doi.org/10.1016/j.asoc.2015.10.048

[21] Hassan, M. H., Daqaq, F., Selim, A., Domínguez-García, J. L., Kamel, S., MOIMPA: Multi-objective improved marine predators algorithm for solving multi-objective optimization problems, Soft Computing 27 (21) (2023) 15 719–15 740. https://doi.org/10.1007/s00500-023-08812-7

[22] Hedar, A. R., Deabes, W., Almaraashi, M., Amin. H. H., Evolutionary algorithms enhanced with quadratic coding and sensing search for global optimization, Mathematical and Computational Applications 25 (1) (2020) No. 7. https://doi.org/10.3390/mca25010007

[23] Hosseinian, A., Bardaran, V., Modified Pareto archived evolution strategy for the multi-skill project scheduling problem with generalized precedence relations, Journal of Industrial Engineering and Management Studies 7 (1) (2020) 59–86. https://doi.org/10.22116/jiems.2020.110007

[24] Hu, P., Chen, S., Huang, H., Zhang, G., Liu L., Improved alpha-guided grey wolf optimizer, IEEE Access 7 (2019) 5 421–5 437. https://doi.org/10.1109/ACCESS.2018.2889816

[25] Huband, S., Hingston, P., Barone, L., While, L., A review of multiobjective test problems and a scalable test problem toolkit, IEEE Transactions on Evolutionary Computation 10 (5) (2006) 477–506. http://dx.doi.org/10.1109/TEVC.2005.861417

[26] Huo, J., Liu, L., An optimization framework of multiobjective artificial bee colony algorithm based on the MOEA framework, Computational Intelligence and Neuroscience 2018 (2018) No. 5865168. https://doi.org/10.1155/2018/5865168

[27] Ishibuchi, H., Imada, R., Setoguchi, Y., Nojima, Y., How to specify a reference point in hypervolume calculation for fair performance comparison, Evolutionary Computation 26 (3) (2018) 411–440. https://doi.org/10.1162/evco_a_00226

[28] Karakoyun, M., Ozkis, A., Kodaz, H., A new algorithm based on grey wolf optimizer and shuffled frog leaping algorithm to solve the multi-objective optimization problems, Applied Soft Computing 96 (2020) No. 106560. https://doi.org/10.1016/j.asoc.2020.106560

[29] Kaur, S., Awasthi, L. K. Sangal, A. L., HMOSHSSA: A hybrid meta-heuristic approach for solving constrained optimization problems, Engineering with Computers 37 (2021) 3 167–3 203. https://doi.org/10.1007/s00366-020-00989-x

[30] Kaur, A., Kumar, Y., A multi-objective vibrating particle system algorithm for data clustering, Pattern Analysis and Applications 25 (2022) 209–239. https://doi.org/10.1007/s10044-021-01052-1

[31] Kaveh, A. Dadras, A., A novel meta-heuristic optimization algorithm: Thermal exchange optimization, Advances in Engineering Software 110 (2017) 69–84. https://doi.org/10.1016/j.advengsoft.2017.03.014

[32] Kaveh, A., Dadras Eslamlou, A., Water strider algorithm: A new metaheuristic and applications, Structures 25 (2020) 520–541. https://doi.org/10.1016/j.istruc.2020.03.033

[33] Kaveh, A., Hoseini Vaez, S. R., Hosseini, P., Enhanced vibrating particles system algorithm for damage identification of truss structures, Scientia Iranica 26 (1) (2019) 246–256. https://doi.org/10.24200/sci.2017.4265

[34] Kaveh, A., Hoseini Vaez, S. R., Hosseini, P., Bakhtyari, M., Optimal design of steel curved roof frames by enhanced vibrating particles system algorithm, Periodica Polytechnica Civil Engineering 63 (4) (2019) 947–960. https://doi.org/10.3311/PPci.14812

[35] Kaveh, A., Ilchi Ghazaan, M., A new hybrid meta-heuristic algorithm for optimal design of large-scale dome structures, Engineering Optimization 50 (2) (2018) 235–252. https://doi.org/10.1080/0305215X.2017.1313250

[36] Kaveh, A., Ilchi Ghazaan, M., A new meta-heuristic algorithm: Vibrating particles system, Scientia Iranica 24 (2) (2017) 551–566. https://doi.org/10.24200/SCI.2017.2417

[37] Kaveh, A., Ilchi Ghazaan, M., A new VPS-based algorithm for multi-objective optimization problems, Engineering with Computers 36 (2020) 1 029–1 040. https://doi.org/10.1007/s00366-019-00747-8

[38] Kaveh, A., Vazirinia, Y., An upgraded sine cosine algorithm for tower crane selection and layout problem, Periodica Polytechnica Civil Engineering 64 (2) (2020) 325–343. https://doi.org/10.3311/PPci.15363

[39] Kong, L., Wang, L., Li, F., Lv, X., Li, J., Ma, Y., Chen, B., Guo, J., Multilayer integration framework for low carbon design based on design features, Journal of Manufacturing Systems 61 (2021) 223–238. https://doi.org/10.1016/j.jmsy.2021.09.008

[40] Li, Y., Kou, Y., Li, Z., An improved non-dominated sorting genetic algorithm III method for solving multiobjective weapon-target assignment – Part I: The value of fighter combat, International Journal of Aerospace Engineering 2018 (2018) No. 8302324. https://doi.org/10.1155/2018/8302324

[41] Lin, Q., Liu, S., Wong, K.-C., Gong, M., Coello Coello, C. A., Chen, J., Zhang, J., A clustering-based evolutionary algorithm for many-objective optimization problems, IEEE Transactions on Evolutionary Computation 23 (3) (2019) 391–405. https://doi.org/10.1109/TEVC.2018.2866927

[42] Lu, C., Gao, L., Li, X., Zeng, B., Zhou, F., A hybrid multi-objective evolutionary algorithm with feedback Mechanism, Applied Intelligence 48 (2018) 4 149–4 173. https://doi.org/10.1007/s10489-018-1211-5

[43] Luo, J., Gupta, A., Ong, Y.-S., Wang, Z., Evolutionary optimization of expensive multiobjective problems with co-sub-Pareto front Gaussian process surrogates, IEEE Transactions on Cybernetics 49 (5) (2019) 1 708–1 721. https://doi.org/10.1109/TCYB.2018.2811761

[44] Luo, J., Huang, X., Yang, Y., Li, X., Wang, Z., Feng, J., A many-objective particle swarm optimizer based on indicator and direction vectors for many-objective optimization, Information Sciences 514 (2020) 166–202. https://doi.org/10.1016/j.ins.2019.11.047

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

[46] Nejlaoui, M., Multi-objective enhanced imperialistic competitive method for multi-criteria engineering issues, International Journal of Computer Applications in Technology 69 (4) (2022) 344–356. https://doi.org/10.1504/IJCAT.2022.129384

[47] Patil, M. V., Kulkarni, A. J., Pareto dominance based multiobjective cohort intelligence algorithm, Information Sciences 538 (2020) 69–118. https://doi.org/10.1016/j.ins.2020.05.019

[48] Peng, X., Xia, X., Liao, W., Guo, Z., Running time analysis of the Pareto archived evolution strategy on pseudo-Boolean functions, Multimedia Tools and Applications 77 (2018) 11 203–11 217. https://doi.org/10.1007/s11042-017-5466-3

[49] Rabiei, M. H., Aalami, M. T., Talatahari, S., Reservoir operation optimization using CBO, ECBO and VPS algorithms, International Journal of Optimization in Civil Engineering 8 (3) (2018) 489–509.

[50] Rezazadeh, F., Talatahari, S., Seismic energy-based design of BRB frames using multi-objective vibrating particles system optimization, Structures 24 (2020) 227–239. https://doi.org/10.1016/j.istruc.2020.01.006

[51] Salimi, P., Rahimi Boderabady, H., Kaveh, A., Optimal design of reinforced concrete frame structures using cascade optimization method, Periodica Polytechnica Civil Engineering 66 (4) (2022) 1 220–1 233. https://doi.org/10.3311/PPci.20868

[52] Vu, V. T., Bui, L. T., Nguyen, T. T., A modified dual-population approach for solving multi-objective problems, Proceedings of the 21st Asia Pacific Symposium on Intelligent and Evolutionary Systems (IES), Hanoi, 2017, pp. 89–94. https://doi.org/10.1109/IESYS.2017.8233567

[53] Yang, T., Zhang, S., Li, C., A multi-objective hyper-heuristic algorithm based on adaptive epsilon-greedy selection, Complex & Intelligent Systems 7 (2021) 765–780. https://doi.org/10.1007/s40747-020-00230-8

[54] Yildizdan, G., Baykan, Ö. K., A new hybrid BA_ABC algorithm for global optimization problems, Mathematics 8 (10) (2020) No. 1749. https://doi.org/10.3390/math8101749

[55] Zhang, Q., Zhou, A., Zhao, S., Suganthan, P. N., Liu, W., Tiwari, S., Multiobjective optimization test instances for the CEC 2009 special session and competition, Technical Report CES-487, University of Essex, 2009.

[56] Zhang, X., Liu, H., Tu, L., Zhao, J., An efficient multi-objective optimization algorithm based on level swarm optimizer, Mathematics and Computers in Simulation 177 (2020) 588–602. https://doi.org/10.1016/j.matcom.2020.05.025

[57] Zhao, M., Ge, H., Zhang, K., Hou, Y., A reference vector based many-objective evolutionary algorithm with feasibility-aware adaptation, 2019, arXiv 1904.06302.

[58] Zhao, Y., Sun, C., Zeng, J., Tan, Y., Zhang, G., A surrogate-ensemble assisted expensive many-objective optimization, Knowledge-Based Systems 211 (2021) No. 106520. https://doi.org/10.1016/j.knosys.2020.106520

[59] Zhou, A., Jin, Y., Zhang, Q., Sendhoff, B., Tsang, E., Combining model-based and genetics-based offspring generation for multi-objective optimization using a convergence criterion, Proceedings of the 2006 IEEE Congress on Evolutionary Computation, Vancouver, 2006, pp. 892–899.

[60] Zhou, J., Yao, X., Gao, L., Hu, C., An indicator and adaptive region division based evolutionary algorithm for many-objective optimization, Applied Soft Computing 99 (2021) No. 106872. https://doi.org/10.1016/j.asoc.2020.106872

[61] Zhu, S., Xu, L., Goodman, E., Deb, K., Lu, Z., The (M-1)+1 Framework of relaxed Pareto dominance for evolutionary many-objective optimization, In: Evolutionary Multi-Criterion Optimization (EMO 2021), (editors) H. Ishibuchi, Q. Zhang, R. Cheng, K. Li, H. Li, H. Wang, A. Zhou, Springer, Shenzhen, 2021, pp. 349–361. https://doi.org/10.1007/978-3-030-72062-9_28

Appendix

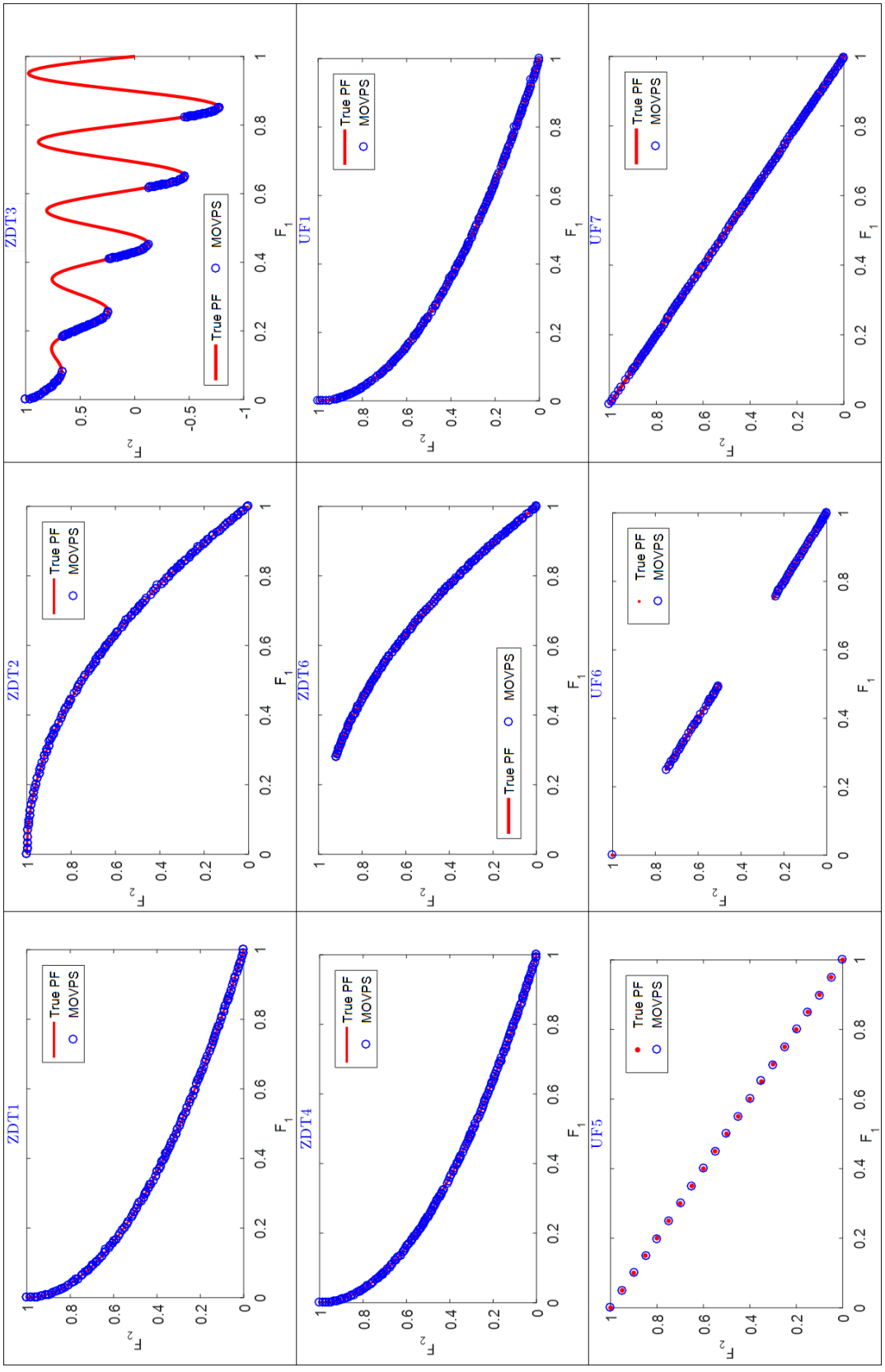


Fig. 18. The true and Pareto fronts of bi-objective test functions

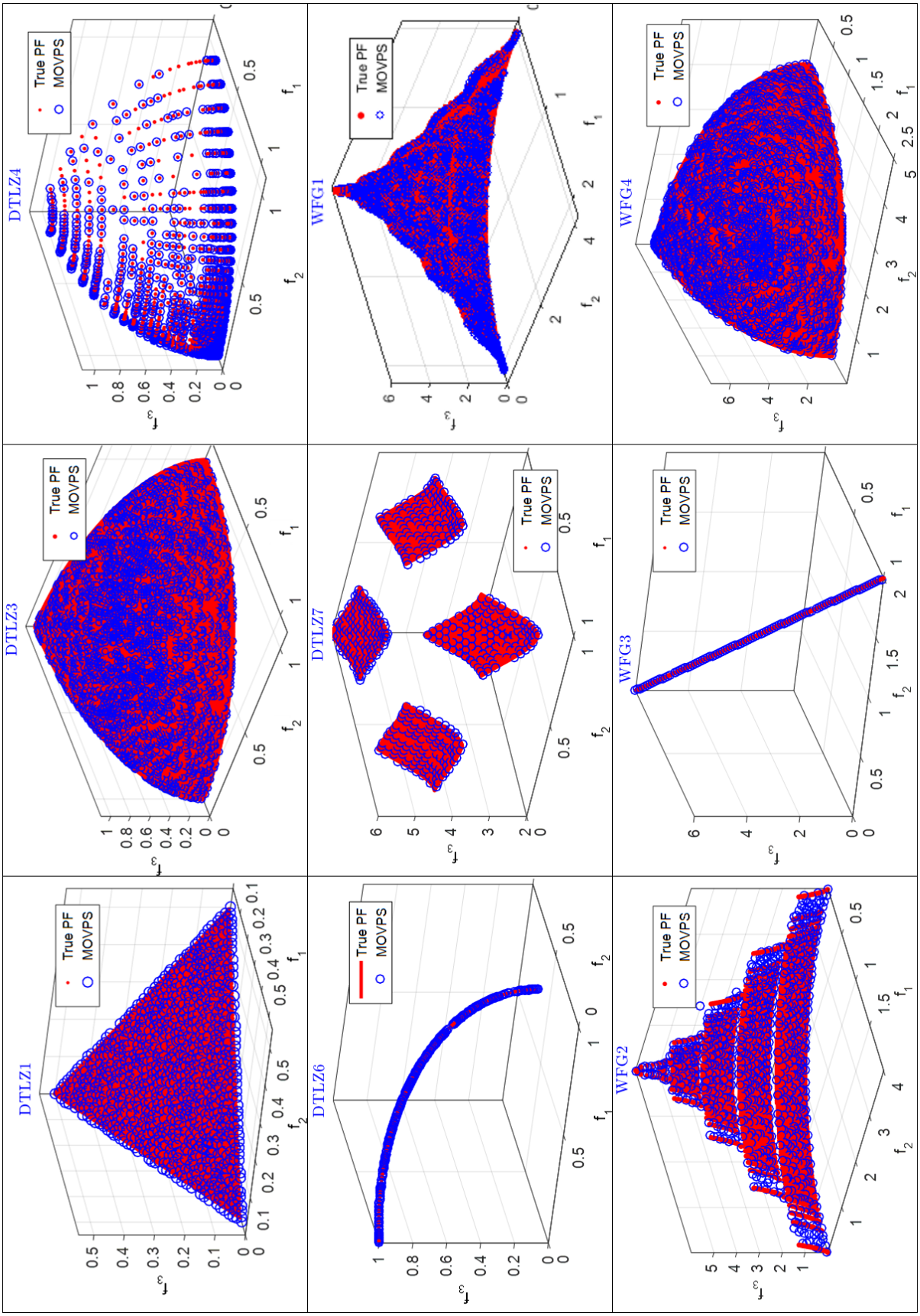


Fig. 19. The true and Pareto fronts of tri-objective test functions

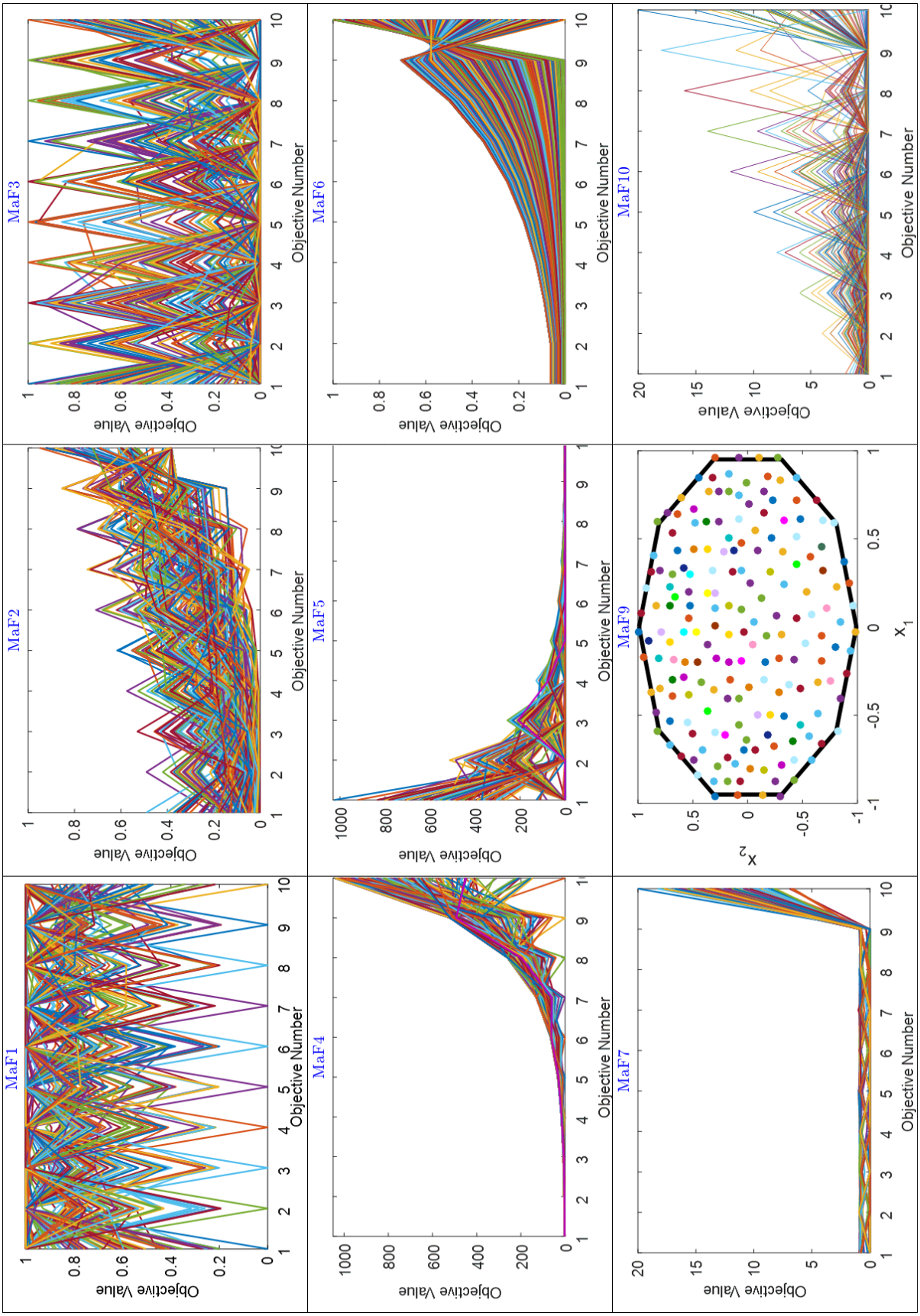


Fig. 20. Parallel coordinate plot of many-objective test functions

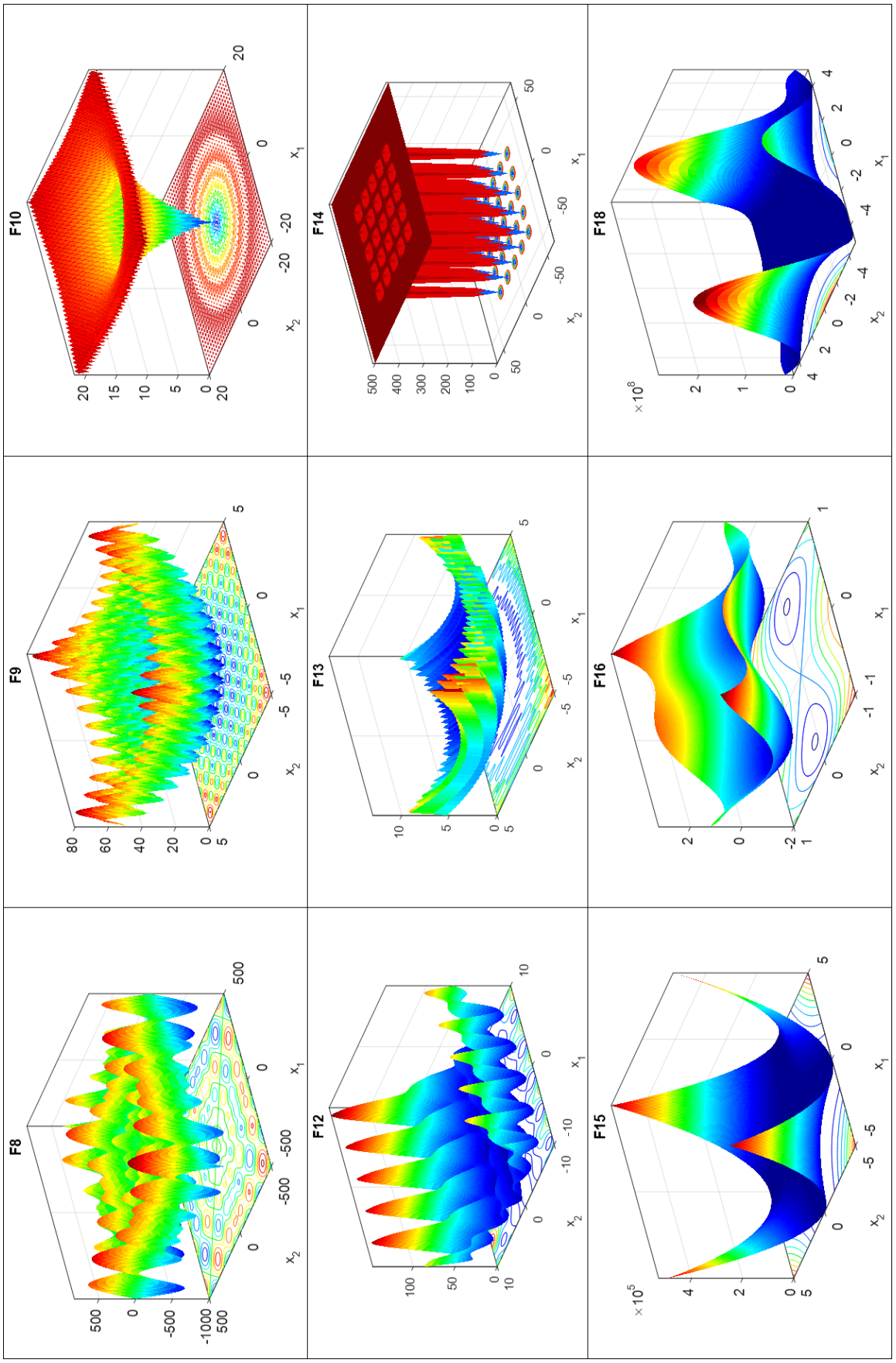


Fig. 21. Multimodal issues

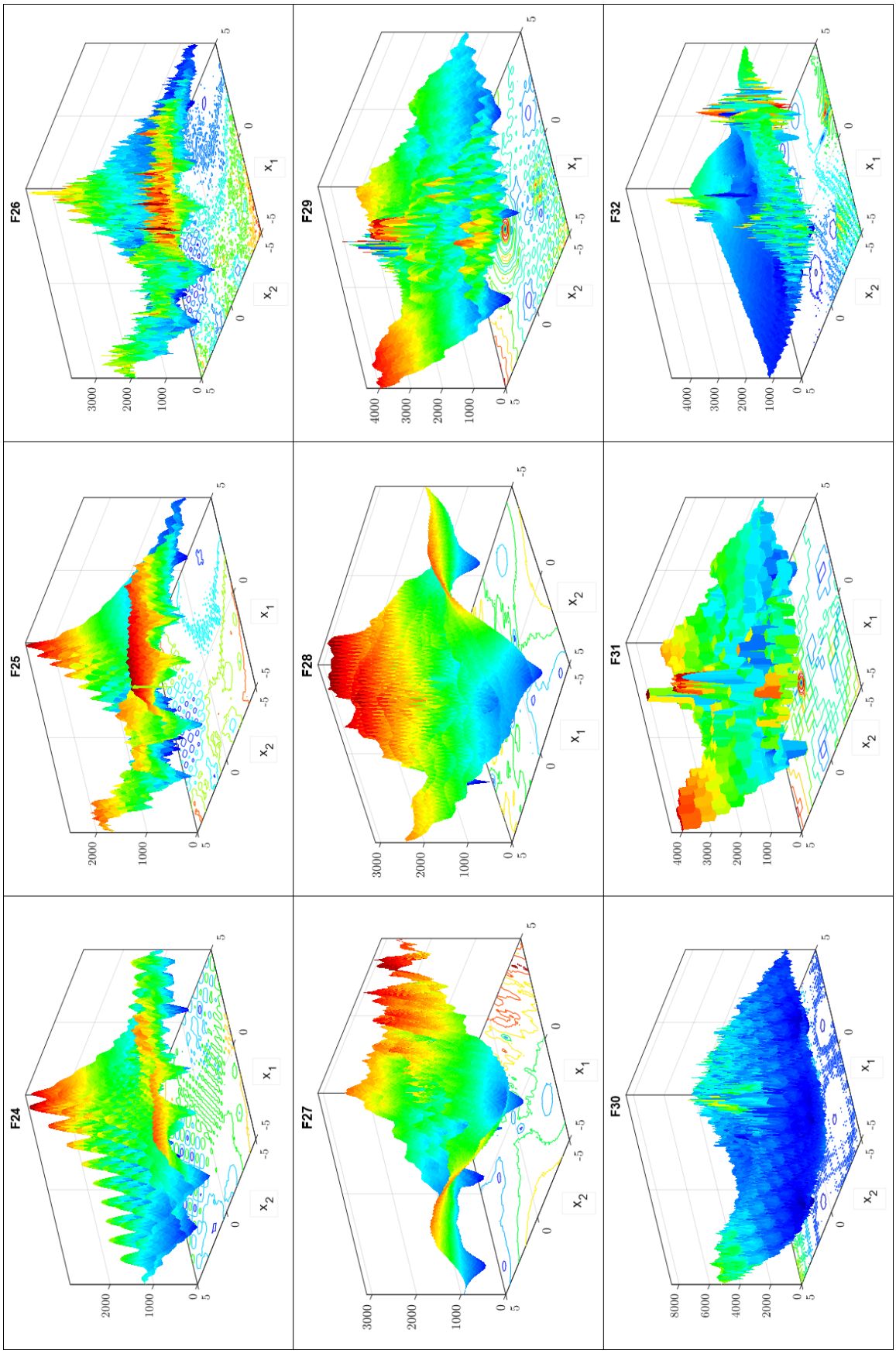


Fig. 22. Composite test functions

Table 4. IGD and HV values for bi-objective test functions

	Function	MOCI [47]	DMOEA-εC [11]	DPP2 [52]	MOSG [28]	eMOABC [6]	Proposed MOVPS
IGD values	ZDT1	1.12e-03 ± 2.21e-04	3.763e-03 ± 6.690e-05	5.553788e-05 ± 1.7e-07	1.34e-04 ± 8.2e-07	3.737e-03 ± 2.8e-05	4.931e-05 ± 9.701e-08
	ZDT2	8.25e-04 ± 9.50e-05	3.800e-03 ± 7.120e-05	4.660187e-05 ± 4.9e-08	1.41e-04 ± 1.7e-06	3.850e-03 ± 3.2e-05	3.538e-05 ± 3.234e-08
	ZDT3	1.20e-03 ± 1.11e-04	5.141e-03 ± 1.696e-04	8.861649e-05 ± 5.4e-07	1.00e-04 ± 1.2e-06	5.164e-03 ± 3.3e-04	8.104e-05 ± 4.962e-07
	ZDT4	8.09e-04 ± 6.36e-05	4.060e-03 ± 2.799e-05	5.900098e-05 ± 3.3e-07	1.36e-04 ± 1.3e-06	3.710e-03 ± 1.9e-05	3.279e-05 ± 2.819e-07
	ZDT6	9.55e-04 ± 1.10e-04	3.000e-03 ± 8.050e-05	4.627856e-05 ± 3.9e-09	1.34e-04 ± 8.5e-07	3.025e-03 ± 1.4e-05	3.511e-05 ± 1.775e-09
	UF1	1.22e-03 ± 1.14e-04	4.407e-03 ± 9.870e-05	7.137613e-05 ± 7.6e-06	5.69e-03 ± 5.5e-04	1.480e-02 ± 1.9e-03	5.907e-05 ± 9.709e-07
	UF2	2.03e-03 ± 3.67e-04	6.503e-03 ± 5.229e-04	4.182316e-04 ± 1.6e-04	3.49e-03 ± 2.8e-04	1.407e-02 ± 3.4e-03	2.407e-04 ± 8.826e-05
	UF3	1.73e-03 ± 3.38e-05	7.209e-03 ± 1.355e-02	6.285715e-04 ± 8.8e-04	1.76e-02 ± 5.6e-03	1.075e-01 ± 1.6e-02	9.939e-05 ± 1.105e-04
	UF4	1.49e-03 ± 1.30e-04	6.017e-02 ± 5.017e-03	1.955271e-03 ± 1.4e-04	–	4.123e-02 ± 9.8e-04	1.188e-03 ± 1.092e-04
	UF5	–	3.845e-01 ± 1.251e-01	6.882732e-02 ± 1.5e-02	–	1.625e-01 ± 2.2e-02	5.976e-02 ± 9.859e-03
	UF6	–	2.662e-01 ± 2.012e-01	6.071352e-03 ± 6.1e-03	–	2.174e-01 ± 3.7e-02	6.132e-03 ± 5.677e-03
	UF7	1.55e-03 ± 3.75e-04	4.079e-03 ± 1.269e-03	2.521310e-04 ± 3.7e-04	–	1.872e-02 ± 5.4e-03	1.469e-04 ± 2.998e-04
HV values	ZDT1	7.23e-01 ± 4.43e-04	8.727e-01 ± 1.541e-02	6.648521e-01 ± 4.4e-05	6.62e-01 ± 3.1e-05	6.620e-01 ± 3.6e-05	7.639e-01 ± 2.939e-05
	ZDT2	4.48e-01 ± 1.75e-04	5.434e-01 ± 1.405e-02	3.315724e-01 ± 4.0e-05	3.29e-01 ± 2.2e-05	3.287e-01 ± 3.2e-05	6.281e-01 ± 2.010e-05
	ZDT3	5.85e-01 ± 2.08e-04	1.003e-01 ± 1.304e-02	5.162230e-01 ± 9.9e-06	5.16e-01 ± 3.6e-05	5.155e-01 ± 1.4e-04	6.105e-01 ± 8.719e-06
	ZDT4	7.24e-01 ± 8.50e-05	8.668e-01 ± 4.691e-02	6.649724e-01 ± 8.8e-06	6.62e-01 ± 2.1e-05	6.620e-01 ± 1.7e-05	8.792e-01 ± 7.993e-06
	ZDT6	3.91e-01 ± 2.08e-04	5.086e-01 ± 1.469e-02	4.047281e-01 ± 2.2e-07	4.01e-01 ± 1.2e-05	4.014e-01 ± 1.7e-05	5.414e-01 ± 2.015e-07
	UF1	7.23e-01 ± 1.39e-04	8.741e-01 ± 1.434e-02	6.635530e-01 ± 2.4e-04	5.19e-01 ± 1.4e-02	6.430e-01 ± 3.3e-03	7.839e-01 ± 1.395e-04
	UF2	7.22e-01 ± 5.78e-04	8.700e-01 ± 1.382e-02	6.570243e-01 ± 3.2e-03	5.83e-01 ± 4.2e-03	6.497e-01 ± 2.3e-03	8.947e-01 ± 4.978e-04
	UF3	7.23e-01 ± 9.22e-05	8.687e-01 ± 1.633e-02	6.490813e-01 ± 1.6e-02	1.81e-01 ± 1.0e-01	4.981e-01 ± 2.3e-02	8.981e-01 ± 8.554e-05
	UF4	4.47e-01 ± 5.26e-04	4.436e-01 ± 1.965e-02	2.450716e-01 ± 5.5e-03	–	2.683e-01 ± 1.4e-03	4.698e-01 ± 4.747e-04
	UF5	–	1.256e-01 ± 4.731e-02	8.646766e-02 ± 8.3e-02	–	2.579e-01 ± 2.6e-02	2.315e-01 ± 2.496e-02
	UF6	–	3.1704e-01 ± 1.061e-01	2.045398e-01 ± 1.0e-01	–	2.241e-01 ± 2.9e-02	4.001e-01 ± 2.112e-02
	UF7	5.84e-01 ± 1.54e-03	7.054e-01 ± 1.701e-02	4.949293e-01 ± 2.6e-03	–	4.704e-01 ± 8.1e-03	7.246e-01 ± 1.309e-03

Table 5. SPA and SP values for bi-objective test functions

	Function	MOIMPA [21]	MOVPS [37]	MOALO [21]	MOMVO [21]	Proposed MOVPS
SPA values	ZDT1	6.59e-03 ± 1.92e-03	0.00270 ± 0.00014	6.79e-03 ± 2.85e-03	1.11e-02 ± 1.95e-03	7.171e-04 ± 8.843e-05
	ZDT2	1.17e-02 ± 2.25e-02	–	2.74e-03 ± 3.18e-03	2.08e-02 ± 8.59e-03	9.538e-04 ± 6.296e-04
	ZDT3	8.12e-03 ± 2.15e-03	0.00595 ± 0.00693	1.40e-02 ± 9.28e-03	8.25e-02 ± 1.59e-01	2.154e-03 ± 1.962e-03
	ZDT4	8.56e-03 ± 6.06e-03	0.01458 ± 0.00904	4.44e-02 ± 6.00e-02	7.00e-01 ± 2.11e+00	3.279e-03 ± 2.848e-03
	ZDT6	3.12e-02 ± 8.01e-02	0.00355 ± 0.00146	5.71e-02 ± 2.97e-02	3.38e-02 ± 2.87e-02	3.101e-03 ± 9.755e-04
	UF1	5.45e-03 ± 1.35e-03	–	8.82e-03 ± 1.24e-02	4.27e-03 ± 1.61e-03	2.907e-03 ± 9.384e-04
	UF2	7.26e-03 ± 2.11e-03	–	1.85e-02 ± 6.76e-03	1.62e-02 ± 1.07e-02	2.411e-03 ± 8.826e-04
	UF3	5.18e-03 ± 2.39e-03	–	1.68e-02 ± 1.46e-02	3.45e-02 ± 4.33e-02	3.243e-03 ± 1.195e-03
	UF4	4.97e-03 ± 6.93e-04	–	1.87e-02 ± 4.23e-03	6.65e-03 ± 3.23e-03	5.018e-03 ± 6.192e-04
	UF5	1.60e-02 ± 6.35e-03	–	4.09e-02 ± 4.62e-02	1.82e-02 ± 2.20e-02	8.976e-03 ± 3.859e-03
	UF6	1.00e-02 ± 4.17e-03	–	2.30e-02 ± 1.13e-02	2.72e-02 ± 2.16e-02	8.132e-03 ± 2.677e-03
	UF7	5.26e-03 ± 1.03e-03	–	9.19e-03 ± 3.21e-03	9.57e-03 ± 5.05e-03	1.946e-03 ± 9.208e-04

	Function	MOHGD [42]	MOABCrandElite [6]	MOVPS [37]	eMOABC [6]	Proposed MOVPS
SP values	ZDT1	1.10e-01 ± 8.7e-03	1.255e-01 ± 1.1e-02	0.99632 ± 0.00050	1.164e-01 ± 1.0e-02	8.914e-02 ± 2.710e-03
	ZDT2	1.24e-01 ± 1.3e-02	1.171e-01 ± 1.1e-02	–	1.124e-01 ± 1.1e-02	8.538e-02 ± 7.254e-03
	ZDT3	7.11e-01 ± 5.2e-03	8.547e-01 ± 3.6e-02	0.93264 ± 0.01469	7.997e-01 ± 2.7e-02	3.104e-01 ± 4.962e-03
	ZDT4	4.86e-01 ± 2.2e-01	9.867e-02 ± 1.1e-02	0.99103 ± 0.00577	9.142e-02 ± 1.3e-02	5.257e-02 ± 2.819e-03
	ZDT6	1.65e-01 ± 9.3e-03	1.263e-01 ± 7.2e-02	0.99598 ± 0.02799	1.309e-01 ± 2.0e-01	7.011e-02 ± 6.795e-03
	UF1	1.34 ± 1.2e-01	8.216e-01 ± 1.1e-01	–	7.576e-01 ± 9.0e-02	5.209e-01 ± 3.539e-02
	UF2	5.64e-01 ± 5.4e-02	7.667e-01 ± 4.9e-02	–	7.886e-01 ± 5.3e-02	1.147e-01 ± 4.006e-02
	UF3	1.07 ± 2.0e-01	7.216e-01 ± 7.9e-02	–	7.714e-01 ± 1.2e-01	2.313e-01 ± 5.104e-02
	UF4	3.87e-01 ± 4.2e-02	6.090e-01 ± 5.7e-02	–	5.923e-01 ± 5.1e-02	9.650e-02 ± 1.125e-02
	UF5	1.12 ± 2.8e-01	5.392e-01 ± 1.4e-01	–	6.204e-01 ± 1.1e-01	1.061e-01 ± 9.851e-02
	UF6	9.44e-01 ± 2.3e-01	7.076e-01 ± 1.8e-01	–	7.535e-01 ± 1.2e-01	9.132e-02 ± 5.767e-02
	UF7	1.10 ± 2.7e-01	9.238e-01 ± 8.1e-02	–	8.840e-01 ± 7.0e-02	1.649e-01 ± 2.989e-03

Table 6. IGD and HV values for tri-objective test functions

	Function	MOCI [47]	NSGAII [56]	DPP2 [52]	MOSG [28]	EMOSO [56]	Proposed MOVPS
IGD values	DTLZ1	9.46e-03 ± 4.04e-03	–	3.474254e-04 ± 1.1e-06	1.20e-02 ± 1.5e-02	–	2.953e-04 ± 8.949e-07
	DTLZ2	3.22e-02 ± 4.48e-03	6.9120e-02 ± 2.26e-03	4.306460e-04 ± 2.0e-06	6.96e-04 ± 2.3e-05	5.9421e-02 ± 1.22e-03	3.763e-04 ± 9.768e-07
	DTLZ3	3.01e-02 ± 1.29e-03	–	7.225804e-04 ± 4.3e-06	–	–	5.686e-04 ± 3.744e-06
	DTLZ4	4.65e-02 ± 9.50e-03	–	8.112918e-04 ± 1.9e-04	1.13e-03 ± 1.1e-04	–	6.907e-04 ± 9.108e-05
	DTLZ5	1.16e-03 ± 2.81e-04	6.1544e-03 ± 3.17e-04	1.519963e-05 ± 1.2e-07	2.74e-05 ± 6.1e-06	8.7620e-03 ± 9.41e-04	1.909e-05 ± 8. 969e-08
	DTLZ6	9.99e-04 ± 6.03e-05	2.3862e-02 ± 7.39e-02	3.453052e-05 ± 2.9e-08	3.39e-05 ± 1.1e-06	1.4081e-01 ± 3.31e-01	1.873e-05 ± 2.055e-08
	DTLZ7	2.49e-02 ± 6.02e-03	1.0182e-01 ± 5.08e-02	6.889806e-03 ± 7.7e-03	2.93e-03 ± 4.6e-03	7.0427e-02 ± 3.32e-03	2.518e-03 ± 1.896e-03
	WFG1	–	9.9846e-01 ± 9.21e-02	2.678682e-04 ± 1.5e-05	1.21e-02 ± 3.5e-04	1.9531e-00 ± 3.95e-02	9.475e-05 ± 1.682e-05
	WFG2	1.37e-01 ± 6.35e-03	2.4348e-01 ± 7.62e-02	6.028959e-04 ± 9.1e-06	3.45e-04 ± 1.2e-04	1.7762e-01 ± 5.08e-03	2.093e-04 ± 7.751e-06
	WFG3	–	1.2671e-01 ± 1.74e-02	5.478467e-05 ± 7.7e-08	1.36e-04 ± 7.4e-06	2.5041e-01 ± 2.29e-02	4.370e-05 ± 5.997e-08
	WFG4	8.02e-02 ± 6.94e-03	2.7724e-01 ± 6.31e-03	6.323277e-05 ± 3.9e-06	1.78e-04 ± 3.3e-05	2.7081e-01 ± 7.90e-03	5.003e-05 ± 9.977e-07
	WFG5	1.29e-01 ± 4.08e-03	2.8296e-01 ± 9.30e-03	9.326983e-04 ± 5.2e-07	1.05e-04 ± 1.7e-05	2.5086e-01 ± 8.49e-03	1.211e-04 ± 4.010e-07
	WFG6	1.14e-01 ± 1.26e-02	3.3193e-01 ± 1.60e-02	9.139314e-05 ± 9.8e-08	2.33e-04 ± 2.6e-05	2.4222e-01 ± 1.05e-02	7.144e-05 ± 2.504e-08
	WFG7	1.00e-01 ± 5.54e-03	2.8277e-01 ± 1.03e-02	4.054533e-05 ± 1.6e-08	1.62e-04 ± 1.9e-05	2.2987e-01 ± 3.38e-03	3.712e-05 ± 9.432e-09

	Function	MOCI [47]	NSGAII [56]	DPP2 [52]	MOSG [28]	EMOSO [56]	Proposed MOVPS
HV values	DTLZ1	8.08e-01 ± 2.03e-02	–	7.849109e-01 ± 3.6e-04	5.56e-02 ± 1.2e-01	–	8.121e-01 ± 1.339e-04
	DTLZ2	4.33e-01 ± 6.41e-03	5.2932e-01 ± 4.79e-03	4.18484e-01 ± 7.7e-04	3.82e-01 ± 3.8e-03	5.4001e-01 ± 3.32e-03	6.008e-01 ± 4.748e-04
	DTLZ3	4.35e-01 ± 3.64e-03	–	4.187471e-01 ± 1.0e-03	–	–	5.573e-01 ± 1.471e-03
	DTLZ4	4.29e-01 ± 2.64e-03	–	4.118296e-01 ± 2.2e-02	3.83e-01 ± 4.9e-03	–	4.948e-01 ± 3.296e-03
	DTLZ5	1.91e-01 ± 1.73e-02	1.9848e-01 ± 2.00e-04	9.469711e-02 ± 5.7e-06	9.32e-02 ± 1.3e-04	1.9540e-01 ± 8.49e-04	3.109e-01 ± 5.511e-06
	DTLZ6	9.83e-02 ± 1.07e-03	1.8971e-01 ± 3.80e-02	9.566277e-02 ± 8.1e-07	9.50e-02 ± 2.8e-05	1.6980e-01 ± 7.05e-02	2.976e-01 ± 7.947e-07
	DTLZ7	3.30e-01 ± 5.53e-03	2.4777e-01 ± 5.56e-03	2.723934e-01 ± 2.1e-02	2.88e-01 ± 3.5e-02	2.6578e-01 ± 3.08e-03	3.623e-01 ± 2.934e-03
	WFG1	–	4.8681e-01 ± 3.98e-02	6.348506e-01 ± 2.1e-04	1.41e-01 ± 8.4e-03	6.2754e-02 ± 2.03e-02	6.945e-01 ± 2.056e-04
	WFG2	9.16e-01 ± 1.33e-03	8.9761e-01 ± 3.30e-02	5.646771e-01 ± 1.2e-05	5.53e-01 ± 9.0e-04	9.0951e-01 ± 2.61e-03	6.077e-01 ± 1.771e-05
	WFG3	–	3.7878e-01 ± 6.74e-03	4.980028e-01 ± 4.5e-06	4.91e-01 ± 6.7e-04	3.0146e-01 ± 1.17e-02	7.010e-01 ± 4.028e-06
	WFG4	4.49e-01 ± 6.37e-03	5.0335e-01 ± 5.88e-03	2.212260e-01 ± 1.1e-04	2.08e-01 ± 6.2e-04	4.8720e-01 ± 4.14e-03	5.784e-01 ± 1.102e-04
	WFG5	4.49e-01 ± 7.32e-03	4.8212e-01 ± 6.91e-03	1.981450e-01 ± 1.0e-05	2.20e-01 ± 2.9e-03	4.7832e-01 ± 7.30e-03	5.382e-01 ± 9.450e-06
	WFG6	4.32e-01 ± 1.34e-03	4.4840e-01 ± 1.25e-02	2.128722e-01 ± 4.2e-06	2.06e-01 ± 1.2e-03	5.1364e-01 ± 1.09e-02	6.101e-01 ± 3.723e-06
	WFG7	4.39e-01 ± 4.17e-03	5.1179e-01 ± 4.30e-03	2.128553e-01 ± 5.7e-06	2.08e-01 ± 3.8e-04	5.2921e-01 ± 3.44e-03	5.928e-01 ± 4.512e-06

Table 7. SPA and SP values for tri-objective test functions

	Function	MOIMPA [21]	ε-MOEA [26]	MOVPS [37]	RMOABC [48]	Proposed MOVPS
SPA values	DTLZ1	2.10e-02 ± 1.81e-03	–	0.07461 ± 0.00711	–	1.431e-02 ± 3.017e-04
	DTLZ2	4.96e-02 ± 1.54e-03	–	0.00862 ± 0.00309	–	3.358e-02 ± 3.234e-04
	DTLZ3	–	–	–	–	2.849e-02 ± 7.962e-05
	DTLZ4	4.12e-02 ± 2.84e-03	–	0.01005 ± 0.00073	–	3.072e-03 ± 2.183e-05
	DTLZ5	1.08e-02 ± 1.84e-03	–	–	–	3.511e-03 ± 1.775e-05
	DTLZ6	1.73e-02 ± 5.44e-03	–	–	–	5.907e-03 ± 9.709e-06
	DTLZ7	–	–	–	–	5.613e-03 ± 9.911e-06
	WFG1	–	3.000e-02 ± 4.651e-03	–	4.409e-01 ± 2.246e-02	2.939e-03 ± 1.145e-04
	WFG2	–	7.093e-02 ± 1.079e-02	–	1.406e-01 ± 2.625e-02	1.818e-03 ± 1.092e-04
	WFG3	–	3.668e-02 ± 2.593e-03	–	4.222e-02 ± 2.523e-03	1.016e-02 ± 9.552e-04
	WFG4	–	5.002e-02 ± 2.149e-03	–	6.549e-02 ± 6.137e-03	6.103e-03 ± 5.117e-04
	WFG5	–	5.034e-02 ± 2.364e-03	–	7.442e-02 ± 9.141e-03	1.120e-02 ± 2.098e-04
	WFG6	–	6.228e-02 ± 3.131e-03	–	8.773e-02 ± 6.767e-03	1.007e-02 ± 5.327e-04
	WFG7	–	4.877e-02 ± 2.382e-03	–	6.723e-02 ± 4.472e-03	5.564e-03 ± 2.018e-04

	Function	MOHGD [42]	EMOSO [56]	MOSG [28]	eMOABC [6]	Proposed MOVPS
SP values	DTLZ1	7.48e-01 ± 4.1e-02	–	6.80e−01 ± 7.8e−02	7.164e−01 ± 3.3e−02	4.319e-02 ± 4.071e-03
	DTLZ2	6.35e-01 ± 3.4e-02	1.4472e−01 ± 1.93e−02	6.27e−01 ± 3.6e−02	6.357e−01 ± 3.7e−02	3.358e-02 ± 3.212e-03
	DTLZ3	6.53e-01 ± 3.8e-02	–	–	6.223e−01 ± 4.0e−02	6.014e-02 ± 4.109e-03
	DTLZ4	6.51e-01 ± 3.8e-02	–	6.4e−01 ± 3.7e−02	9.999e−01 ± 3.3e−01	1.027e-01 ± 2.819e-04
	DTLZ5	1.49e-01 ± 4.5e-02	2.3368e−01 ± 2.41e−02	3.04e−01 ± 3.2e−02	1.261e−01 ± 1.4e−02	3.115e-02 ± 1.775e-04
	DTLZ6	9.73e-02 ± 1.2e-02	3.1101e−01 ± 1.37e−01	1.03e−01 ± 1.5e−02	1.109e−01 ± 1.4e−02	5.907e-03 ± 9.709e-04
	DTLZ7	7.13e-01 ± 4.4e-02	3.0914e−01 ± 3.53e−02	6.73e−01 ± 5.8e−02	7.205e−01 ± 5.6e−02	2.074e-03 ± 8.268e-04
	WFG1	5.82e-01 ± 2.2e-01	1.0067e+00 ± 2.81e−02	8.85e−01 ± 8.6e−02	8.685e−01 ± 8.4e−02	1.113e-01 ± 2.135e-04
	WFG2	7.67e-01 ± 4.2e-03	4.9065e−01 ± 5.08e−02	9.57e−01 ± 4.2e−02	8.984e−01 ± 7.6e−02	1.128e-03 ± 5.062e-04
	WFG3	5.98e-02 ± 6.2e-03	3.0144e−01 ± 3.02e−02	1.19e−01 ± 1.2e−02	4.013e−01 ± 1.6e−02	5.986e-02 ± 5.854e-03
	WFG4	1.45e-01 ± 1.6e-02	2.4646e−01 ± 2.21e−02	2.39e−01 ± 2.4e−02	1.355e−01 ± 1.4e−02	3.127e-03 ± 5.673e-04
	WFG5	1.47e-01 ± 8.6e-03	2.2608e−01 ± 2.00e−02	1.36e−01 ± 1.8e−02	1.481e−01 ± 1.8e−02	1.089e-02 ± 2.908e-04
	WFG6	1.93e-01 ± 4.6e-02	2.2360e−01 ± 2.43e−02	2.24e−01 ± 2.9e−02	1.483e−01 ± 4.0e−02	4.113e-03 ± 4.473e-04
	WFG7	1.25e-01 ± 1.2e-02	2.0938e−01 ± 2.07e−02	2.42e−01 ± 2.4e−02	1.179e−01 ± 1.2e−02	1.007e-02 ± 1.991e-04

Table 8. IGD and HV values for many-objective test functions

	Function	MaOEA-CS [53]	SAEMO [58]	IREA [60]	TEEA [57]	HH_EG [53]	Proposed MOVPS
IGD values	MaF1	2.273e-01 ± 1.3e-03	6.6863e-01 ± 5.7785e-02	2.275e-01 ± 2.59e-03	2.33e-01 ± 7.45e-03	2.8638e-01 ± 5.66e-03	2.397e-01 ± 8.019e-04
	MaF2	2.424e-01 ± 2.2e-02	3.1078e-01 ± 2.0800e-02	1.743e-01 ± 9.00e-03	1.53e-01 ± 2.99e-03	2.3511e-01 ± 4.49e-02	9.656e-02 ± 2.682e-03
	MaF3	1.049e-01 ± 3.1e-03	2.5746e+05 ± 1.3368e+05	1.025e-01 ± 3.26e-03	8.36e-02 ± 2.20e-03	3.3780e+04 ± 1.23e+05	7.797e-02 ± 1.143e-03
	MaF4	5.290e+01 ± 2.4e+00	8.2728e+04 ± 2.0907e+04	5.304e+01 ± 3.68e+00	7.50e+01 ± 2.25e+00	9.9485e+01 ± 7.10e+00	4.819e+01 ± 1.529e+00
	MaF5	4.727e+01 ± 1.7e+00	9.3240e+01 ± 2.1285e+01	4.555e+01 ± 1.53e+00	8.82e+01 ± 1.19e+00	7.8122e+01 ± 7.96e-01	3.978e+01 ± 5.921e-01
	MaF6	2.599e-03 ± 2.2e-05	1.1201e+00 ± 1.1541e+00	1.746e-03 ± 1.34e-05	2.75e-03 ± 6.25e-04	3.6128e-01 ± 1.08e-01	8.817e-03 ± 1.037e-05
	MaF7	8.983e-01 ± 1.6e-02	7.3245e+00 ± 1.6668e+00	8.128e-01 ± 4.60e-03	8.35e-01 ± 1.03e-01	1.0356e+00 ± 7.35e-02	7.993e-01 ± 3.720e-03
	MaF8	1.154e-01 ± 2.9e-03	2.1237e+00 ± 1.8094e+00	4.541e-02 ± 2.71e-03	1.35e-01 ± 4.99e-03	–	3.866e-02 ± 2.473e-03
	MaF9	1.214e-01 ± 1.6e-02	4.6610e+00 ± 5.5160e+00	4.307e-02 ± 3.11e-04	1.85e-01 ± 1.27e-02	–	1.172e-02 ± 2.156e-04
	MaF10	1.794e+00 ± 1.6e-01	4.5134e+00 ± 2.4950e-01	8.717e-01 ± 2.65e-02	1.02e-00 ± 2.98e-02	–	6.658e-01 ± 9.827e-03

	Function	MaOEA-CS [53]	MaOEA/C [41]	MultiGPO [61]	AGPSO [10]	NSGA-III [43]	Proposed MOVPS
HV values	MaF1	3.226e-07 ± 5.4e-07	5.646e-07 ± 2.80e-08	–	6.16e-07 ± 3.49e-08	2.353e-07 ± 6.2e-08	7.500e-07 ± 1.593e-08
	MaF2	2.127e-01 ± 5.8e-03	2.436e-01 ± 2.52e-03	2.263e-01 ± 2.90e-03	2.45e-01 ± 2.08e-03	2.093e-01 ± 9.7e-03	3.214e-01 ± 6.594e-04
	MaF3	1.000e+00 ± 9.2e-06	9.919e-01 ± 7.00e-03	2.769e-01 ± 4.36e-01	9.85e-01 ± 2.26e-03	5.230e-02 ± 2.0e-01	9.969e-01 ± 1.310e-06
	MaF4	5.210e-05 ± 1.3e-05	4.716e-04 ± 3.56e-05	8.466e-05 ± 1.35e-05	5.66e-04 ± 1.89e-05	2.067e-04 ± 1.6e-05	6.511e-04 ± 8.672e-06
	MaF5	9.538e-01 ± 1.9e-03	1.000e+00 ± 4.41e-06	9.122e-01 ± 7.18e-03	9.67e-01 ± 1.32e-03	9.664e-01 ± 7.2e-03	9.788e-01 ± 2.62e-06
	MaF6	1.012e-01 ± 2.5e-04	1.008e-01 ± 1.89e-04	6.122e-02 ± 4.33e-02	9.22e-02 ± 1.18e-02	4.302e-02 ± 3.1e-02	5.094e-01 ± 7.808e-05
	MaF7	1.695e-01 ± 7.2e-03	2.006e-01 ± 1.07e-02	9.403e-02 ± 2.38e-02	1.83e-01 ± 3.18e-02	1.386e-01 ± 9.7e-03	4.967e-01 ± 5.051e-03
	MaF8	1.097e-02 ± 1.1e-04	–	1.121e-02 ± 5.81e-05	–	8.885e-03 ± 3.2e-04	2.870e-02 ± 4.173e-05
	MaF9	1.851e-02 ± 2.3e-04	–	1.883e-02 ± 1.41e-04	–	8.166e-03 ± 1.9e-03	2.035e-02 ± 1.108e-04
	MaF10	6.001e-01 ± 6.3e-02	–	–	–	9.991e-01 ± 3.6e-04	1.000e+00 ± 2.378e-06

Table 9. Results of unimodal and multimodal benchmark functions

	Function	FDO [1]	WOA [17]	VLE [13]	CSO [1]	Proposed MOVPS
Unimodal	F1	2.13e-23 ± 1.06e-22	3.84e-11 ± 6.5e-11	4.498e-07 ± 1.41e-06	3.50e-14 ± 6.34e-14	6.717e-28 ± 6.334e-15
	F2	0.047175 ± 0.188922	2.38e-08 ± 4.27e-08	3.084e-06 ± 6.049e-06	2.68e-08 ± 2.61e-08	7.158e-17 ± 3.389e-08
	F3	2.39e-06 ± 1.28e-05	9. 84e-04 ± 2.67e+07	5.2020 ± 0.79863	7.17e–09 ± 1.16e-08	6.947e-09 ± 8.911e-09
	F4	4.93e-08 ± 9.09e-08	5. 74e+01 ± 2.82e+01	–	0.010352 ± 0.007956	4.901e-08 ± 2.407e-08
	F5	21.58376 ± 39.66721	2. 88e+01 ± 3.9e-02	79.199 ± 37.400	8.587858 ± 0.598892	22.768 ± 2.653e-02
	F6	7.15e-22 ± 2.80e-21	2. 22 ± 6.43e-01	–	1.151759 ± 0.431511	0.874 ± 2.969e-05
	F7	0.612389 ± 0.299315	3. 55e-02 ± 3.66e-02	–	0.026026 ± 0.015039	0.001 ± 7.692e-04

	Function	FDO [1]	WOA [17]	VLE [13]	CSO [1]	Proposed MOVPS
Multimodal	F8	- 10502.1 ± 15188.77	-1. 63e+03 ± 0.00	-1.256e+04 ± 68.705	- 2855.11 ± 359.1697	-1.356e+04 ± 0.0000
	F9	7.940883 ± 4.110302	3. 89 ± 1.91e+01	34.583 ± 17.886	24.01772 ± 6.480946	0.298 ± 7.448
	F10	7.76e-15 ± 2.46e-15	9. 52e-07 ± 1.44e-06	3.17 ± 3.921	3.754226 ± 1.680534	1.059e-13 ± 1.017e-06
	F11	0.175694 ± 0.148586	3. 68e-02 ± 1.84e-01	0.507 ± 0.504	0.355631 ± 0.19145	0.004 ± 1.318e-04
	F12	7.737715 ± 4.714534	2. 1e-01 ± 1.56e-01	0.236 ± 0.287	1.900773 ± 1.379549	0.053 ± 0. 019
	F13	4.724571 ± 6.448214	1. 37 ± 3.82e-01	–	1.160662 ± 0.53832	0.654 ± 9.38e-06

Table 10. Results of fixed dimension multimodal and composite benchmark functions

	Function	FDO [1]	WOA [17]	VLE [13]	CSO [1]	MOVPS
Fixed dimension multimodal	F14	2.448453 ± 1.766953	5.41 ± 4.84	0.998 ± 2.52e-07	0.998004 ± 3.39e-07	4.042 ± 2.345e-05
	F15	0.001492 ± 0.003609	0.002 ± 4.08e-03	–	0.001079 ± 0.00117	3.01e-4 ± 0. 00002
	F16	- 1.00442 ± 0.149011	-1.0316 ± 1.61e-06	-1.031 ± 1.84e-04	- 1.03162 ± 1.53e-05	-1.0318 ± 1.02e-06
	F17	0.397887 ± 5.17e-15	0.399 ± 2.66e-03	0.398 ± 4.569e-04	0.304253 ± 1.81e-06	0.302 ± 2.189e-15
	F18	3.0000 ± 2.37e-07	7.53 ± 1.06e+01	3.009 ± 1.625e-02	3.003667 ± 0.004338	3.000 ± 9.877e-08
	F19	- 3.86015 ± 0.003777	-3.827 ± 1.54e-01	-3.862 ± 6.688e-05	- 3.8625 ± 0.00063	-3.8916 ± 7.324e-06
	F20	- 3.06154 ± 0.380813	-3.14 ± 2.88e-01	-3.317 ± 2.131e-02	- 3.30564 ± 0.045254	-3.32586 ± 9.878e-03
	F21	- 4.19074 ± 2.664305	-6.16 ± 2.55	–	- 9.88163 ± 0.90859	-10.151 ± 0.8971
	F22	- 4.89633 ± 3.085016	-6.05 ± 2.27	–	- 10.2995 ± 0.094999	-10.4034 ± 7.018e-04
	F23	- 4.03276 ± 2.517357	-5.65 ± 2.17	–	- 10.0356 ± 1.375583	-10.5343 ± 1.0589

	Function	QCGA [22]	BA_ABC [54]	dBA [9]	IAgGWO [24]	MOVPS
Composite	F24	216.5 ± 2.21e+02	4.43e+02 ± 1.39e+02	2.075e+02 ± 6.305e+01	–	1.328e+02 ± 9.71365
	F25	361.2 ± 4.66e+01	1.29e+02 ± 1.76e+01	1.511e+02 ± 1.083e+01	–	1.251e+02 ± 8.81064
	F26	910.0 ± 5.30e-09	5.00e+02 ± 1.97e+02	4.853e+02 ± 1.689e+02	–	4.503e+01 ± 4.879e-10
	F27	910.0 ± 1.68e-03	6.15e+02 ± 1.96e+02	4.205e+02 ± 1.847e+02	1.570e+03 ± 4.267e+02	6.497e+01 ± 1.317e-03
	F28	918.9 ± 4.45e+01	5.28e+02 ± 1.99e+02	3.964e+02 ± 1.677e+02	1.710e+03 ± 3.801e+02	5.684e+01 ± 38.51207
	F29	1616.7 ± 1.64e+02	4.45e+02 ± 1.57e+02	4.080e+02 ± 8.121e+01	1.843e+03 ± 2.119e+02	3.695e+02 ± 41.03148
	F30	1273.5 ± 5.28e+01	7.29e+02 ± 1.29e+02	6.307e+02 ± 1.203e+02	1.744e+03 ± 1.779e+02	5.112e+02 ± 19.19403
	F31	1615.1 ± 5.17e+01	5.24e+02 ± 5.04e+01	5.464e+02 ± 3.703e+01	1.713e+03 ± 7.413e+02	5.099e+02 ± 10.41957
	F32	1361.1 ± 3.63e+02	2.00e+02 ± 1.53e-12	2.000e+02 ± 8.343e-13	1.274e+03 ± 2.407e+02	2.651e+02 ± 5.93e-14