Expert Systems With Applications 226 (2023) 120184

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

A dual-stage large-scale multi-objective evolutionary algorithm with dynamic learning strategy

Jie Cao [a,b], Kaiyue Guo [a,b], Jianlin Zhang [a,b,*], Zuohan Chen [a,b]

[a] School of Computer and Communication Technology, Lanzhou University of Technology, Lanzhou 730050, China

[b] Gansu Engineering Research Center of Manufacturing Information, Lanzhou University of Technology, Lanzhou 730050, China

ARTICLE INFO

Keywords:
Large-scale optimization
Multi-objective optimization
Dual-stage optimization strategy
Dynamic learning strategy

ABSTRACT

Large-scale multi-objective optimization problems (LSMOPs) bring significant challenges due to their large number of decision variables. Most of the existing algorithms fail to obtain high-quality solutions for the LSMOPs. To remedy this issue, an algorithm named dual-stage large-scale multi-objective evolutionary algorithm with dynamic learning strategy (DLMOEA-DLS) is proposed in this paper. In the DLMOEA-DLS, the entire evolution process mainly includes two stages, and each stage plays a different role in the searching process. In the first stage, the decision variables are clustering into two categories to be optimized independently for the convergence of the population. In the second stage, a dynamic learning strategy is designed to generate new offspring, in which each solution learns from a leader with better fitness and coupled control parameter for each solution is adaptively updated by learning from the historical behaviors of the solution. Moreover, an environmental selection operator is adopted to reserve promising solutions for the next iteration. To verify the performance of the DLMOEA-DLS, five state-of-the-art algorithms are used for comparison on 36 LSMOP benchmark instances, 48 LMF benchmark instances, and 6 real-world TREE benchmark instances. The experimental results demonstrate the superiority of the DLMOEA-DLS over the five state-of-the-art algorithms.

1. Introduction

In the real world, many problems can be attributed to multi-objective optimization problems (MOPs) (Cao, Zhao, Gu, Ling, & Ma, 2020; Tian, Yang, Zhang, Duan, & Zhang, 2019), which involve multiple conflicting objectives. The mathematical formulation of an MOP is summarized as:

$$\begin{array}{ll} \text{minimize}: & F(x) = (f_1(x), f_2(x), \ldots, f_m(x))^T \\ \text{subject to}: & x = (x_1, x_2, \ldots, x_n) \in \Omega \in \mathbb{R}^n \end{array} \tag{1}$$

where x and Ω denote the decision vector and the decision space, respectively. A single solution cannot minimize all of the conflicting objectives. Hence, instead of a single solution, a set of trade-off solutions is expected to be found for each MOP. Suppose x_A and x_B are the solutions of an MOP, x_A is known as Pareto dominated x_B, if and only if, $\forall i \in (1, 2, \cdots, m)$, $f_j(x_A) \le f_j(x_B)$, and $\exists j \in \{1, 2, \cdots, m\}$, $f_j(x_A) < f_j(x_B)$. All the optimal solutions in the objective space are named as Pareto front (PF), while they are defined as Pareto set (PS) in the decision space.

In the last two decades, a series of multi-objective evolutionary algorithms (MOEAs) have been proposed to address the MOPs (Tang, Yu, Zou, Yang, & Zheng, 2022; Yang, Zou, Yang, Zheng, & Liu, 2021). According to the environmental selection mechanism, the MOEAs can be roughly divided into the dominance-based (Deb, Pratap, Agarwal, & Meyarivan, 2002; Iorio & Li, 2004), the decomposition-based (Liu et al., 2019; Song, Yang, Chen, & Zhang, 2016; Zhang & Li, 2008), and the indicator-based MOEAs (Li, Yen, Sahoo, Chang, & Gu, 2021; Song, Wang, & Xu, 2022). In most cases, the existing MOEAs can solve the small-scale MOPs well (He, Cheng, & Yazdani, 2022; Liu, Lin, Tian, & Tan, 2021). However, most of MOEAs will achieve a poor performance in approaching the true PF for the MOPs with hundreds or even thousands of decision variables. Specifically, the LSMOPs refer to the MOPs with more than 100 decision variables (Jinlu, Lixin, Rui, Hao, & Ziyu, 2022; Tian, Zheng, Zhang, & Jin, 2020). With the number of decision variables increasing, the search space will grow exponentially, which leads to the "curse of dimensionality" (He et al., 2019). In recent years, numerous large-scale multi-objective evolutionary algorithms (LSMOEAs) have been proposed to solve the above problems. The existing LSMOEAs can be roughly classified into four categories.

The first category is based on CC. This category adopts grouping mechanisms (e.g., random grouping (Omidvar, Li, Yang, & Yao, 2010),

* Corresponding author at: School of Computer and Communication Technology, Lanzhou University of Technology, Lanzhou 730050, China.
E-mail address: zhangjl_lut@lut.edu.cn (J. Zhang).

https://doi.org/10.1016/j.eswa.2023.120184
Received 30 March 2022; Received in revised form 31 January 2023; Accepted 15 April 2023
Available online 18 April 2023
0957-4174/

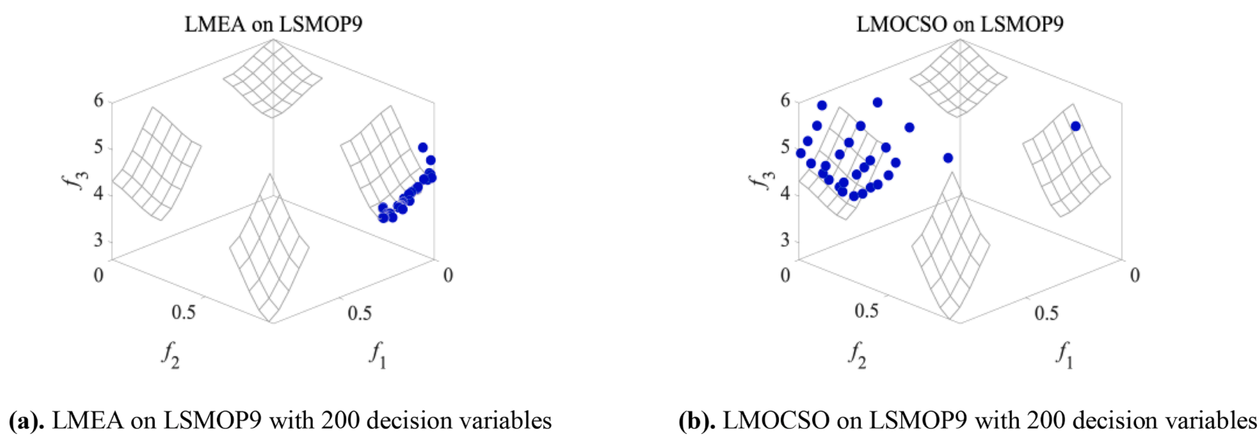

Fig. 1. (a). LMEA on LSMOP9 with 200 decision variables **(b).** LMOCSO on LSMOP9 with 200 decision variables.

linear grouping (Song et al., 2016), ordered grouping (Tan, Feng, & Jiang, 2021), and differential grouping (Li & Wei, 2018; Omidvar, Li, Mei, & Yao, 2014)) to divide the decision variables into several groups. Then each group is cooperatively optimized, and the new solution is formed via the decision variables from each group. For example, a representative algorithm, named CCGDE3 (Antonio & Coello, 2013), randomly divides the decision variables into several groups of equal size by random grouping. Each group is cooperatively optimized by the GDE3 (Kukkonen & Lampinen, 2005) to form the offspring in each iteration. A decomposition-based approach named MOEA/D^2 (Miguel Antonio & Coello Coello, 2016) has been proposed for solving the LSMOPs. The MOEA/D^2 makes use of the divide-and-conquer technology and divides the decision variables into several subcomponents. Each subcomponent is optimized by the MOEA/D (Zhang & Li, 2008).

The second category is based on decision variables analysis and decision variable clustering. This category analyzes the effect of each decision variable for each objective. The decision variables with the same effect tend to be clustered in the same group and then each group is optimized separately. A representative approach of this category is MOEA/DVA (Ma et al., 2016), which classifies the decision variables into position variables, distance variables, and mixed variables by analyzing the position characteristics in objective space in terms of relations with decision variables. After that, each type of decision variable is optimized as an independent subcomponent. Meanwhile, some algorithms with similar idea have also been proposed, such as the LMEA (Zhang, Tian, Cheng, & Jin, 2018) and the DPCCMOEA (Cao, Zhao, Lv, & Liu, 2017). The importance of decision variables to objectives is analyzed in the LVIDE (Liu, Lin, Tian, et al., 2021). Specifically, the differential value of each objective caused by each decision variable is recorded, and all the decision variables are partitioned into several subsets according to the differential values. At each iteration, several subsets are randomly selected to be evolved.

The third category is based on problem transformation, which reduces the dimensionality of the decision space, and optimizes the transformed problem instead of the original problem. A weighted optimization framework (WOF) (Zille, Ishibuchi, Mostaghim, & Nojima, 2018) divides the decision variables into several groups, where each decision variables group is associated with a weight value according to a dimensionality reduction formula. Afterwards, the weight value is optimized instead of the decision variables. Hence, the original LSMOP is transformed into a small-scale MOP, which improves the efficiency of the searching process. In recent years, a large number of algorithms based on the idea of the WOF have been proposed (Junhao, Lingjie, Qiuzhen, & Zhong, 2021; Liu, Liu, Li, & Liu, 2020). In the LSMOF (He et al., 2019), two weight variables are associated with a solution to update the position along with the two weight directions for generating promising offspring. The LSMOF optimizes a set of weight variables for several solutions while the WOF optimizes a bunch of weight values for each solution.

The fourth category is based on novel searching mechanism. For example, the LMOCSO (Tian et al., 2020) randomly selects two solutions one as winner and the other as loser. Then the loser is directly optimized by the winner through a learning mechanism. The LMOCSO does not employ any grouping strategy and decision variables clustering method. The DGEA (He et al., 2022) generates offspring along direction vectors. A solution is randomly selected from a nondominated set as start point, and end points are chosen adaptively from a dominated or a nondominated set, which enhances the convergence or the diversity of the population. Then the direction vectors are formed by the start and the end points. Moreover, considerable attention has also been given to the research concentrating on combining machine learning with optimization algorithms for dealing with the LSMOPs (Wang, Zhan, Kwong, Jin, & Zhang, 2021).

Most existing algorithms employ grouping strategies to reduce the dimensionality of the LSMOPs and achieve competitive results. However, grouping strategies have some limitations (Liu, Lin, Li, & Tan, 2021). In the case of many decision variables with strong interaction, unreasonable decision variables groups can mislead the searching process and cause the population falling into a local optimum. The algorithms with novel searching method evolve the decision variables simultaneously without grouping strategies. However, the convergence speed of the population becomes slow in the evolution process. Especially, the final optimal solutions become worsen when solving problems with complex landscapes.

The above analysis shows that few LSMOEAs can balance the convergence and the diversity of the final population well, the search efficiency still needs to be improved. This paper proposes a dual-stage LSMOEA with dynamic learning strategy to address the LSMOPs. In the proposed algorithm, two different operators are performed to enhance the convergence and the diversity of the population in each stage. The main contributions of this paper are summarized as follows:

(1) A dual-stage optimization strategy is adopted to divide the entire evolution process into convergence and diversity stages. Each stage has a different emphasis, the convergence stage is concentrated to accelerate the convergence of the population, and the diversity stage focuses on enhancing the diversity of the population.
(2) A dynamic learning strategy (DLS) is designed to improve the diversity of the population. At each time, two solutions are randomly selected. The solution with worse fitness learns from the other one to generate promising offspring, in which the coupled control parameter is dynamically reinitialized by learning from the previous behaviors of the solution.

Table 1
The IGD values of compared algorithms on tri-objective LSMOP1-LSMOP9, where the best result is shown in bold font and gray background.

Problem	Dec	DGEA	LMEA	LMOCSO	LMOEA-DS	WOF	DLMOEA-DLS
LSMOP1	100	1.5847e−1 (3.60e−2) -	6.3736e−2 (4.83e−2) -	**3.1014e−2 (5.04e−5) =**	1.9165e−1 (4.38e−2) -	6.8666e−2 (1.99e−2) -	3.3457e−2 (4.54e−2)
	200	2.7199e−1 (3.59e−2) -	6.2584e−2 (4.83e−2) -	**3.1003e−2 (2.93e−5) =**	2.4484e−1 (5.09e−2) -	7.5976e−2 (1.45e−2) -	3.2912e−2 (4.47e−4)
	500	2.9776e−1 (1.02e−2) -	5.7076e−2 (3.57e−2) -	**3.0981e−2 (2.63e−5) =**	2.9377e−1 (2.38e−2) -	8.3624e−2 (1.94e−2) -	3.2086e−2 (2.04e−4)
	1000	3.0828e−1 (4.51e−3) -	1.3650e−1 (1.55e−1) -	**3.0958e−2 (1.49e−5) +**	3.1978e−1 (1.69e−2) -	1.0007e−1 (3.16e−2) -	3.2244e−2 (4.48e−4)
LSMOP2	100	8.6955e−2 (1.91e−2) -	7.4176e−2 (7.77e−2) -	**3.8882e−2 (7.57e−4) +**	8.2578e−2 (1.35e−3) -	9.5600e−2 (1.94e−2) -	5.6152e−2 (3.91e−3)
	200	4.9754e−2 (1.10e−3) =	7.5035e−2 (2.50e−2) -	**3.6315e−2 (7.33e−4) +**	5.9498e−2 (1.44e−3) -	6.1712e−2 (3.47e−3) -	4.9379e−2 (4.57e−3)
	500	3.6766e−2 (4.95e−4) =	7.2330e−2 (7.78e−2) -	**3.3697e−2 (2.65e−4) +**	4.2771e−2 (1.94e−3) -	3.9872e−2 (9.78e−4) -	3.6766e−2 (1.40e−3)
	1000	3.3515e−2 (1.47e−3) -	1.3388e−1 (1.62e−1) -	**3.2289e−2 (6.18e−5) =**	3.7004e−2 (1.03e−3) -	3.4401e−2 (7.33e−4) -	3.2888e−2 (1.84e−4)
LSMOP3	100	5.2289e−1 (3.57e−2) -	4.6919e−1 (1.55e−1) -	3.1280e−1 (7.98e−2) -	6.5206e−1 (1.12e−1) -	4.5644e−1 (1.92e−1) -	**1.7476e−1 (8.84e−2)**
	200	5.3543e−1 (2.91e−2) -	4.6303e−1 (1.23e−1) -	3.8222e-1 (6.39e-2) -	7.8440e-1 (5.27e-2) -	5.0265e-1 (1.98e-1) -	**2.1254e−1 (1.67e−1)**
	500	5.3410e-1 (1.60e-2) -	8.1571e-1 (6.98e-1) -	3.8119e-1 (2.15e-2) -	8.2647e-1 (4.07e-2) -	6.1056e-1 (2.07e-1) -	**1.9359e−1 (5.87e−2)**
	1000	5.4358e-1 (1.80e-2) -	2.4834e+0 (3.06e+0) -	5.8396e−1 (8.59e−2) -	8.0075e−1 (4.42e−2) -	5.4258e−1 (1.95e−1) -	**3.9520e−1 (2.76e−2)**
LSMOP4	100	1.4196e−1 (1.30e−2) -	1.0598e−1 (8.32e−2) -	**3.6326e−2 (1.31e−3) +**	1.9950e−1 (2.16e−2) -	1.1860e−1 (1.53e−2) -	5.4239e−2 (2.45e−3)
	200	1.3085e−1 (6.31e−3) -	1.2702e−1 (1.03e−1) -	**3.3389e−2 (7.72e−4) +**	1.2859e−1 (3.95e−3) -	1.1358e−1 (5.34e−3) -	4.4120e−2 (2.25e−3)
	500	8.0673e−2 (2.15e−3) -	6.2411e−2 (2.84e−2) -	**3.1800e−2 (1.92e−4) +**	9.4373e−2 (1.82e−3) -	7.4755e−2 (4.99e−3) -	3.8994e−2 (8.87e−3)
	1000	5.2187e−2 (5.13e−4) -	5.8481e−2 (1.72e−2) -	**3.1357e−2 (9.32e−5) +**	6.4797e−2 (1.55e−3) -	5.3636e−2 (3.39e−3) -	3.4057e−2 (4.47e−4)
LSMOP5	100	2.5565e−1 (1.64e−1) -	6.7869e−1 (6.94e−1) -	**4.0988e−2 (9.97e−6) =**	2.7273e−1 (2.25e−2) -	3.5494e−1 (4.76e−2) -	4.3102e−2 (9.57e−2)
	200	2.4156e−1 (7.84e−2) -	1.7009e+0 (1.50e+0) -	4.2724e−2 (7.90e−4) -	2.7771e−1 (7.44e−3) -	3.1383e−1 (9.35e−2) -	**4.0986e−2 (1.29e−5)**
	500	2.2870e−1 (3.96e−3) =	3.7139e+0 (3.04e+0) -	8.5221e−2 (1.88e−1) -	2.7148e−1 (1.11e−2) =	2.3493e−1 (1.22e−1) =	**4.0984e−2 (9.25e−6)**
	1000	2.2969e−1 (3.74e−3) -	5.9856e+0 (4.27e+0) -	8.8076e−2 (2.26e−1) -	2.8191e−1 (1.02e−2) -	2.1069e−1 (9.18e−2) -	**8.6230e−2 (2.02e−1)**
LSMOP6	100	6.0089e−1 (1.63e−1) =	2.1129e+1 (6.56e+1) -	7.1039e−1 (1.18e−1) -	6.1759e−1 (1.02e−1) -	7.1699e−1 (2.87e−2) -	**5.3930e−1 (5.84e−2)**
	200	6.0315e−1 (7.75e−2) =	4.2305e+1 (1.61e+2) -	9.2349e−1 (2.08e−1) -	7.6588e−1 (1.02e−1) -	9.5610e−1 (3.87e−2) -	**5.6746e−1 (4.06e−2)**
	500	5.7141e−1 (2.47e−2) -	6.1888e+1 (2.70e+2) -	1.1851e+0 (2.82e−1) -	8.1075e−1 (1.50e−1) -	1.1654e+0 (1.10e−1) -	**5.2040e−1 (8.34e−2)**
	1000	5.7527e−1 (3.12e−2) -	8.1785e+2 (2.16e+3) -	1.0502e+0 (5.77e−1) -	8.6736e−1 (2.17e−1) -	1.2474e+0 (1.26e−1) -	**5.3453e−1 (2.34e−2)**
LSMOP7	100	7.2258e−1 (5.68e−2) -	1.9001e+0 (8.23e−1) -	8.7231e−1 (1.81e−1) -	7.2081e−1 (3.91e−2) -	7.1344e−1 (9.45e−3) -	**6.0518e−1 (1.15e−1)**
	200	7.7903e−1 (3.96e−2) -	1.5240e+0 (5.23e−1) -	9.4593e−1 (3.15e−7) -	8.2666e−1 (2.63e−2) -	7.7778e−1 (5.40e−3) -	**5.5410e−1 (1.89e−1)**
	500	7.5870e−1 (9.62e−2) -	9.1082e−1 (9.53e−2) -	9.4593e−1 (1.01e−7) -	8.4255e−1 (2.71e−2) -	8.2643e−1 (4.11e−2) -	**6.5371e−1 (1.69e−1)**
	1000	7.4741e−1 (5.27e−2) -	6.2373e−1 (1.03e−1) -	9.3889e−1 (1.79e−2) -	8.3981e−1 (7.12e−3) -	7.9877e−1 (9.70e−2) -	**5.3624e−1 (8.58e−3)**
LSMOP8	100	1.4064e−1 (6.64e−2) -	1.8987e−1 (6.49e−2) -	**4.8817e−2 (3.77e−3) +**	3.0650e−1 (5.85e−2) -	1.1190e−1 (6.19e−2) =	9.6931e−2 (1.90e−2)
	200	8.9838e−2 (4.54e−2) -	1.1693e−1 (2.53e−2) -	**5.0961e−2 (2.14e−3) +**	1.5932e−1 (7.36e−2) -	8.3977e−2 (1.57e−2) -	6.1129e−2 (4.68e−3)
	500	6.2831e−2 (1.84e−3) -	7.1856e−2 (7.77e−3) -	**4.9174e−2 (1.45e−3) +**	1.1634e−1 (1.36e−2) -	8.7778e−2 (2.33e−2) -	5.3716e−2 (2.66e−3)
	1000	6.0426e−2 (1.08e−3) -	6.2552e−2 (2.19e−3) -	**4.3835e−2 (8.22e−4) =**	1.1403e−1 (1.38e−2) -	8.9183e−2 (3.35e−2) -	4.4243e−2 (1.66e−3)
LSMOP9	100	5.2411e−1 (6.72e−2) -	6.6364e−1 (1.29e−1) -	5.7370e−1 (3.34e−1) -	5.8609e−1 (1.43e−3) -	1.1677e+0 (8.73e−2) -	**3.5200e−1 (7.00e−2)**
	200	5.1747e−1 (7.20e−2) -	7.5950e−1 (2.75e−1) -	4.1077e−1 (2.57e−1) -	5.8109e−1 (2.17e−3) -	1.1737e+0 (1.57e−1) -	**3.6078e−1 (1.08e−2)**
	500	5.4086e−1 (8.80e−2) -	8.4357e−1 (4.59e−1) -	3.5108e−1 (3.76e−3) -	5.7677e−1 (1.68e−3) -	1.0341e+0 (3.11e−1) -	**2.2569e−1 (1.67e−1)**
	1000	5.0377e−1 (2.96e−2) -	6.1591e−1 (2.74e−1) -	3.5548e−1 (5.21e−3) -	5.7531e−1 (1.76e−3) -	8.1962e−1 (3.26e−1) -	**1.5286e−1 (9.61e−2)**
+ / - / =		0/31/5	0/36/0	11/19/6	0/35/1	0/34/2	

(3) The performance of the DLMOEA-DLS is compared with five state-of-the-art LSMOEAs on 90 LSMOPs benchmark instances. The experimental results show that the DLMOEA-DLS achieves superior performance on most benchmark instances.

The rest of the paper is organized as follows. The variants of competitive swarm optimizer (CSO) and differential evolution (DE) are briefly introduced in Section 2. The framework and details of the DLMOEA-DLS are described in Section 3. The experimental results and analysis are presented in Section 4. Finally, the paper is concluded in Section 5.

2. Related work

In this section, some typical variants of the CSO and the DE are introduced first and then the motivations of the DLMOEA-DLS are presented.

2.1. The variants of the CSO

The CSO is an effective swarm intelligence optimization algorithm (Cheng & Jin, 2015; Wang, Zhang, Wang, & Jin, 2021; Zhang, Zheng, Cheng, Qiu, & Jin, 2018), which is different from the particle swarm optimization (PSO) (Lin et al., 2016; Nebro et al., 2009; Ning, Peng, Dai, Bi, & Wang, 2019). The CSO randomly selects two particles for comparison, where the velocity of the particle with worse fitness is updated towards the position of the particle with better fitness. In the CSO, the update of particles has neither global nor historical optimal information. Therefore, the CSO avoids falling into a local optimum in the evolution process. The CSO has gained significant attention due to its suitable properties for solving large-scale optimization problems.

The OBL-CPSO (Zhou, Fang, Wu, Sun, & Cheng, 2016) borrows the same idea as the CSO. It reduces the computing budget, and effectively improves the search range in the decision space. The OBL-CPSO randomly selects three solutions from the population for fitness comparison. The solution with worst fitness value is updated by learning from the solution with best fitness value as shown in (2) and (3), while the other solution is updated through the opposition-based learning mechanism (OBL) at each time.

$$V_l(t+1) = r_1(t)V_l(t) + r_2(t)(X_w(t) - X_l(t)) + \varphi r_3(t)(\overline{X}(t) - X_l(t)) \tag{2}$$

$$X_l(t+1) = X_l(t) + V_l(t+1) \tag{3}$$

where $X_w(t)$ and $X_l(t)$ are the positions of the solution with best fitness value and the solution with worst fitness value in iteration t, respectively, $\overline{X}(t)$ is the mean position of all the solutions in iteration t, $V_l(t)$ is the velocity of the loser in iteration t, $r_1(t)$, $r_2(t)$, and $r_3(t)$ are three random values within $[0,1]$ and φ is a manual parameter.

In the CMOPSO (Zhang, Zheng, et al., 2018), particle learning mechanism and environmental selection operator are executed cyclically during the evolution process. The CMOPSO constructs an elite particle set y by non-dominated sort and density estimation (Zhang, Tian, Cheng, & Jin, 2015) first. Then, it randomly selects a particle p from the original population, two particles a, and b from the elite particle set y, and calculates the acute angles between p and a, p and b. Lastly, the elite particle with a smaller angle to p is regarded as the winner, and p is optimized by learning from the winner.

2.2. The variants of the DE

As a classic intelligent optimization algorithm, the DE exhibits a strong searching ability for the MOPs with complex spaces (Cheng, Pan, Liang, Gao, & Gao, 2021; Li, Shi, Yue, Shang, & Qu, 2019). The original DE mainly performs three operators, i.e., mutation, crossover, and selection. The mutation is an essential operator to generate better solutions. The three commonly used mutation strategies are DE/rand/1, DE/best/1, and DE/current-to-best/1. After that, generated solutions to be updated by the crossover, and then the selection operator will choose the promising solution for the next iteration.

The mutation and crossover play an essential role in the algorithm. Since the control parameters F and CR are randomly generated in the original DE, the robustness of the DE decreases rapidly when tackling the various LSMOPs. Therefore, it is crucial to obtain reasonable values of the parameters F and CR in the DE. Therefore, several variants of the DE with dynamic parameters have been proposed to resolve the problem above mentioned.

The jDE (Brest, Greiner, Boskovic, Mernik, & Zumer, 2006) maintains F and CR for each solution and dynamically reinitializes them in the evolution process. In each iteration, once a random value is less than the preset value τ_1, F is reinitialized to a new random value in the range $[0.1,1]$, otherwise, F is reserved for next iteration. CR is reinitialized with a similar idea with F in $[0,1]$. The dynamic adjustment strategies of F and CR are defined as:

$$F_{i,G+1} = \begin{cases} F_l + r_1 F_u, & \text{if } r_2 < \tau_1 \\ F_{i,G}, & \text{otherwise} \end{cases} \tag{4}$$

$$CR_{i,G+1} = \begin{cases} r_3, & \text{if } r_4 < \tau_2 \\ CR_{i,G}, & \text{otherwise} \end{cases} \tag{5}$$

where τ_1 and τ_2 are the predefined parameters.

The SaDE (Qin, Huang, & Suganthan, 2008) proposes a self-adaptive strategy to dynamically select the generation strategy and the coupled control parameters F and CR. The SaDE adopts a generation selection pool obtained by several generation strategies and maintains an archive for each solution to store the number of successful and unsuccessful offspring which are generated by the generation strategy. At each time, one generation strategy is selected by a probability calculated by the number of successful and unsuccessful offspring. Therefore, one generation strategy with a higher successful number is biased to be selected. The F of each individual is reinitialized by the normal distribution $\mathcal{N}(0.5, 0.3)$ at each time, while CR is $\mathcal{N}(CR_{m_k}, 0.1)$. Meanwhile, the successful values of CR are stored in an archive for each mutation strategy for calculating CR_{m_k}.

In addition to the above-mentioned algorithms, recent variants of the DE with dynamic parameters include the SHADE (Tanabe & Fukunaga, 2013), the SAKPDE (Fan, Wang, & Yan, 2019), and the FDDE (Cheng et al., 2021).

2.3. Motivations

Although various LSMOEAs have been designed to solve the LSMOPs, most of them fail to balance convergence and diversity well. Most of the existing algorithms transform the original problems into low-dimensionality problems. However, they perform sensitively to the problem's dimensionality and variable's characteristics, which may lead to fall into local optimum. The solutions with good convergence may fail to maintain diversity well and vice versa when dealing with high-dimensionality problems. To illustrate this phenomenon, Fig. 1 (a) and (b) plot the distribution of the final nondominated solutions obtained by the LMEA and the LMOCSO on LSMOP9 with 200 decision variables, respectively. It can be observed that the solutions obtained by the LMEA locate in the part of the PF with uneven distribution and the LMOCSO cannot approximate the true PF well. It is reasonable to conclude that multiple generation operators rather than a single operator are expected to cope with the phenomenon mentioned above.

Recently, a multi-stage optimization strategy (MS) has exhibited promising performance and has been employed to address various MOPs (Cao, Zhang, Zhao, & Chen, 2021; Dong, Gong, Ming, & Wang, 2022; Ma, Wei, Tian, Cheng, & Zhang, 2021; Ming et al., 2021; Shen, Wang, Dong, Li, & Wang, 2022). Generally speaking, the algorithms close to the PF in the first stage to ensure convergence, and the diversity is enhanced in the second stage to explore the whole PF. However, it is worth noting that seldom algorithm employs the MS to deal with the LSMOPs. Therefore, a new LSMOEA employs the MS in this paper to solve the LSMOPs, the convergence and the diversity of the population are improved in two sequential stages. In addition, two different reproduction operators are designed to accelerate convergence and to explore unknown areas of PF.

3. Proposed method

3.1. General framework of the DLMOEA-DLS

The DLMOEA-DLS adopts a dual-stage optimization strategy to balance the convergence and the diversity of the population. The first stage accelerates the convergence of the population, and the second stage improves the diversity of the population.

Algorithm 1: General framework of the DLMOEA-DLS

Input: N (population size), Z (optimization problem)
Output: P (final population)
P = Initialization population with size N
F = Random initialization with size N in [0, 1]
$[CV, DV] = VariableClustering(Z)$
While *convergence stage termination criterion not fulfilled* **do**
$P = OptimizationOperator(P, CV, DV)$
end While
While *diversity stage termination criterion not fulfilled* **do**
$[P, F] = DynamicLearningStrategy(P, F)$
end While

The general framework of the DLMOEA-DLS is shown in Algorithm 1. Firstly, the initial population P with size N is randomly generated, and the control parameter F with size N is randomly initialized between 0 and 1. Then the decision variables are divided into convergence-related and diversity-related variables by the variables clustering strategy (Zhang, Tian, et al., 2018), called CV and DV, respectively. After that the optimization operator is performed in the convergence stage until

Table 2
The HV values of compared algorithms on tri-objective LSMOP1-LSMOP9, where the best result is shown in bold font and gray background.

Problem	Dec	DGEA	LMEA	LMOCSO	LMOEA-DS	WOF	DLMOEA-DLS
LSMOP1	100	6.9306e−1 (4.27e−2) -	7.9838e−1 (6.05e−2) -	8.4909e−1 (1.90e−4) =	6.3121e−1 (7.32e−2) -	8.0364e−1 (2.21e−2) -	**8.5003e−1 (7.99e−2)**
	200	5.2449e−1 (4.78e−2) -	8.0822e−1 (2.82e−2) -	**8.4906e−1 (1.25e−4) =**	5.4609e−1 (8.92e−2) -	7.9720e−1 (1.38e−2) -	8.4485e−1 (8.54e−4)
	500	4.8771e−1 (1.18e−2) -	7.9771e−1 (6.37e−2) -	**8.4912e−1 (1.27e−4) =**	4.7209e−1 (4.20e−2) -	7.9028e−1 (1.70e−2) -	8.4629e−1 (4.77e−4)
	1000	4.6756e−1 (5.84e−3) -	6.9055e−1 (2.09e−1) -	**8.4921e−1 (7.63e−5) =**	4.2701e−1 (2.87e−2) -	7.7678e−1 (2.86e−2) -	8.4587e−1 (9.19e−4)
LSMOP2	100	7.8155e−1 (2.29e−2) -	7.9520e−1 (6.51e−2) =	**8.3566e−1 (1.07e−3) +**	7.7831e−1 (3.22e−3) -	7.7454e−1 (2.60e−2) -	8.1582e−1 (4.30e−3)
	200	8.2217e−1 (1.30e−3) =	7.8876e−1 (2.76e−2) -	**8.3863e−1 (1.08e−3) =**	8.0405e−1 (2.68e−3) -	8.1006e−1 (3.45e−3) -	8.2290e−1 (5.26e−3)
	500	8.3751e−1 (7.68e−4) =	8.0237e−1 (5.23e−2) -	**8.4257e−1 (4.89e−4) =**	8.1803e−1 (3.19e−3) -	8.3346e−1 (1.22e−3) -	8.3776e−1 (2.08e−3)
	1000	8.4177e−1 (6.40e−3) -	7.6738e−1 (1.09e−1) -	**8.4540e−1 (1.38e−4) =**	8.2292e−1 (2.95e−3) -	8.4115e−1 (1.09e−3) -	8.4385e−1 (3.58e−4)
LSMOP3	100	2.2682e−1 (2.71e−2) -	3.5889e−1 (1.30e−1) -	4.9033e−1 (9.85e−2) -	1.3831e−1 (8.26e−2) -	4.1391e−1 (1.86e−1) -	**6.6990e−1 (1.13e−1)**
	200	1.9801e−1 (2.02e−2) -	3.5295e−1 (1.02e−1) -	3.8544e−1 (7.75e−2) -	9.2685e−2 (4.05e−3) -	3.5497e−1 (1.81e−1) -	**6.2346e−1 (2.08e−1)**
	500	1.8673e−1 (8.43e−3) -	2.5193e−1 (1.59e−1) -	3.8830e−1 (3.05e−2) -	9.1234e−2 (2.97e−4) -	2.6810e−1 (1.60e−1) -	**6.4947e−1 (7.55e−2)**
	1000	1.7600e−1 (2.41e−2) -	1.0520e−1 (8.39e−2) -	1.8194e−1 (7.52e−2) -	9.1508e−2 (4.77e−4) -	3.2922e−1 (1.51e−1) -	**3.7050e−1 (3.78e−2)**
LSMOP4	100	7.2766e−1 (1.64e−2) -	7.4642e−1 (1.06e−1) -	**8.3919e−1 (1.65e−3) =**	6.3307e−1 (3.28e−2) -	7.4675e−1 (1.38e−2) -	8.1661e−1 (3.19e−3)
	200	7.4041e−1 (6.91e−3) -	7.2551e−1 (1.16e−1) -	**8.4328e−1 (1.24e−3) =**	7.2532e−1 (4.78e−3) -	7.5509e−1 (5.52e−3) -	8.2858e−1 (2.93e−3)
	500	7.9169e−1 (2.23e−3) -	8.0186e−1 (3.30e−2) -	**8.4630e−1 (4.60e−4) =**	7.6663e−1 (3.44e−3) -	7.9647e−1 (4.94e−3) -	8.3556e−1 (1.03e−2)
	1000	8.2018e−1 (5.64e−4) -	8.0819e−1 (1.98e−2) -	**8.4753e−1 (3.07e−4) +**	7.9786e−1 (4.18e−3) -	8.1846e−1 (3.51e−3) -	8.4195e−1 (8.18e−4)
LSMOP5	100	3.8418e−1 (7.05e−2) -	2.7337e−1 (2.46e−1) -	5.6335e−1 (1.22e−4) =	4.0551e−1 (2.26e−3) -	3.9255e−1 (1.83e−2) -	**5.6959e−1 (2.37e−3)**
	200	3.7726e−1 (6.78e−2) -	1.6836e−1 (2.14e−1) -	5.6353e−1 (2.03e−3) =	4.0375e−1 (1.62e−3) -	4.0668e−1 (3.78e−2) -	**5.6931e−1 (1.32e−4)**
	500	3.9019e−1 (1.30e−4) -	1.0198e−1 (1.48e−1) -	4.3941e−1 (7.98e−2) -	4.0229e−1 (2.51e−3) -	4.3684e−1 (5.60e−2) -	**5.6923e−1 (1.08e−4)**
	1000	3.9007e−1 (2.11e−5) -	9.4742e−2 (1.40e−1) -	4.4308e−1 (1.04e−1) -	3.9873e−1 (2.35e−3) -	4.4059e−1 (4.52e−2) -	**5.4529e−1 (1.07e−1)**
LSMOP6	100	4.0217e−2 (3.46e−2) =	6.2346e−3 (2.29e−2) -	3.4084e−2 (4.45e−2) =	2.0611e−2 (7.73e−3) =	**5.5449e−2 (1.24e−2) +**	2.3373e−2 (1.09e−2)
	200	**5.0795e−2 (1.85e−2) =**	0.0000e+0 (0.00e+0) -	1.3085e−2 (2.92e−2) -	8.1820e−3 (7.59e−3) -	2.3675e−3 (1.06e−2) -	4.4943e−2 (1.46e−2)
	500	7.1399e−2 (3.17e−3) -	0.0000e+0 (0.00e+0) -	1.8848e−2 (5.90e−2) -	5.9229e−3 (6.84e−3) -	6.7754e−3 (3.03e−2) -	**8.7633e−2 (3.81e−2)**
	1000	7.5720e−2 (1.74e−2) -	0.0000e+0 (0.00e+0) -	5.8409e−2 (8.91e−2) -	6.7990e−3 (5.61e−3) -	6.2310e−3 (2.79e−2) -	**8.9953e−2 (1.19e−2)**
LSMOP7	100	1.4238e−2 (3.31e−2) -	1.0804e−2 (2.74e−2) -	8.5372e−2 (4.27e−2) -	2.5305e−3 (6.68e−3) -	8.2216e−2 (1.90e−2) -	**9.9633e−2 (2.54e−2)**
	200	2.7569e−2 (3.98e−2) -	2.5192e−2 (6.70e−2) -	**9.0908e−2 (6.06e−7) =**	0.0000e+0 (0.00e+0) -	8.1881e−2 (1.80e−2) =	8.3113e−2 (6.29e−2)
	500	3.9532e−2 (4.23e−2) =	0.0000e+0 (0.00e+0) -	**9.0909e−2 (1.94e−7) +**	0.0000e+0 (0.00e+0) -	7.0158e−2 (2.88e−2) +	2.8043e−2 (5.25e−2)
	1000	6.2915e−2 (2.67e−2) +	1.0894e−2 (4.66e−3) -	**9.0918e−2 (4.02e−5) +**	1.8686e−2 (5.47e−3) -	6.0813e−2 (2.83e−2) +	2.4453e−2 (5.90e−3)
LSMOP8	100	4.4841e−1 (4.52e−2) -	3.1714e−1 (1.06e−1) -	**5.4723e−1 (5.83e−3) +**	3.8998e−1 (1.15e−2) -	4.9937e−1 (3.03e−2) =	4.9150e−1 (2.29e−2)
	200	4.8703e−1 (3.40e−2) -	4.1908e−1 (4.35e−2) -	**5.4325e−1 (2.90e−3) +**	4.2026e−1 (1.80e−2) -	5.1586e−1 (1.25e−2) =	5.1978e−1 (7.84e−3)
	500	5.1385e−1 (3.62e−3) -	4.9883e−1 (1.90e−2) -	**5.4787e−1 (2.72e−3) +**	4.4620e−1 (1.09e−2) -	5.1355e−1 (1.03e−2) -	5.3627e−1 (2.45e−3)
	1000	5.1658e−1 (2.22e−3) -	5.1641e−1 (3.35e−3) -	**5.5649e−1 (2.41e−3) =**	4.5323e−1 (1.04e−2) -	5.0901e−1 (2.64e−2) -	5.5490e−1 (6.04e−3)
LSMOP9	100	7.5696e−2 (3.98e−2) -	1.0565e−1 (2.93e−2) -	1.3774e−1 (8.94e−2) -	1.9218e−1 (5.68e−5) -	1.4203e−1 (1.32e−2) -	**2.3530e−1 (9.14e−3)**
	200	7.9026e−2 (4.10e−2) -	9.8398e−2 (2.71e−2) -	1.5968e−1 (8.05e−2) -	1.9197e−1 (2.26e−4) -	1.4008e−1 (1.63e−2) -	**2.3487e−1 (4.25e−3)**
	500	5.9279e−2 (3.05e−2) -	7.5061e−2 (2.63e−2) -	2.1556e−1 (5.24e−2) -	1.9118e−1 (3.30e−4) -	1.5195e−1 (2.90e−2) -	**2.3915e−1 (1.78e−3)**
	1000	6.1409e−2 (2.94e−2) -	1.1434e−1 (4.61e−2) -	2.3528e−1 (3.35e−2) -	1.9135e−1 (2.72e−4) -	1.7088e−1 (2.97e−2) -	**2.3755e−1 (2.20e−3)**
+ / - / =		1/30/5	0/35/1	7/14/15	0/35/1	4/28/4	

the termination criterion is met. Finally, the population enters the diversity stage, in which the population is updated by the DLS. The details of the optimization operator and the DLS are introduced in Algorithm 2 and Algorithm 3, respectively.

For the dual-stage optimization strategy, a preset parameter named θ is generated between 0 and 1. The number of function evaluations in the convergence stage is defined as (6), in which *maxFE* represents the total number of function evaluations of the DLMOEA-DLS. Therefore, the number of function evaluations in the diversity stage is $(1-\theta) \times maxFE$.

$$convergenceFE = \theta \times maxFE \tag{6}$$

3.2. Optimization operator in the convergence stage

The convergence stage aims to enable the generated population to approximate the true PF rapidly. Based on the various effects of the decision variables to the objective values, the DLMOEA-DLS divides the decision variables into *CV* and *DV*, and optimizes them independently by the optimization operator, in which any generation operator can be employed.

Algorithm 2: $P = OptimizationOperator(P, CV, DV)$

Input: *P* (population), *CV* (convergence-related variables), *DV* (diversity-related variables)
Output: *P* (population)

```
    /* Convergence-related Optimization */
1:  CPop = update the variables in CV for each solution in P
2:  Cchange = fitness(CPop) > fitness(P)
3:  P(Cchange) = CPop(Cchange)
    /* Diversity-related Optimization */
4:  DPop = update the variables in DV for each solution in P
5:  Dchange = fitness(DPop) > fitness(P)
6:  P(Dchange) = DPop(Dchange)
```

The optimization operator is illustrated in Algorithm 2. The convergence stage optimization includes 2 parts: convergence-related and diversity-related optimization. Firstly, the DLMOEA-DLS optimizes the current population by updating the decision variables in *CV* to generate new

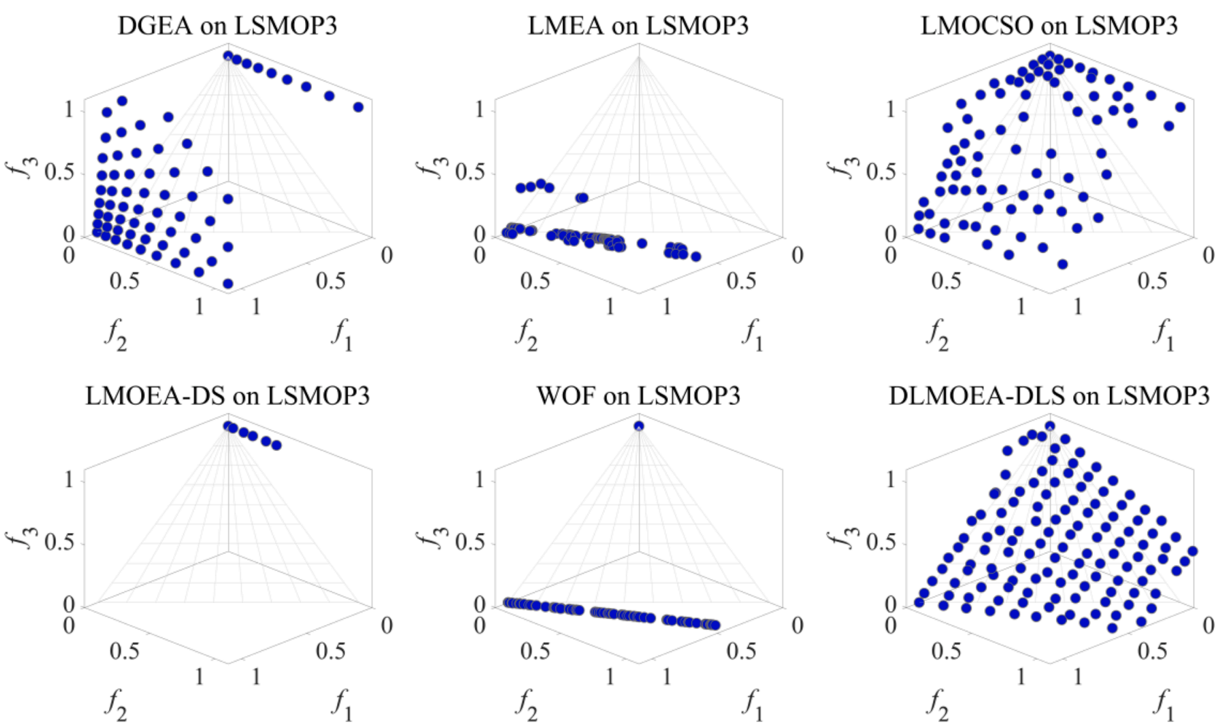

Fig. 2. The final population obtained by compared algorithms on LSMOP3 with 200 decision variables.

population *CPop*, then calculates the fitness value of *CPop* using the shift-based density estimation (SDE) strategy (Li, Yang, & Liu, 2014), and reserves the solutions with better fitness. Secondly, the decision variables in *DV* are updated using the same approach, resulting in the population *P*. The diversity-related optimization assists the generated population to avoid falling into a local optimum.

The fitness of solution *p* is defined as the minimum SDE-based distance between the solution and others in the current population, as shown in (7):

$$fitness(p) = \min_{q \in P-(p)} \sqrt{\sum_{i=1}^{M} (\max\{0, f_i(q) - f_i(p)\})^2} \tag{7}$$

where M is the number of the objectives and $f_i(p)$ is the i th objective value of p.

3.3. Dynamic learning strategy in the diversity stage

In the diversity stage, the DLS is designed to improve the diversity of the population. Specifically, solutions are devised to learn from leaders with better fitness values, and the coupled control parameters are dynamically adjusted according to the historical behaviors.

The detail of the DLS is presented in Algorithm 3. Firstly, two solutions are randomly selected and compared at each time. The solution with better fitness is called x_w and the other one is called x_l. Moreover, x_w as the leader guides x_l to generate x_o as follow:

$$x_o(t+1) = r_1 x_l(t) + r_2(x_w(t) - x_l(t)) + F(x_w(t) - x_r(t)) \tag{8}$$

where r_1 and r_2 are the parameters generated in range $[0,1]$, F is control parameter in range $[0,1]$ and $x_r(t)$ is a random solution in the iteration t. The DLS repeats the above loop until there is no solution in the current population P. Lastly, the environmental selection operator (Tian, Cheng, Zhang, Cheng, & Jin, 2017; Tian, Zhang, Cheng, & Jin, 2016) is performed to maintain N solutions for next iteration from x_w, x_l, and x_o.

Algorithm 3: $[P, F] = DynamicLearningStrategy(P, F)$

```
Input: P (population), F(dynamic parameter)
Output: P (population), F (dynamic parameter)
Offspringset=∅
While |P|⟩1 do
    [p, q] = randomly select two solutions from P and remove them
    [x_w, x_l] = compare p and q using (7)
    x_o = update the x_l by learning from x_w by (8)
    Offspringset = Offspringset+ (x_w, x_l, x_o)
    if fitness(x_o) < fitness(x_l)
        update the F by (9)
    end if
end While
P = EnviromentalSelection(Offspringset, N)
```

The setting of the control parameter F is crucial for generating promising offspring. An unreasonable value of F may result in x_l learning less information from a random solution, thus the diversity of generated offspring will become worse. The control parameter F of each solution is different from others. This encourages a diverse search and maintains the diversity of the population. In the DLS, F of x_l will be reinitialized when the fitness of generated x_o is worse than x_l, otherwise, F will be reserved for next iteration.

$$F_l(t+1) = \begin{cases} F_l(t), & fitness(x_o) > fitness(x_l) \\ rand(0,1), & otherwise \end{cases} \tag{9}$$

where $F_l(t)$ is the control parameter F of x_l in iteration t.

By employing the above strategies, solutions tend to have better convergence and diversity. Thus, the final population will be able to approximate and distribute PF well.

3.4. Computational complexity analysis

The DLMOEA-DLS involves four main parts, i.e., variables clustering strategy, optimization operator, DLS, and environmental selection. Assuming that the population size is N, the numbers of decision vari-

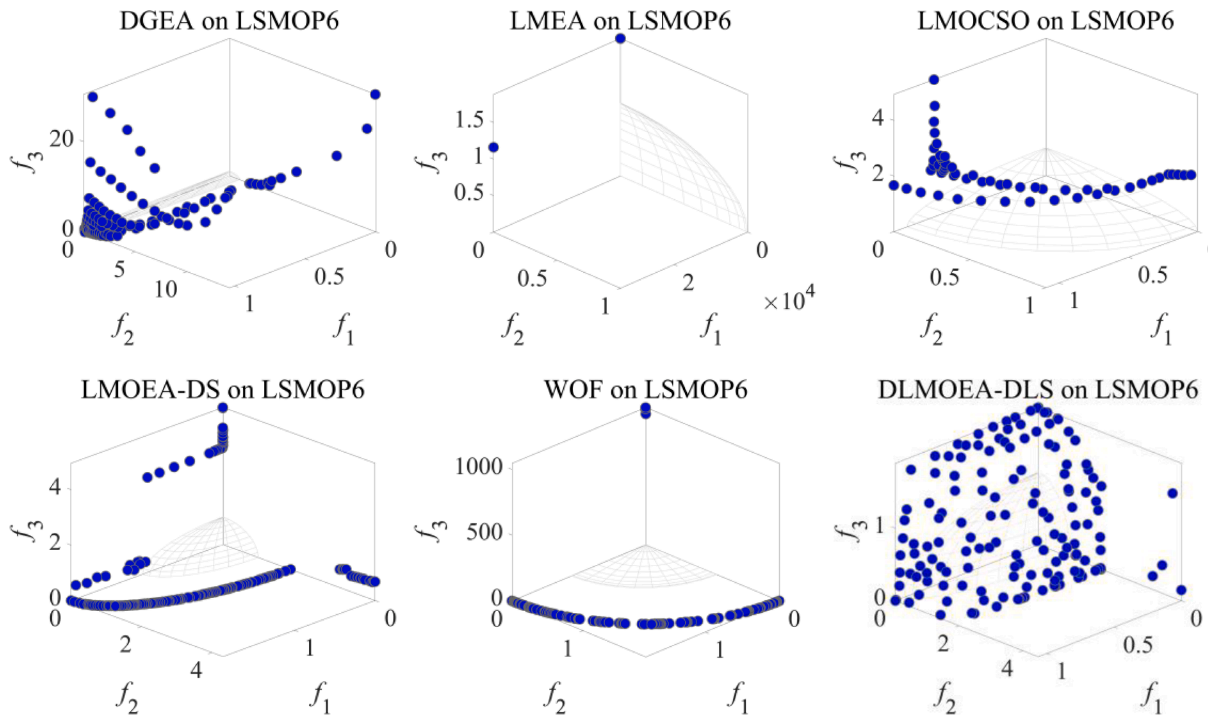

Fig. 3. The final population obtained by compared algorithms on LSMOP6 with 500 decision variables.

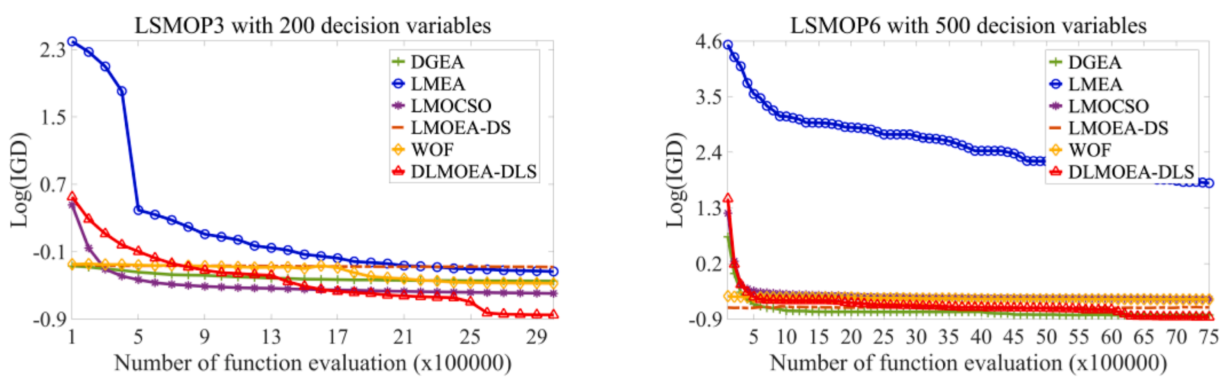

Fig. 4. The logarithm of the IGD values obtained by compared algorithms.

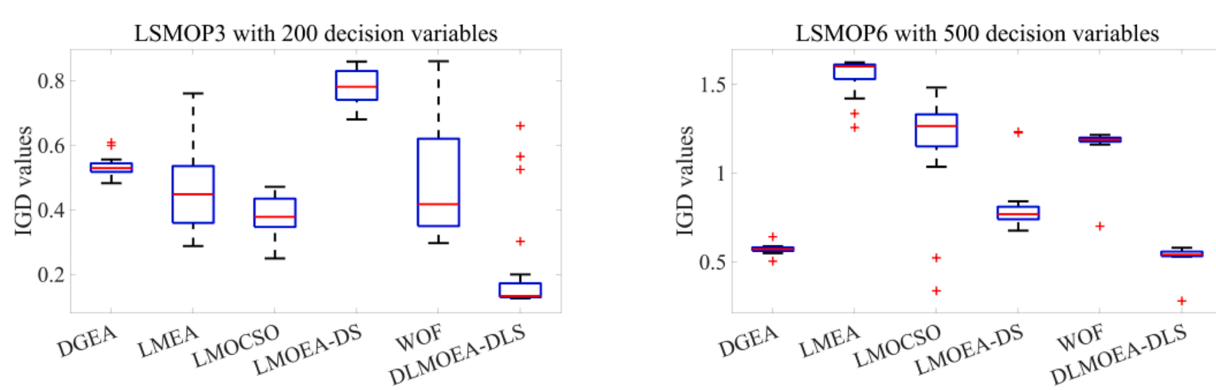

Fig. 5. The boxplots of the IGD values obtained by compared algorithms.

ables and objectives are D and M, respectively. In the variables clustering strategy, the time complexity is $O(nqD + Dlog_D)$, where n is the number of selected solutions and q is the time of decision variables analysis. In the optimization operator, the time complexity is $O(ND)$. In the DLS, N solutions are compared and updated, and F are adjusted at each iteration. Thus, the time complexity is $2 \times 1/2O(N) + 3/2O(N) =$

Table 3
The IGD values of compared algorithms on tri-objective LMF1-LMF12, where the best result is shown in bold font and gray background.

Problem	Dec	DGEA	LMEA	LMOCSO	LMOEA-DS	WOF	DLMOEA-DLS
LMF1	100	**2.6178e−1 (6.02e−3) +**	2.0492e+0 (1.51e+0) -	3.2507e−1 (4.95e−2) =	3.1536e−1 (2.67e−2) =	9.0715e−1 (1.02e−1) -	4.2365e−1 (1.70e−1)
	200	**2.6739e−1 (9.51e−3) +**	1.4791e+1 (1.11e+1) -	3.3351e−1 (6.39e−2) +	5.8937e−1 (1.02e−1) =	8.1373e−1 (1.49e−1) -	7.0680e−1 (1.72e−1)
	500	**2.7385e−1 (7.01e−3) +**	1.0955e+2 (8.50e+1) -	3.9613e−1 (4.85e−2) +	6.6189e−1 (6.59e−3) =	8.5634e−1 (1.52e−1) =	8.0663e−1 (1.74e−1)
	1000	**2.9060e−1 (3.26e−2) +**	5.0223e+2 (2.82e+2) -	1.2805e+1 (3.75e+0) -	6.6485e−1 (4.38e−3) +	9.2522e−1 (9.26e−2) -	8.4086e−1 (1.63e−1)
LMF2	100	1.4207e−1 (1.82e−3) -	3.6187e−1 (2.48e−1) -	2.4628e−1 (1.66e−3) -	2.3555e−1 (9.94e−3) -	2.4152e−1 (3.11e−2) -	**1.0277e−1 (6.67e−2)**
	200	1.6464e−2 (2.34e−3) =	7.1827e−1 (4.01e−1) -	2.4088e−1 (2.40e−2) -	2.3717e−1 (8.91e−4) -	2.3859e−1 (8.62e−3) -	**1.6173e−1 (8.24e−2)**
	500	**1.7398e−2 (7.88e−2) =**	1.4214e+0 (3.79e−1) -	2.2384e−1 (4.62e−2) -	2.3700e−1 (8.79e−4) -	2.3662e−1 (7.27e−3) -	2.0281e−1 (6.17e−1)
	1000	**1.5892e−1 (8.44e−2) =**	1.9845e+0 (8.21e−1) -	3.3450e−1 (1.28e−1) -	2.3732e−1 (9.26e−4) -	2.3636e−1 (6.88e−3) -	2.2811e−1 (2.92e−2)
LMF3	100	1.6400e−1 (2.03e−2) -	8.3939e−1 (6.21e−1) -	8.6072e−1 (1.14e−16) -	6.8722e−1 (1.07e−1) -	3.8799e−1 (3.14e−2) -	**1.5309e−1 (1.76e−2)**
	200	**2.1534e−1 (2.15e−2) =**	3.5372e+0 (2.96e+0) -	8.6072e−1 (1.14e−16) -	2.3467e+0 (5.10e−1) -	5.4142e−1 (7.61e−2) -	3.0926e−1 (5.66e−2)
	500	**3.1905e−1 (2.44e−2) =**	2.1240e+1 (2.64e+1) -	8.6072e−1 (1.14e−16) +	9.6704e+1 (9.73e+1) -	8.3028e−1 (6.29e−2) +	3.9020e−2 (5.83e−1)
	1000	**4.3114e−1 (2.93e−2) =**	7.7260e+1 (1.20e+2) -	3.0894e+0 (7.84e−1) -	1.0711e+3 (3.16e+2) -	8.6072e−1 (1.14e−16) -	4.4877e−1 (2.12e−2)
LMF4	100	3.1643e−1 (3.12e−2) =	7.8627e−1 (5.59e−1) -	**2.7882e−1 (7.80e−2) =**	4.5197e−1 (4.24e−2) -	5.5430e−1 (2.66e−2) -	3.1270e−1 (5.09e−2)
	200	4.6397e−1 (8.24e−2) -	2.4530e+0 (1.63e+0) -	4.7743e−1 (4.38e−2) -	4.2911e−1 (1.93e−2) -	6.2909e−1 (1.77e−2) -	**4.1874e−1 (3.36e−2)**
	500	5.8706e−1 (1.39e−2) -	1.0915e+1 (5.01e+0) -	6.3803e−1 (7.90e−2) -	5.4633e−1 (2.64e−2) -	7.5772e−1 (8.14e−2) -	**4.8583e−1 (1.47e−2)**
	1000	6.1021e−1 (1.28e−2) -	2.5594e+1 (1.07e+1) -	6.5122e−1 (6.98e−2) -	7.2114e−1 (7.32e−3) -	7.4030e−1 (7.70e−2) -	**4.8028e−1 (1.65e−2)**
LMF5	100	7.2077e−2 (3.39e−2) -	2.1771e−1 (1.79e−1) -	6.7372e−2 (6.59e−2) -	1.1670e−1 (1.29e−2) -	1.9557e−1 (4.36e−2) -	**3.9558e−2 (2.49e−3)**
	200	9.2594e−2 (3.56e−2) -	2.8864e−1 (2.69e−1) -	6.8337e−2 (4.60e−2) -	1.2068e−1 (3.24e−2) -	2.0143e−1 (3.04e−2) -	**4.3344e−2 (2.26e−3)**
	500	8.7678e−1 (7.55e−2) -	3.5792e+0 (4.18e+0) -	4.2566e−1 (3.68e−3) -	4.6818e−1 (9.49e−2) -	8.7678e−1 (0.00e+0) -	**3.8235e−1 (5.33e−2)**
	1000	**1.3196e−1 (1.12e−1) =**	3.2998e+1 (2.27e+1) -	3.6117e−1 (9.12e−2) -	4.1878e−1 (2.02e−1) -	2.6715e−1 (1.48e−1) -	1.9637e−1 (2.85e−2)
LMF6	100	1.9382e−1 (1.40e−2) =	4.1406e−1 (1.27e−1) -	**4.4700e−2 (2.74e−3) +**	2.3399e−1 (1.27e−2) =	4.1770e−1 (4.75e−2) -	2.1620e−1 (5.71e−2)
	200	**1.6347e−1 (4.54e−2) +**	6.3027e−1 (1.57e−1) -	1.6792e−1 (8.90e−2) +	2.7620e−1 (1.57e−2) +	4.7913e−1 (3.61e−2) -	4.0397e−1 (5.39e−2)
	500	**7.8461e−1 (8.01e−2) +**	1.8639e+0 (7.95e−1) -	8.6072e−1 (3.51e−6) =	8.4719e−1 (1.20e−2) =	8.4195e−1 (3.09e−2) =	8.6072e−1 (7.02e−2)
	1000	5.0306e−1 (5.14e−2) +	4.3498e+0 (4.09e+0) -	8.5574e−1 (4.73e−3) -	**4.7148e−1 (5.79e−3) +**	6.8941e−1 (1.06e−1) +	8.2404e−1 (3.23e−2)
LMF7	100	3.7813e−1 (3.29e−2) -	1.1718e+0 (8.01e−1) -	4.7900e−1 (6.11e−2) -	4.5627e−1 (4.48e−2) -	7.3261e−1 (1.43e−1) -	**3.4411e−1 (3.48e−2)**
	200	5.9519e−1 (4.55e−2) -	4.6548e+0 (4.34e+0) -	4.6612e−1 (2.67e−2) -	5.4224e−1 (4.24e−2) -	7.2748e−1 (1.07e−1) -	**4.4574e−1 (6.34e−3)**
	500	7.0628e−1 (7.73e−2) -	3.0000e+1 (2.28e+1) -	6.6978e−1 (9.06e−2) -	7.8051e−1 (8.81e−2) -	6.8691e−1 (8.30e−2) -	**4.6459e−1 (3.07e−3)**
	1000	7.5557e−1 (1.05e−1) -	9.8792e−1 (0.00e+0) -	6.8679e−1 (9.16e−2) -	7.9054e−1 (2.12e−2) -	7.0348e−1 (3.79e−2) -	**4.7402e−1 (1.61e−2)**
LMF8	100	**3.2856e−1 (7.72e−2) +**	9.9490e−1 (4.30e−1) -	6.8332e−1 (9.91e−2) =	5.0789e−1 (3.58e−2) +	8.4551e−1 (5.00e−2) -	6.4606e−1 (1.01e−1)
	200	**4.0127e−1 (3.62e−2) +**	1.1922e+0 (6.67e−1) -	7.7001e−1 (3.75e−2) =	5.6769e−1 (2.49e−2) +	8.4939e−1 (5.11e−2) -	7.8260e−1 (5.45e−2)
	500	7.3567e−1 (8.28e−2) =	1.2890e−1 (4.57e−1) -	9.1581e−1 (1.05e−2) -	**6.5403e−1 (3.25e−2) +**	8.7477e−1 (5.22e−2) -	7.3519e−1 (3.66e−2)
	1000	8.6468e−1 (9.02e−2) =	9.8017e−1 (1.14e−16) -	9.4401e−1 (1.75e−2) -	**7.1858e−1 (1.65e−2) +**	8.6558e−1 (1.71e−2) =	8.7209e−1 (6.98e−2)
LMF9	100	**3.2856e−1 (3.57e−2) +**	1.2657e+1 (1.11e+1) -	5.8676e−1 (3.01e−2) +	8.0213e−1 (1.64e−1) =	9.3496e−1 (2.96e−2) -	7.9023e−1 (8.92e−2)
	200	**5.1069e−1 (5.68e−2) +**	7.1474e+1 (9.47e+1) -	6.0970e−1 (3.59e−2) +	9.0012e−1 (3.25e−1) -	9.4278e−1 (9.68e−3) -	8.8862e−1 (5.48e−2)
	500	6.5478e−1 (4.62e−2) +	5.8740e+2 (8.08e+2) -	**6.3116e−1 (3.68e−2) +**	7.9824e−1 (1.21e−1) +	9.3977e−1 (2.75e−2) =	9.4592e−1 (1.03e−2)
	1000	6.8666e−1 (6.36e−2) +	2.6714e+3 (3.08e+3) -	**6.5340e−1 (4.40e−2) +**	7.1207e−1 (3.13e−2) +	9.3698e−1 (3.21e−2) =	9.3560e−1 (3.31e−2)
LMF10	100	8.5839e−1 (4.21e−3) -	1.2101e+1 (1.50e+1) -	8.6036e−1 (1.25e−4) -	8.6013e−1 (1.13e−3) -	8.6056e−1 (4.54e−4) -	**7.8162e−1 (1.20e−1)**
	200	8.6073e−1 (2.47e−3) -	1.1952e+2 (8.50e+1) -	8.6047e−1 (1.80e−4) -	8.6053e−1 (7.67e−4) -	8.5752e−1 (1.36e−2) -	**7.2541e−1 (1.47e−1)**
	500	8.5989e−1 (1.12e−4) -	7.8857e+2 (5.19e+2) -	8.6071e−1 (1.03e−5) -	8.6072e−1 (6.32e−4) -	8.6070e−1 (6.02e−5) -	**6.4789e−1 (1.01e−1)**
	1000	9.1397e−1 (7.00e−2) -	2.0444e+3 (2.54e+3) -	2.2719e+2 (3.33e+1) -	8.6025e−1 (2.10e−3) -	8.6070e−1 (7.28e−5) -	**6.1155e−1 (5.47e−2)**
LMF11	100	5.1391e−1 (7.25e−2) -	1.2250e+0 (5.40e−1) -	5.6892e−1 (1.76e−2) -	7.1611e−1 (5.65e−2) -	7.9790e−1 (1.07e−1) -	**4.8273e−1 (2.45e−2)**
	200	5.0581e−1 (1.67e−2) -	1.5782e+0 (8.80e−1) -	5.6334e−1 (1.81e−2) -	7.5441e−1 (4.89e−2) -	8.7464e−1 (7.26e−2) -	**4.5849e−1 (1.87e−1)**
	500	5.2637e−1 (1.25e−2) -	2.9328e+0 (1.09e+0) -	5.6179e−1 (1.67e−2) -	7.2833e−1 (2.07e−2) -	9.3145e−1 (3.45e−2) -	**4.4030e−1 (6.77e−4)**
	1000	5.9682e−1 (1.33e−2) -	9.4592e−1 (1.14e−5) -	5.5988e−1 (2.33e−2) -	7.0108e−1 (2.20e−2) -	7.8579e−1 (1.38e−1) -	**5.4001e−1 (1.26e−3)**
LMF12	100	**7.6564e−1 (9.22e−2) +**	2.8716e+0 (2.06e+0) -	7.8261e−1 (5.17e−2) +	9.8815e−1 (8.21e−3) -	1.0003e+0 (2.42e−2) -	9.4479e−1 (2.22e−1)
	200	8.7009e−1 (1.13e−1) +	1.0103e+1 (1.61e+1) -	**8.0020e−1 (5.59e−2) +**	1.0420e+0 (1.17e−3) =	1.0442e+0 (2.13e−2) =	1.0554e+0 (1.74e−2)
	500	**8.6238e−1 (3.59e−2) +**	1.2910e+2 (2.30e+2) -	9.8792e−1 (3.63e−2) +	1.0594e+0 (3.83e−2) =	1.0582e+0 (2.14e−2) =	1.0744e+0 (9.20e−1)
	1000	**8.6230e−1 (1.33e−2) +**	1.0806e+0 (2.28e+1) -	1.0458e+0 (2.08e−2) =	1.0139e+0 (9.06e−2) =	1.0127e+0 (7.61e−2) =	1.0797e+0 (1.87e−2)
+/-/=		17/20/11	0/48/0	12/30/6	9/30/9	2/38/8	

$O(N)$. In the environmental selection, the time complexity is $O(N^2 + MR)$, in which R is the number of reference points. Overall, the computational complexity of the DLMOEA-DLS is $O(N^2 + MR + ND)$.

4. Experimental studies

In this section, several experiments are conducted to evaluate the performance of the DLMOEA-DLS. First, the DLMOEA-DLS is compared with five state-of-the-art LSMOEAs on a set of LSMOPs benchmarks.

Table 4
The HV values of compared algorithms on tri-objective LMF1-LMF12, where the best result is shown in bold font and gray background.

Problem	Dec	DGEA	LMEA	LMOCSO	LMOEA-DS	WOF	DLMOEA-DLS
LMF1	100	**3.1923e−1 (5.00e−3) +**	8.3473e−3 (1.74e−2) -	2.7533e−1 (3.49e−2) =	2.5805e−1 (1.42e−2) -	1.0494e−1 (4.00e−2) -	2.8796e−1 (5.59e−2)
	200	**3.1576e−1 (5.91e−3) +**	0.0000e+0 (0.00e+0) -	2.6760e−1 (4.30e−2) +	2.0037e−1 (2.19e−3) =	1.4213e−1 (6.63e−2) =	1.9523e−1 (8.85e−2)
	500	**3.1189e−1 (3.89e−3) +**	0.0000e+0 (0.00e+0) -	2.3408e−1 (3.90e−2) +	2.0021e−1 (4.40e−4) =	1.2647e−1 (6.84e−2) =	1.5865e−1 (9.48e−2)
	1000	**3.0217e−1 (2.20e−2) +**	0.0000e+0 (0.00e+0) -	0.0000e+0 (0.00e+0) -	2.0015e−1 (5.89e−4) +	1.0054e−1 (4.31e−2) -	1.3932e−1 (8.30e−2)
LMF2	100	8.7369e−1 (3.02e−2) -	5.8432e−1 (2.90e−1) -	8.3746e−1 (5.41e−3) -	8.4905e−1 (2.80e−3) -	8.1553e−1 (4.41e−2) -	**9.5408e−1 (4.22e−3)**
	200	9.4885e−1 (4.16e−3) =	2.8101e−1 (3.71e−1) -	8.3827e−1 (5.83e−3) -	8.4876e−1 (1.94e−3) -	8.4134e−1 (7.74e−3) -	**9.4973e−1 (3.73e−2)**
	500	9.1102e−1 (3.96e−2) -	3.8148e−2 (8.17e−2) -	8.4377e−1 (1.36e−2) -	8.4874e−1 (1.76e−3) -	8.4386e−1 (6.05e−3) -	**9.6835e−1 (1.35e−2)**
	1000	**8.7116e−1 (3.36e−2) =**	5.6820e−2 (1.99e−1) -	6.0550e−1 (1.15e−1) -	8.4636e−1 (2.51e−3) -	8.4594e−1 (6.06e−3) -	8.5478e−1 (4.60e−3)
LMF3	100	**6.4997e−1 (2.55e−2) =**	2.1384e−1 (1.85e−1) -	9.0909e−2 (4.27e−17) -	9.9350e−2 (3.72e−2) -	3.5941e−1 (3.69e−2) -	6.4043e−1 (2.66e−2)
	200	**5.8028e−1 (2.87e−2) =**	6.2958e−5 (2.82e−4) -	9.0909e−2 (4.27e−17) -	0.0000e+0 (0.00e+0) -	2.4382e−1 (5.32e−2) -	5.5509e−1 (5.60e−2)
	500	**4.5110e−1 (2.61e−2) +**	1.7810e−3 (7.96e−3) =	9.0909e−2 (4.27e−17) +	0.0000e+0 (0.00e+0) =	1.0247e−1 (2.44e−2) +	0.0000e+0 (0.00e+0)
	1000	**3.3942e−1 (2.68e−2) =**	0.0000e+0 (0.00e+0) -	0.0000e+0 (0.00e+0) -	0.0000e+0 (0.00e+0) -	9.0909e−2 (4.27e−17) -	3.0823e−1 (1.77e−2)
LMF4	100	2.0052e−1 (2.27e−2) -	1.8483e−1 (1.40e−1) -	**2.5742e−1 (5.94e−2) =**	8.6805e−2 (2.45e−2) -	1.1074e−1 (8.30e−3) -	2.5194e−1 (5.60e−2)
	200	1.3060e−1 (1.83e−2) -	7.2407e−2 (1.29e−1) -	1.3071e−1 (4.11e−3) -	1.2628e−1 (1.07e−2) -	8.9477e−2 (6.83e−3) -	**1.5912e−1 (7.45e−2)**
	500	1.0342e−1 (1.99e−3) -	4.5579e−2 (1.11e−1) -	9.5234e−2 (2.60e−3) -	8.4433e−2 (7.86e−3) -	6.8049e−2 (1.65e−2) -	**1.1127e−1 (3.34e−3)**
	1000	1.0009e−1 (2.70e−3) -	2.7037e−2 (8.32e−2) -	1.0600e−1 (7.48e−4) -	1.0541e−1 (7.28e−4) -	8.1943e−2 (2.22e−2) -	**1.0682e−1 (1.24e−3)**
LMF5	100	9.1267e−1 (3.37e−2) -	7.4658e−1 (2.02e−1) -	9.3572e−1 (4.21e−2) -	8.7371e−1 (7.55e−3) -	7.8045e−1 (6.11e−2) -	**9.5036e−1 (5.71e−3)**
	200	8.9037e−1 (3.50e−2) -	6.7771e−1 (2.71e−1) -	9.2251e−1 (3.34e−2) -	8.7192e−1 (2.08e−2) -	8.2006e−1 (3.65e−2) -	**9.4473e−1 (4.14e−3)**
	500	2.3014e−1 (6.69e−2) -	1.8504e−1 (3.28e−1) -	4.1060e−1 (3.01e−2) -	4.9666e−1 (1.36e−1) -	9.0907e−2 (4.27e−17) -	**6.8285e−1 (6.43e−3)**
	1000	8.4146e−1 (7.67e−3) -	1.6308e−1 (2.92e−1) -	6.7028e−1 (1.09e−1) -	5.3558e−1 (1.96e−1) -	7.2680e−1 (1.62e−1) -	**8.4319e−1 (3.28e−2)**
LMF6	100	7.1730e−1 (1.80e−2) +	3.3655e−1 (9.94e−2) -	**8.0046e−1 (7.40e−3) +**	5.8826e−1 (1.83e−2) =	3.6410e−1 (3.58e−2) -	5.6344e−1 (6.72e−2)
	200	**6.4915e−1 (5.54e−2) +**	1.6030e−1 (9.65e−2) -	6.2310e−1 (1.18e−1) +	5.1304e−1 (1.61e−2) +	3.3530e−1 (2.61e−2) -	3.7019e−1 (4.38e−2)
	500	**1.1815e−1 (3.14e−2) =**	1.6676e−3 (7.36e−3) -	9.0909e−2 (1.47e−6) -	9.1304e−2 (4.36e−4) -	9.0911e−2 (2.76e−6) -	1.1103e−2 (7.85e−2)
	1000	**2.7210e−1 (6.33e−2) +**	5.0000e−2 (4.64e−2) -	9.1314e−2 (1.37e−4) -	2.6793e−1 (5.44e−3) +	1.7016e−1 (6.37e−2) =	9.5332e−2 (6.48e−3)
LMF7	100	1.6674e−1 (8.27e−2) -	1.2165e−1 (1.39e−1) -	1.3972e−1 (2.11e−2) -	9.2405e−2 (2.72e−2) -	4.5905e−2 (3.61e−2) -	**2.0880e−1 (1.76e−2)**
	200	9.5077e−2 (2.26e−3) -	5.0648e−2 (9.58e−2) -	1.2098e−1 (6.43e−2) =	5.1096e−2 (2.39e−2) -	5.1757e−2 (3.09e−2) -	**1.3257e−1 (1.68e−2)**
	500	7.0141e−2 (2.97e−2) -	1.9349e−2 (5.95e−2) -	7.1269e−2 (1.84e−2) -	5.6596e−3 (1.70e−2) -	7.5543e−2 (2.48e−2) -	**8.6637e−2 (8.34e−4)**
	1000	5.4347e−2 (3.43e−2) -	0.0000e+0 (0.00e+0) -	5.3732e−2 (1.10e−2) -	4.6508e−2 (2.38e−3) -	8.1830e−2 (1.52e−2) -	**8.2866e−2 (3.67e−3)**
LMF8	100	**4.3840e−1 (7.92e−2) +**	8.6041e−2 (9.50e−2) -	1.2842e−1 (4.61e−2) -	2.2584e−1 (4.22e−2) +	3.7406e−2 (2.17e−2) -	1.6990e−1 (7.59e−2)
	200	**3.5679e−1 (3.59e−2) +**	6.1850e−2 (6.33e−2) =	5.4498e−2 (1.28e−2) -	1.6684e−1 (2.18e−2) +	3.8247e−2 (2.25e−2) -	8.4143e−2 (4.62e−2)
	500	6.9269e−2 (6.00e−2) +	0.0000e+0 (0.00e+0) =	0.0000e+0 (0.00e+0) =	**6.9319e−2 (2.80e−2) +**	1.9046e−2 (1.56e−2) +	0.0000e+0 (0.00e+0)
	1000	1.8607e−2 (2.94e−2) +	0.0000e+0 (0.00e+0) =	0.0000e+0 (0.00e+0) =	**2.3667e−2 (8.04e−3) +**	1.6645e−2 (7.92e−3) +	0.0000e+0 (0.00e+0)
LMF9	100	**1.9775e−1 (3.08e−2) +**	0.0000e+0 (0.00e+0) -	6.8540e−2 (3.99e−3) +	1.2773e−2 (1.33e−2) -	9.4070e−3 (4.36e−3) -	2.5877e−2 (1.43e−2)
	200	**7.7045e−2 (3.14e−2) +**	0.0000e+0 (0.00e+0) -	6.4487e−2 (4.47e−3) +	1.5189e−2 (1.35e−2) =	8.7641e−3 (1.54e−3) -	1.7139e−2 (8.13e−3)
	500	2.2574e−2 (9.75e−3) +	0.0000e+0 (0.00e+0) -	**6.2108e−2 (5.69e−3) +**	2.2060e−2 (1.83e−2) +	9.2100e−3 (4.23e−3) =	9.2643e−3 (1.63e−2)
	1000	4.7733e−2 (2.10e−2) +	3.3057e−3 (4.15e−3) -	**5.8849e−2 (6.60e−3) +**	4.0539e−2 (1.32e−2) +	9.7634e−3 (5.44e−3) -	9.8647e−3 (5.09e−3)
LMF10	100	8.2883e−3 (5.31e−5) -	4.1891e−3 (4.29e−3) -	8.2665e−3 (2.38e−7) -	8.2481e−3 (5.95e−4) -	8.2733e−3 (3.45e−5) -	**8.3073e−3 (7.85e−5)**
	200	8.1028e−3 (4.20e−4) -	2.4889e−3 (3.90e−3) -	8.2655e−3 (1.87e−7) -	8.2760e−3 (3.52e−5) -	8.2701e−3 (9.33e−6) -	**8.3223e−3 (7.75e−5)**
	500	8.2898e−3 (2.44e−5) -	1.6515e−3 (3.39e−3) -	8.2643e−3 (4.89e−8) -	8.2643e−3 (3.50e−6) -	8.2643e−3 (1.12e−7) -	**8.3083e−3 (4.76e−6)**
	1000	8.2849e−3 (2.29e−5) -	4.1244e−3 (4.23e−3) -	0.0000e+0 (0.00e+0) -	8.2653e−3 (4.50e−6) -	8.2643e−3 (4.66e−8) -	**8.9817e−3 (3.68e−3)**
LMF11	100	7.0259e−2 (3.97e−3) -	1.5846e−2 (2.68e−2) -	7.0998e−2 (1.97e−3) =	1.1947e−2 (8.16e−3) -	2.9192e−2 (1.72e−2) -	**8.2180e−2 (2.03e−2)**
	200	6.6639e−2 (1.46e−3) -	5.3195e−2 (1.20e−1) -	7.1245e−2 (2.06e−3) -	8.3737e−3 (4.89e−4) -	1.7686e−2 (1.19e−2) -	**1.5762e−1 (1.41e−1)**
	500	6.0305e−2 (1.17e−3) -	2.5329e−2 (7.73e−2) -	3.0566e−2 (9.36e−3) -	9.5024e−3 (2.92e−4) -	8.9481e−3 (2.30e−3) -	**6.9716e−2 (3.15e−3)**
	1000	6.5637e−2 (1.01e−3) -	8.2643e−3 (1.78e−18) -	9.6066e−3 (2.85e−4) -	3.0714e−2 (8.68e−3) -	2.8615e−2 (1.77e−2) -	**6.8539e−2 (5.19e−3) +**
LMF12	100	0.0000e+0 (0.00e+0) -	0.0000e+0 (0.00e+0) -	0.0000e+0 (0.00e+0) -	0.0000e+0 (0.00e+0) -	0.0000e+0 (0.00e+0) -	**3.4597e−3 (9.77e−3)**
	200	0.0000e+0 (0.00e+0) =	0.0000e+0 (0.00e+0) =	0.0000e+0 (0.00e+0) =	0.0000e+0 (0.00e+0) =	0.0000e+0 (0.00e+0) =	0.0000e+0 (0.00e+0)
	500	0.0000e+0 (0.00e+0) =	0.0000e+0 (0.00e+0) =	0.0000e+0 (0.00e+0) =	0.0000e+0 (0.00e+0) =	0.0000e+0 (0.00e+0) =	0.0000e+0 (0.00e+0)
	1000	0.0000e+0 (0.00e+0) =	0.0000e+0 (0.00e+0) =	0.0000e+0 (0.00e+0) =	0.0000e+0 (0.00e+0) =	0.0000e+0 (0.00e+0) =	0.0000e+0 (0.00e+0)
+/-/=		16/23/9	0/41/7	9/30/9	9/31/8	3/38/7	

Then, the validity of θ setting is experimentally verified. Afterwards, the performance of the DLMOEA-DLS on different termination criterions is analyzed. Finally, the effectiveness of each component in the DLMOEA-DLS is verified, and several experiments are implemented to verify the effectiveness of the DLS. All the experiments were performed on the PlatEMO (Tian, Cheng, Zhang, & Jin, 2017) coded by MATLAB.

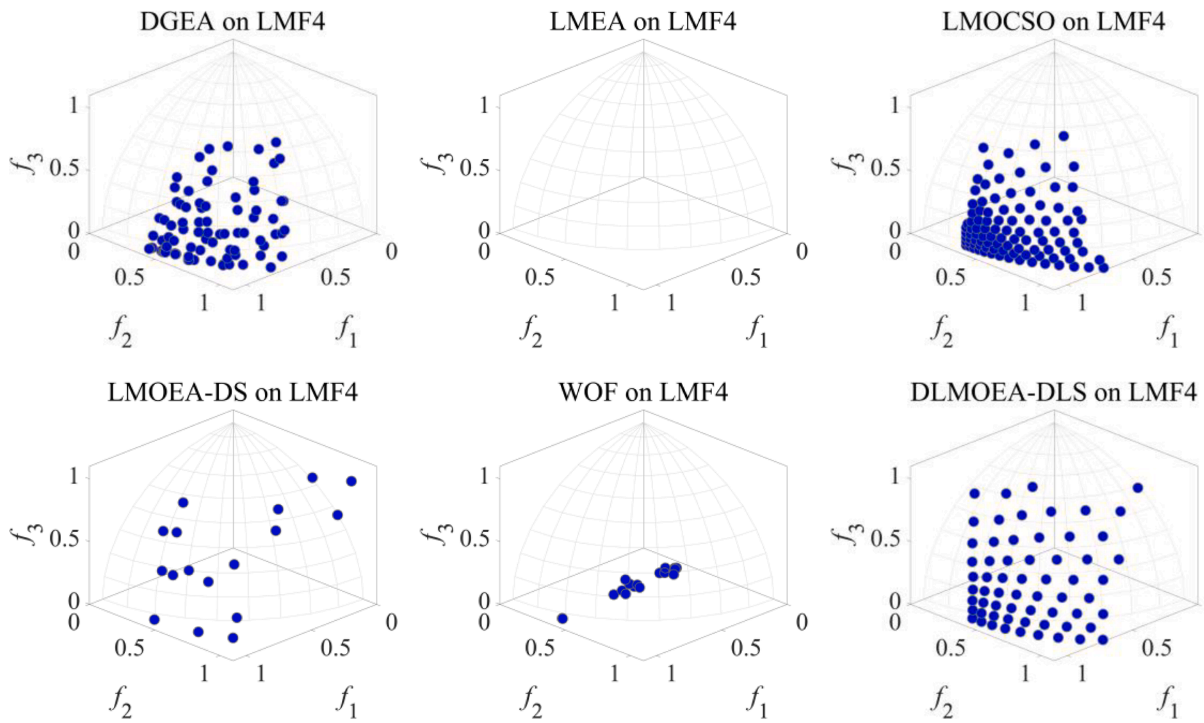

Fig. 6. The final population obtained by compared algorithms on LMF4 with 200 decision variables.

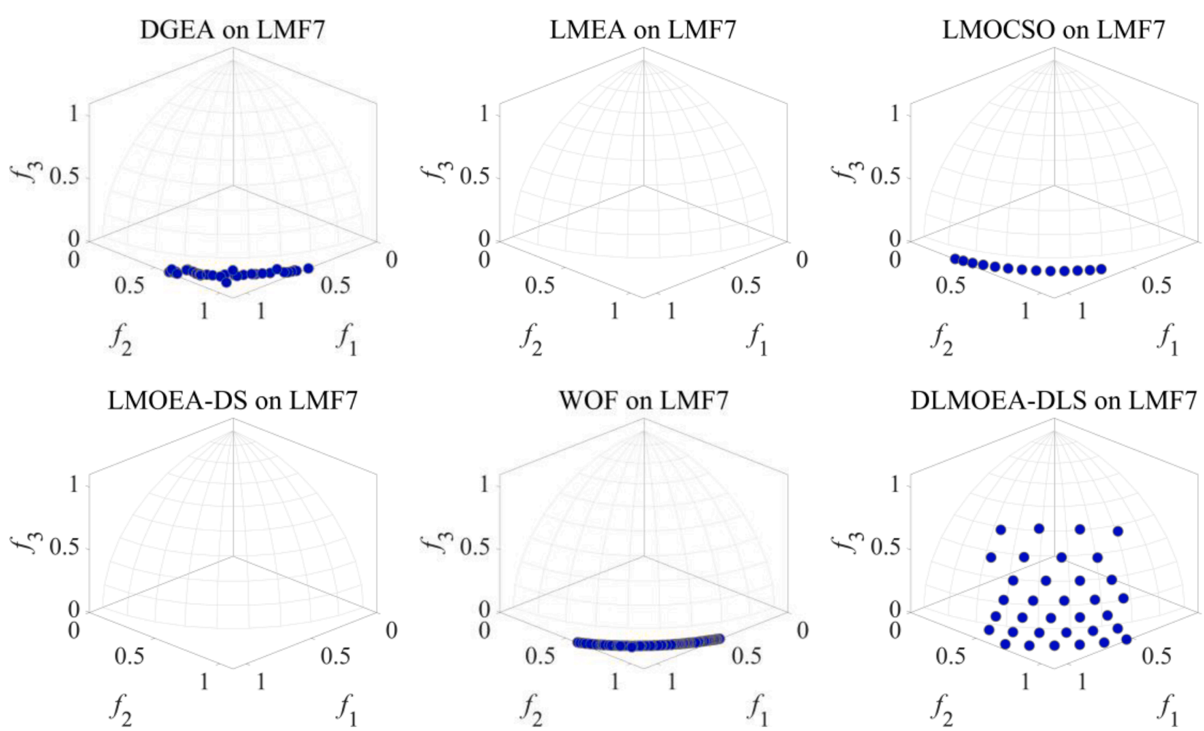

Fig. 7. The final population obtained by compared algorithms on LMF7 with 500 decision variables.

4.1. Compared algorithms

In order to demonstrate the performance of the DLMOEA-DLS, five state-of-the-art LSMOEAs are employed for comparison including the DGEA (He et al., 2022), the LMEA (Zhang, Tian, et al., 2018), the LMOCSO (Tian et al., 2020), the LMOEA-DS (Qin, Sun, Jin, Tan, & Fieldsend, 2021) and the WOF (Zille et al., 2018). The DGEA adopts a novel preselection strategy for parent population maintenance and an adaptive offspring reproduction method. The LMEA is based on decision variables clustering tailored for the LSMOPs. The LMOCSO uses the ideal of the CSO for updating particles in objective space and shows promising potential in addressing the LSMOPs. The LMOEA-DS is based on problem reformulation and the solutions are generated by direct sampling. The WOF is based on problem transformation, which employs grouping strategies and transformation functions.

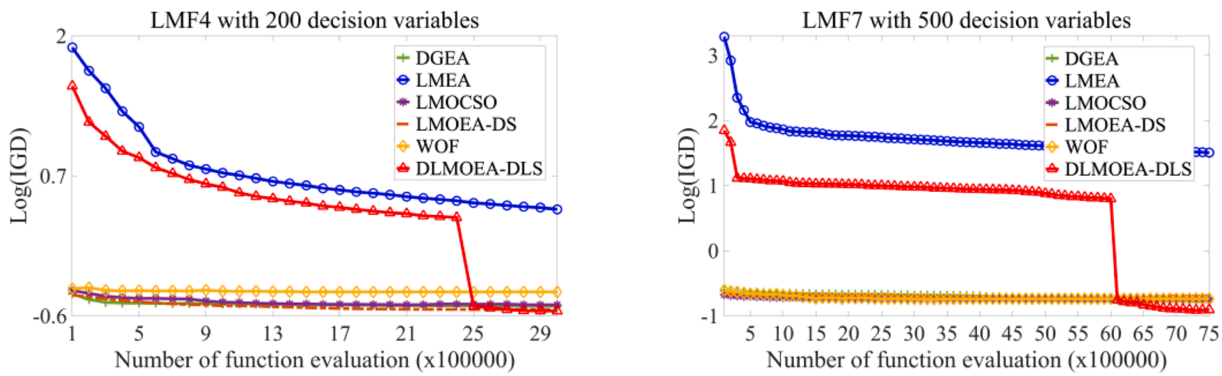

Fig. 8. The logarithm of the IGD values on LMF4 with 200 decision variables and LMF7 with 500 decision variables.

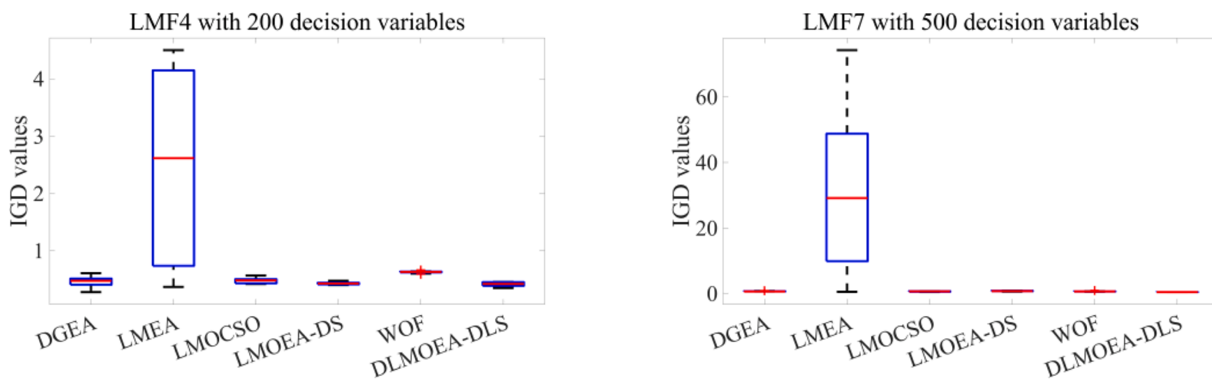

Fig. 9. The boxplots of the IGD values obtained by compared algorithms.

Table 5
The IGD values of compared algorithms on TREE1-TREE6, where the best result is shown in bold font and gray background.

Problem	M	Dec	DGEA	LMEA	LMOCSO	LMOEA-DS	WOF	DLMOEA-DLS
TREE1	2	300	9.8708e+0 (1.24e−4) -	9.8654e+0 (2.45e−2) -	9.8710e+0 (8.32e−5) -	7.6543e+0 (2.49e−1) -	8.5396e+0 (3.16e+0) -	**5.7247e+0 (5.61e−2)**
TREE2	2	300	**3.0174e+1 (1.95e+1)** =	4.4070e+1 (1.82e−1) -	4.4171e+1 (2.36e−5) -	4.3549e+1 (1.11e+0) -	4.4169e+1 (4.81e−4) -	4.3269e+1 (2.41e−2)
TREE3	2	600	5.4224e+0 (6.10e+0) -	1.2005e+1 (2.27e−7) -	1.2005e+1 (1.47e−5) -	4.8244e+0 (6.01e+0) -	1.0569e+1 (2.93e+0) -	**2.5417e−1 (6.40e−1)**
TREE4	2	600	1.2000e+1 (5.33e−5) -	1.2000e+1 (6.48e−5) -	1.2000e+1 (3.00e−5) -	1.1336e+1 (5.55e−1) =	1.2000e+1 (8.02e−5) -	**1.0963e+1 (2.42e−1)**
TREE5	2	600	1.7100e+1 (4.25e+0) -	1.8613e+1 (6.36e−5) -	1.8613e+1 (3.74e−5) -	1.7802e+1 (7.36e−1) -	1.8129e+1 (1.88e+0) -	**6.4984e+0 (1.46e+0)**
TREE6	3	1200	NaN (NaN)	NaN (NaN)	NaN (NaN)	NaN (NaN)	NaN (NaN)	NaN (NaN)
+/-/=			0/4/1	0/5/0	0/5/0	0/4/1	0/5/0	

Table 6
The HV values of compared algorithms on TREE1-TREE6, where the best result is shown in bold font and gray background.

Problem	M	Dec	DGEA	LMEA	LMOCSO	LMOEA-DS	WOF	DLMOEA-DLS
TREE1	2	300	8.5048e−1 (4.04e−5) -	8.4452e−1 (2.76e−2) -	8.5056e−1 (4.79e−5) -	8.3024e−1 (2.92e−4) -	8.5053e−1 (2.12e−4) -	**8.5108e−1 (2.04e−4)**
TREE2	2	300	8.5426e−1 (2.02e−4) -	8.3894e−1 (3.05e−2) -	8.5594e−1 (1.58e−5) -	8.5237e−1 (7.30e−5) -	8.5581e−1 (6.05e−5) -	**8.5608e−1 (3.73e−5)**
TREE3	2	600	8.8224e−1 (4.25e−3) -	8.8721e−1 (3.89e−5) -	8.8396e−1 (3.57e−4) -	8.8742e−1 (4.13e−5) -	8.8507e−1 (1.36e−3) -	**8.8766e−1 (6.95e−5)**
TREE4	2	600	9.5023e−1 (7.69e−4) -	9.3612e−1 (5.85e−3) -	9.4050e−1 (9.13e−4) -	**9.5077e−1 (6.36e−5)** +	9.5024e−1 (1.91e−4) -	9.5070e−1 (2.02e−5)
TREE5	2	600	9.3864e−1 (1.45e−4) -	9.3886e−1 (2.53e−5) -	9.3859e−1 (1.32e−5) -	9.2916e−1 (6.90e−5) -	9.3879e−1 (2.26e−4) -	**9.4000e−1 (1.22e−4)**
TREE6	3	1200	NaN (NaN)	NaN (NaN)	NaN (NaN)	NaN (NaN)	NaN (NaN)	NaN (NaN)
+/-/=			0/5/0	0/5/0	0/5/0	1/4/0	0/5/0	

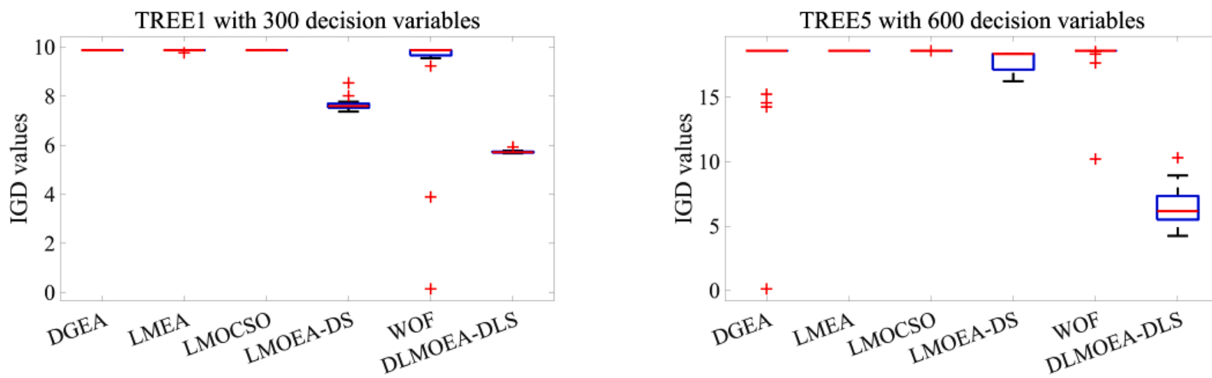

Fig. 10. The boxplots of the IGD values obtained by compared algorithms.

Table 7
The Wilcoxon rank sum (+ / - / =) for the IGD values on tri-objective LSMOP1-LSMOP9.

Dec	D-Theta2 vs DLMOEA-DLS	D-Theta4 vs DLMOEA-DLS	D-Theta6 vs DLMOEA-DLS	D-Theta7 vs DLMOEA-DLS	D-Theta9 vs DLMOEA-DLS
100	0/8/1	0/6/3	0/3/6	0/5/4	0/7/2
200	0/7/2	0/7/2	0/6/3	0/4/5	0/6/3
500	0/7/2	0/7/2	0/6/3	1/6/2	0/7/2

Table 8
The Wilcoxon rank sum (+ / - / =) for IGD values on tri-objective LSMOP1-LSMOP9 with 1500 * *D*.

Dec	DGEA vs DLMOEA-DLS	LMEA vs DLMOEA-DLS	LMOCSO vs DLMOEA-DLS	LMOEA-DS vs DLMOEA-DLS	WOF vs DLMOEA-DLS
100	3/3/3	2/7/0	7/0/2	6/1/2	6/2/1
200	9/0/0	1/1/7	7/0/2	9/0/0	7/0/2
500	9/0/0	1/2/6	7/1/1	9/0/0	8/0/1
1000	9/0/0	0/0/9	9/0/0	9/0/0	9/0/0

Table 9
The Wilcoxon rank sum (+ / - / =) for IGD values on tri-objective LSMOP1-LSMOP9 with 5000 * *D*.

Dec	DGEA vs DLMOEA-DLS	LMEA vs DLMOEA-DLS	LMOCSO vs DLMOEA-DLS	LMOEA-DS vs DLMOEA-DLS	WOF vs DLMOEA-DLS
100	2/5/2	0/7/2	7/1/1	2/4/3	1/6/2
200	2/4/3	0/9/0	4/2/3	3/5/1	2/6/1
500	3/3/3	0/9/0	4/3/2	0/5/4	2/6/1
1000	8/0/1	1/5/3	5/2/2	6/2/1	7/1/1

Table 10
The Wilcoxon rank sum (+ / - / =) for IGD values on tri-objective LSMOP1-LSMOP9 with 15,000 * *D*.

Dec	DGEA vs DLMOEA-DLS	LMEA vs DLMOEA-DLS	LMOCSO vs DLMOEA-DLS	LMOEA-DS vs DLMOEA-DLS	WOF vs DLMOEA-DLS
100	0/8/1	0/9/0	3/4/2	0/9/0	0/8/1
200	0/7/2	0/9/0	3/5/1	0/9/0	0/9/0
500	0/7/2	0/9/0	3/5/1	0/8/1	0/8/1
1000	0/9/0	0/9/0	2/5/2	0/9/0	0/9/0

4.2. Test problems

The LSMOP benchmark (Cheng, Jin, & Olhofer, 2017) is designed for large-scale multi-objective optimization test specifically. Based on the large-scale single-objective optimization problems, the LSMOP benchmark guarantees the generality and extensibility of the LSMOPs. Among the nine instances in the LSMOP benchmark, LSMOP1-LSMOP4 are the linear variables linkage and LSMOP5-LSMOP9 are the nonlinear variables linkage on the PS (Cheng et al., 2017; Tian et al., 2020). In addition, LSMOP1 and LSMOP5 are separable of the objective function, LSMOP2 and LSMOP6 are partially separable, and others are mixed (Tian et al., 2022). The LMF benchmark (Liu, Lin, Wong, Li, & Tan, 2021) inherits the common principles of the LSMOP benchmark, and makes five improvements (e.g., a hybrid formulation model, hybrid linkages between position and distance variables, the flexible grouping of variables, strengthening the shape functions, and imbalanced contribution to the objective function). A real-world LSMOPs benchmark is proposed, namely the time-varying ratio error estimation (TREE) problem (He et al., 2020), which is the first real-world benchmark designed for the LSMOPs.

4.3. Parameter settings

For fair comparison, the parameters of all compared algorithms were set according to their original papers. Specifically, the number of reference vectors in the DGEA for offspring generation was set as 10. The number of selected solutions for decision variables clustering and decision variable interaction analysis was set to 5 and the number of perturbations on each solution for decision variable clustering was set to 50 in the LMEA. In the LMOEA-DS, the numbers of clustering and random samplings along each guiding direction were set to 10 and 30, respectively. SMPSO, ordered grouping strategy and interval transformation function were employed in the WOF, the numbers of evaluation for original problem optimization and transformed problem optimization were set to 1000 and 500, respectively, the numbers of variable groups and chosen solutions were both set to 4. In the DLMOEA-DLS, the θ for controlling the dual-stage optimization was set to 0.8. All the statistical data from the compared algorithms were performed 20 times independently.

For the LSMOP benchmark, i.e., LSMOP1-LSMOP9, the number of subcomponents in each variables group was set to 5. For the LMF benchmark, i.e., LMF1-LMF12, mixed formulation model, mixed variables linkage, nonuniform distance-related variables, and unbalanced contribution of variables were adopted. The population size of each compared algorithm was set to 153, and the number of objectives was set to 2 (i.e., TREE1-TREE5) or 3 (i.e., LSMOP1-LSMOP9, LMF1-LMF12, and TREE6). The termination criterion was set to 15,000 * *D* function evaluations, in which *D* is the number of decision variables.

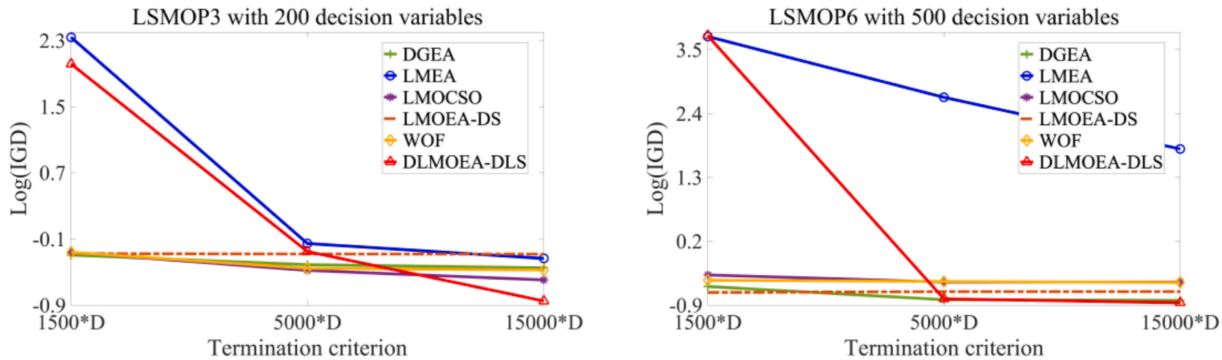

Fig. 11. The logarithm of the IGD values obtained by the compared algorithms with different termination criterions.

4.4. Performance metrics

The two well-known inverted generational distance (IGD) (Czyżżak & Jaszkiewicz, 1998) and hypervolume (HV) (Zitzler & Thiele, 1998) are adopted as the metrics to reflect the results of the comparison. The IGD is designed to calculate the average distance from the true PF to the generated solutions, while the HV computes the volume of the dominated objective space by the final solutions. In addition, to obtain the IGD value, it requires the true PF of specific problem; for the HV, it will consume huge computing resource when the dimensionality of problems become larger.

The formulas of the IGD and the HV are defined as (10) and (11), respectively.

$$IGD(P^*,\Omega)=\frac{\sum_{x\in P^*}dis(x,\Omega)}{|P^*|}, \tag{10}$$

where Ω is the nondominated set, $dis(x,\Omega)$ is the Euclidean distance from x to the solution in Ω, and P^* is a set of evenly reference points.

$$HV=\delta\left(\bigcup_{i=1}^{|S|}v_i\right), \tag{11}$$

where δ represents the Lebesgue measure, which is used to measure the volume, $|S|$ is the number of solutions in the nondominated set, and v_i is the hypervolume formed by the reference point and the i th solution in the nondominated set. It can be seen that a smaller IGD value and a larger HV value represent a better result.

In the experiments, a reference set were sampled on the true PF of each benchmark instance. For the LSMOP and the LMF benchmarks in this study, all the number of objectives was three, and roughly 10,000 reference points for tri-objective problems were evenly sampled from the true PF by the reference point sampling method (Tian, Xiang, Zhang, Cheng, & Jin, 2018) to calculate the IGD values. Meanwhile, the reference point was set to (1.1, 1.1, 1.1) for tri-objective problems and the solutions were normalized to calculate the HV values, in which different reference point would result in different HV values. For the TREE benchmark, since the true PF is unknown, all the nondominated solutions obtained by the compared algorithms in 20 independent runs were collected into an archive, which was as the reference set to calculate the IGD values. Then, the reference point was set to 1.1 times the normalized nadir objective for HV values (He et al., 2020).

A Wilcoxon rank sum (Alcalá-Fdez et al., 2009) test with a 0.05 significance level is adopted to show the statistically significant analysis of the experimental results. The symbols "+", "-" and "=" indicate that the compared algorithms are significantly better than, worse than and similar to the DLMOEA-DLS for solving the LSMOPs.

4.5. Experimental results on the LSMOP benchmark

The IGD and the HV results of the compared algorithms on LSMOP1-LSMOP9 with 100, 200, 500 and 1000 decision variables are shown in Tables 1 and 2, respectively. It can be observed from the tables that the DLMOEA-DLS obtains better results than the other five algorithms. Specifically, compared with the DGEA, the LMEA, the LMOCSO, the LMOEA-DS and the WOF, the proportion of benchmark instances that the DLMOEA-DLS performs better are 31/36, 36/36, 19/36, 35/36 and 34/36, respectively.

Furthermore, it can be concluded that the DLMOEA-DLS performs better than the other five algorithms on the LSMOP benchmark with nonlinear variables linkage. Compared with the LMOCSO, the DLMOEA-DLS can achieve promising results on most of the LSMOP benchmark instances. Specifically, the DLMOEA-DLS achieves better performance on LSMOP3, and LSMOP5-LSMOP7 and LSMOP9. To illustrate the results vividly, the final nondominated population are plotted in Fig. 2 with the mean IGD values obtained by the six algorithms on LSMOP3 with 200 decision variables. Meanwhile, Fig. 3 shows the final nondominated population on LSMOP6 with 500 decision variables. It can be observed from the figures that all the compared algorithms fall to obtain promising solutions with excellent convergence and diversity. However, the DLMOEA-DLS achieves better performance than the other algorithms, where the solutions converge to and explore the entire true PF. It can be demonstrated the superiority of the DLMOEA-DLS on most the LSMOP benchmark instances.

The trends of the logarithm of the mean IGD values obtained by the six algorithms on LSMOP3 with 200 decision variables and LSMOP6 with 500 decision variables are shown in Fig. 4. When the DLMOEA-DLS evolves to the later stage, the IGD value of the DLMOEA-DLS rapidly declines. The reason is that the DLS is executed in the diversity stage. When the population stops converging in the convergence stage, the population generated by the DLS distributes along the PF, which has significantly better diversity. The boxplots of the mean IGD values obtained by the compared algorithms are shown in Fig. 5, in which the extreme outlier point of the LMEA was deleted in order to make the figures intuitive. The DLMOEA-DLS achieves more stable results on most of the LSMOP benchmark instances. It indicates that the DLMOEA-DLS exhibits better robustness.

4.6. Experimental results on the LMF benchmark

The mean IGD and HV values obtained by the compared algorithms on the LMF benchmark are reported in Tables 3 and 4, respectively. Some HV values are zero in Table 4, which means that the compared algorithms fail to generate any nondominated solution in the final population. The tables show that the DLMOEA-DLS could obtain better results than other five state-of-the-art algorithms on most of the LMF benchmark instances. Statistically, the DLMOEA-DLS obtains the best results in 21 of 48 benchmark instances, whereas the DGEA, the LMEA, the LMOCSO, the LMOEA-DS and the WOF perform best in 19, 0, 5, 3, and 0 instances, respectively.

To visually observe the performance, the final nondominated

Table 11
The IGD values of compared algorithms on tri-objective LSMOP1-LSMOP9, where the best result is shown in bold font and gray background.

Problem	Dec	DLMOEA-DLS/A	DLMOEA-DLS/B	DLMOEA-DLS/C	DLMOEA-DLS
LSMOP1	100	3.5695e−1 (3.87e−2) -	3.5930e−2 (7.30e−3) =	4.8180e−2 (1.97e−2) -	**3.3457e−2 (4.54e−2)**
	200	3.7161e−1 (8.51e−2) -	3.2926e−2 (3.35e−4) =	6.7352e−2 (5.43e−2) -	**3.2912e−2 (4.47e−4)**
	500	3.8829e−1 (6.61e−2) -	3.2323e−2 (3.52e−4) -	5.0527e−2 (2.86e−2) -	**3.2086e−2 (2.04e−4)**
LSMOP2	100	7.3415e−2 (4.60e−3) -	6.8488e−2 (9.65e−3) -	8.0414e−2 (3.01e−2) -	**5.6152e−2 (3.91e−3)**
	200	4.9710e−2 (2.03e−3) =	4.9997e−2 (3.62e−3) =	6.9381e−2 (1.34e−2) -	**4.9379e−2 (4.57e−3)**
	500	**3.6650e−2 (9.20e−4) =**	3.6681e−2 (1.42e−3) =	4.9125e−2 (2.33e−3) -	3.6766e−2 (1.40e−3)
LSMOP3	100	8.5228e−1 (8.62e−3) -	**1.6289e−1 (6.70e−2) =**	1.9304e−1 (1.25e−1) =	1.7476e−1 (8.84e−2)
	200	8.5446e−1 (6.97e−3) -	2.1700e−1 (1.53e−1) =	**2.0683e−1 (1.36e−1) =**	2.1254e−1 (1.67e−1)
	500	8.5433e−1 (6.95e−3) -	3.0884e−1 (2.07e−1) -	3.2752e−1 (2.12e−1) -	**1.9359e−1 (5.87e−2)**
LSMOP4	100	2.5573e−1 (1.19e−2) -	5.4871e−2 (1.90e−3) =	5.5432e−2 (7.47e−3) =	**5.4239e−2 (2.45e−3)**
	200	1.5662e−1 (2.13e−3) -	**4.3795e−2 (1.82e−3) =**	8.0477e−2 (6.67e−2) -	4.4120e−2 (2.25e−3)
	500	8.0053e−2 (1.07e−3) -	**3.8121e−2 (6.57e−3) =**	5.5176e−2 (2.33e−2) -	3.8994e−2 (8.87e−3)
LSMOP5	100	4.8140e−1 (1.16e−1) -	1.2860e−1 (1.05e−1) -	1.8108e−1 (1.22e−1) -	**4.3102e−2 (9.57e−2)**
	200	5.5396e−1 (1.32e−1) -	2.4867e−1 (1.77e−1) -	2.9499e−1 (1.77e−1) -	**4.0986e−2 (1.29e−5)**
	500	5.7038e−1 (1.23e−1) -	4.2992e−1 (1.35e−1) -	4.5738e−1 (2.20e−1) -	**4.0984e−2 (9.25e−6)**
LSMOP6	100	7.5334e−1 (3.91e−2) -	5.4616e−1 (5.96e−2) =	8.1850e−1 (1.86e−1) -	**5.3930e−1 (5.84e−2)**
	200	7.7009e−1 (4.58e−2) -	6.0521e−1 (5.02e−2) -	1.0833e+0 (1.56e−1) -	**5.6746e−1 (4.06e−2)**
	500	7.9088e−1 (2.49e−2) -	5.9558e−1 (4.72e−2) -	1.1574e+0 (2.13e−1) -	**5.2040e−1 (8.34e−2)**
LSMOP7	100	1.1249e+0 (1.08e−1) -	1.0160e+0 (2.20e−1) -	1.1393e+0 (3.80e−1) -	**6.0518e−1 (1.15e−1)**
	200	1.0264e+0 (7.80e−2) -	9.4348e−1 (1.62e−1) -	1.2966e+0 (3.18e−1) -	**5.5410e−1 (1.89e−1)**
	500	9.3089e−1 (5.71e−2) -	7.4595e−1 (7.96e−2) -	8.3038e−1 (1.22e−1) -	**6.5371e−1 (1.69e−1)**
LSMOP8	100	1.7191e−1 (2.63e−2) -	1.0941e−1 (2.49e−2) =	1.3187e−1 (3.53e−2) -	**9.6931e−2 (1.90e−2)**
	200	1.1190e−1 (2.33e−3) -	7.3083e−2 (6.99e−3) -	8.5298e−2 (1.41e−2) -	**6.1129e−2 (4.68e−3)**
	500	8.3060e−2 (1.03e−3) -	**5.1883e−2 (1.31e−3) +**	5.6877e−2 (3.98e−3) -	5.3716e−2 (2.66e−3)
LSMOP9	100	1.4001e+0 (1.92e−1) -	5.4799e−1 (2.23e−1) -	5.9351e−1 (2.44e−1) -	**3.5200e−1 (7.00e−2)**
	200	1.3038e+0 (1.86e−1) -	5.5569e−1 (3.49e−1) -	5.3938e−1 (2.68e−1) -	**3.6078e−1 (1.08e−2)**
	500	1.1920e+0 (1.38e−1) -	5.2381e−1 (3.22e−1) -	4.1407e−1 (3.24e−2) -	**2.2569e−1 (1.67e−1)**
+/-/=		0/25/2	1/15/11	0/24/3	

population are plotted in Figs. 6 and 7 obtained by the compared algorithms on LMF4 with 200 decision variables and LMF7 with 500 decision variables. It can be seen that no algorithm achieves promising results with good convergence and diversity on the LMF benchmark. However, the DLMOEA-DLS obtains the best results on LMF4 with 200 decision variables than the other compared algorithms. Meanwhile, the DLMOEA-DLS also achieves a better performance on diversity than the WOF on LMF7 with 500 decision variables.

The evolutionary curves of the logarithm of the IGD values obtained by the compared algorithms on LMF4 with 200 decision variables and LMF7 with 500 decision variables are plotted in Fig. 8. The IGD value obtained by the DLMOEA-DLS decreases quickly in the later stage of the evolution process, moreover, the final population obtained by the DLMOEA-DLS is slightly better than the other algorithms. The reason for this phenomenon may be that the DLS performs global search to explore more part of the PF in the diversity stage. The boxplots of the compared algorithms are shown in Fig. 9. It can be seen from the figure that the DLMOEA-DLS can obtain better mean IGD values and smaller interquartile ranges of the IGD values than the other algorithms. As a consequence, the DLMOEA-DLS performs robustly on most of the LMF benchmark instances.

4.7. Experimental results on the TREE benchmark

All the compared algorithms were performed on the TREE benchmark to investigate the effectiveness of the DLMOEA-DLS for the real-world LSMOPs. The TREE benchmark instances can be classified into three types. Specifically, the first type (TREE1, TREE2 and TREE3) includes two objectives, three constraints, and data with primary voltage only; the second type (TREE4 and TREE5) includes two objectives, four constraints and data with both primary and secondary voltage values; the third type (TREE6) includes three objectives, six constraints and data with both voltage and phase angle values.

Table 5 shows the mean IGD values obtained by the compared algorithms, and Table 6 shows the mean HV values. The IGD and the HV values on TREE6 are NaN, the reason is that the compared algorithms fail to obtain feasible and nondominated solution. It can be seen that the DLMOEA-DLS obtains the best results than other compared algorithms on most instances. To be more specific, the DLMOEA-DLS attains the best performance on TREE1-TREE5. Fig. 10 plots the boxplots of compared algorithms on TREE1 and TREE5. The DLMOEA-DLS can obtain the smaller median IGD values and interquartile ranges than the other algorithms. Therefore, it is reasonable to conclude that the DLMOEA-DLS also exhibits significantly competitive performance on the TREE benchmark.

4.8. The sensitivity analysis of θ

In the DLMOEA-DLS, θ is an important threshold for dividing the entire evolution process into dual-stage optimization. To determine the number of termination criteria for each stage optimization, a series of experiments were performed on tri-objective LSMOP1-LSMOP9 with

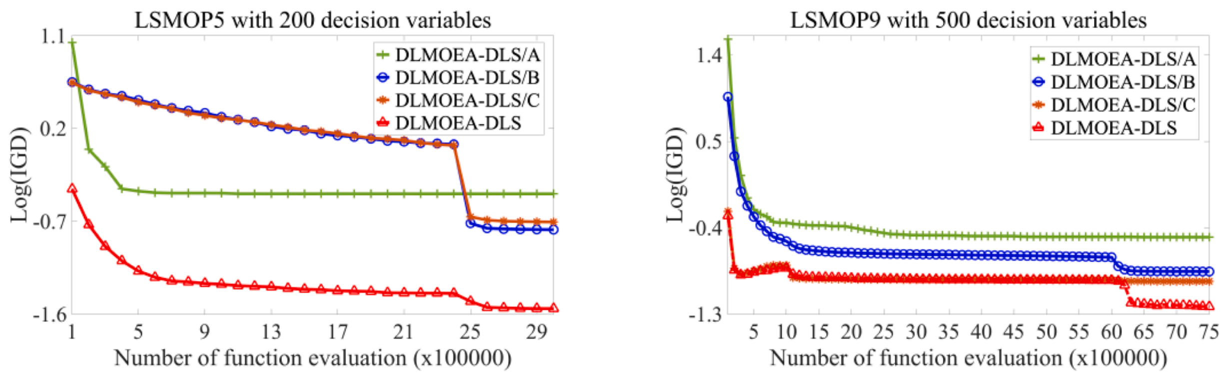

Fig. 12. The logarithm of the IGD values on LSMOP5 with 200 decision variables and LSMOP9 with 500 decision variables.

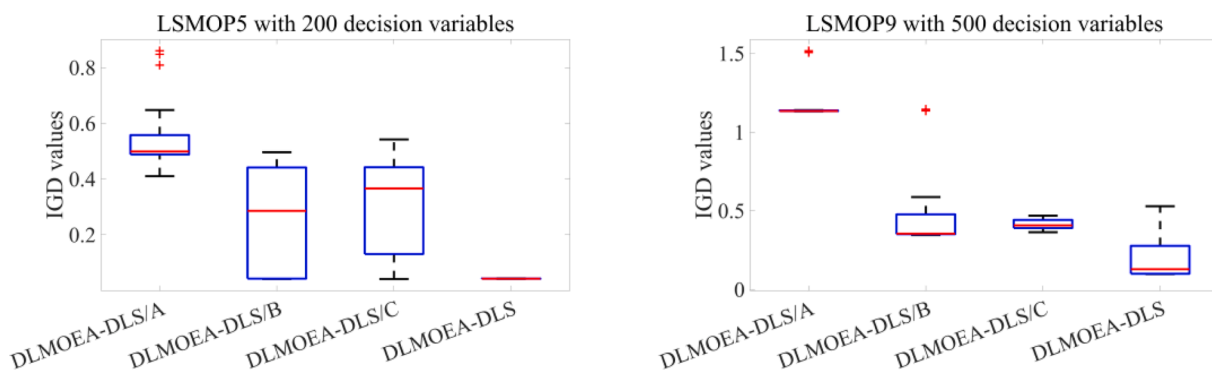

Fig. 13. The boxplots of the IGD values obtained by compared algorithms.

100, 200 and 500 decision variables. The value of θ was set to 0.2, 0.4, 0.6, 0.7, 0.9 and 0.8, which are termed as D-Theta2, D-Theta4, D-Theta6, D-Theta7, D-Theta9 and DLMOEA-DLS, respectively. The population size was set to 153, and 20 runs independently were performed.

The experimental results of the Wilcoxon rank sum for the IGD values are shown in Table 7. It can be observed that DLMOEA-DLS achieves competitive performance on LSMOP benchmark. D-Theta2, D-theta4 and D-Theta9 obtain worse performance than DLMOEA-DLS, meanwhile, D-Theta6 performs poor IGD values with the decision variables increasing, meaning the population exhibits bad convergence and diversity. It can be found that the algorithm with different θ obtains varying experimental results. In summary, these experimental results indicate that θ is sensitive to the algorithm, moreover, the DLMOEA-DLS achieves the best results when θ is set to 0.8.

4.9. The analysis of the termination criterion

For further fair comparison, all the compared algorithms were independently performed 20 times with three different termination criterions (i.e., 1500 * D, 5000 * D, and 15,000 * D) to analyze the effectiveness of the DLMOEA-DLS. The Wilcoxon rank sum for IGD values obtained by the compared algorithms are shown in Tables 8–10, respectively.

As observed in Table 8, it can be seen that the DLMOEA-DLS achieves the worst results when the termination criterion is set to 1500 * D, and the LMEA has similar performance with the DLMOEA-DLS. The reason for this phenomenon may be that the termination criterion is inadequate to perform the variables clustering strategy, and the dual-stage optimization is not performed. Moreover, the same phenomenon also can be observed in Table 9, the DLMOEA-DLS fails to obtain promising results when the number of the decision variables is set to 1000. However, it can be seen that the DLMOEA-DLS performs better than the other compared algorithms when the number of decision variables is less than 1000. Specifically, the proportion of benchmark instances that the DLMOEA-DLS performs better than and similar to the DGEA, the LMEA, the LMOCSO, the LMOEA-DS and the WOF are 20/27, 27/27, 12/27, 22/27 and 22/27, respectively. The reason for above experimental results may be that the DLMOEA-DLS is completely performed on LSMOP benchmark with 100, 200 and 500 decision variables. Moreover, when the number of function evaluations is set to 15,000 * D, the DLMOEA-DLS could perform completely and obtain the promising results, the results from Table 10 demonstrate that the DLMOEA-DLS exhibits competitive performance on the LSMOP benchmark.

To visually show the effectiveness of termination criterion for the compared algorithms, Fig. 11 plots the logarithm of the IGD values obtained by the compared algorithms on LSMOP3 with 200 decision variables and LSMOP6 with 500 decision variables. It can be seen that the DLMOEA-DLS achieves better IGD values under larger computation time. The experimental results indicate that the DLMOEA-DLS could improve the quality of the final solutions and exhibit significantly competitive performance when the larger number of function evaluations is set for termination criterion (i.e., 15,000 * D).

4.10. The effectiveness of each component in the DLMOEA-DLS

Three variants of the DLMOEA-DLS, named as DLMOEA-DLS/A, DLMOEA-DLS/B and DLMOEA-DLS/C, respectively, were constructed and performed to determine the contribution of each component in the DLMOEA-DLS. The DLMOEA-DLS/A adopted a single-stage optimization strategy, in which the DLS was performed. In the DLMOEA-DLS/B, two solutions were randomly selected without comparison. The DLMOEA-DLS/C did not employ the random population in (8), and the other part of the algorithm was still performed. A serious of experiments were performed 20 times independently on tri-objective LSMOP benchmark with 100, 200, and 500 decision variables. The experimental results are listed in Table 11. It can be observed that the DLMOEA-DLS obtains the

Table 12
The IGD values of compared algorithms on tri-objective LSMOP1-LSMOP9, where the best result is shown in bold font and gray background.

Problem	Dec	DLMOEA-Fix	DLMOEA-Random	DLMOEA-Null	DLMOEA-DLS
LSMOP1	100	3.4012e−2 (2.71e−3) =	3.5947e−2 (6.49e−3) =	4.8180e−2 (1.97e−2) -	**3.3457e−2 (4.54e−2)**
	200	3.5358e−2 (7.17e−3) =	3.7739e−2 (1.04e−2) -	6.7352e−2 (5.43e−2) -	**3.2912e−2 (4.47e−4)**
	500	3.5462e−2 (9.88e−3) =	3.3740e−2 (6.86e−3) =	5.0527e−2 (2.86e−2) -	**3.2086e−2 (2.04e−4)**
LSMOP2	100	6.9200e−2 (1.08e−2) -	6.7195e−2 (1.04e−2) -	8.0414e−2 (3.01e−2) -	**5.6152e−2 (3.91e−3)**
	200	4.9224e−2 (3.92e−3) =	**4.8259e−2 (4.53e−3) =**	6.9381e−2 (1.34e−2) -	4.9379e−2 (4.57e−3)
	500	3.7069e−2 (9.12e−4) =	3.7500e−2 (6.09e−4) -	4.9125e−2 (2.33e−3) -	**3.6766e−2 (1.40e−3)**
LSMOP3	100	**1.3901e−1 (3.45e−2) =**	2.1444e−1 (1.43e−1) =	1.9304e−1 (1.25e−1) =	1.7476e−1 (8.84e−2)
	200	2.1733e−1 (1.50e−1) =	2.4406e−1 (1.85e−1) =	**2.0683e−1 (1.36e−1) =**	2.1254e−1 (1.67e−1)
	500	3.5265e−1 (2.31e−1) -	2.9067e−1 (2.03e−1) -	3.2752e−1 (2.12e−1) -	**1.9359e−1 (5.87e−2)**
LSMOP4	100	6.9507e−2 (4.54e−2) =	**5.3749e−2 (4.31e−3) =**	5.5432e−2 (7.47e−3) =	5.4239e−2 (2.45e−3)
	200	4.7706e−2 (1.59e−2) =	5.5782e−2 (3.02e−2) -	8.0477e−2 (6.67e−2) -	**4.4120e−2 (2.25e−3)**
	500	**3.5481e−2 (6.30e−4) +**	3.9004e−2 (7.54e−3) =	5.5176e−2 (2.33e−2) -	3.8994e−2 (8.87e−3)
LSMOP5	100	1.0294e−1 (4.67e−2) -	1.2237e−1 (8.60e−2) -	1.8108e−1 (1.22e−1) -	**4.3102e−2 (9.57e−4)**
	200	2.3713e−1 (1.91e−1) -	3.1120e−1 (1.69e−1) -	2.9499e−1 (1.77e−1) -	**4.0986e−2 (1.29e−5)**
	500	4.0835e−1 (1.81e−1) -	3.5684e−1 (1.83e−1) -	4.5738e−1 (2.20e−1) -	**4.0984e−2 (9.25e−6)**
LSMOP6	100	5.7196e−1 (6.24e−2) =	**5.2528e−1 (7.15e−2) =**	8.1850e−1 (1.86e−1) -	5.3930e−1 (5.84e−2)
	200	5.9818e−1 (4.89e−2) -	5.8391e−1 (5.13e−2) =	1.0833e+0 (1.56e−1) -	**5.6746e−1 (4.06e−2)**
	500	5.7127e−1 (3.80e−2) -	5.7139e−1 (6.68e−2) -	1.1574e+0 (2.13e−1) -	**5.2040e−1 (8.34e−2)**
LSMOP7	100	1.0475e+0 (2.13e−1) -	9.7393e−1 (2.74e−1) -	1.1393e+0 (3.80e−1) -	**6.0518e−1 (1.15e−1)**
	200	9.4456e−1 (2.67e−1) -	9.2913e−1 (2.63e−1) -	1.2966e+0 (3.18e−1) -	**5.5410e−1 (1.89e−1)**
	500	7.7425e−1 (1.26e−2) -	7.5826e−1 (4.83e−2) -	8.3038e−1 (1.22e−1) -	**6.5371e−1 (1.69e−1)**
LSMOP8	100	1.1157e−1 (1.59e−2) -	1.0621e−1 (1.87e−2) =	1.3187e−1 (3.53e−2) -	**9.6931e−2 (1.90e−2)**
	200	7.4496e−2 (1.15e−2) -	7.5584e−2 (1.13e−2) -	8.5298e−2 (1.41e−2) -	**6.1129e−2 (4.68e−3)**
	500	5.3089e−2 (1.82e−3) =	**5.0637e−2 (4.22e−3) +**	5.6877e−2 (3.98e−3) -	5.3716e−2 (2.66e−3)
LSMOP9	100	5.7020e−1 (2.63e−1) -	4.7810e−1 (1.78e−1) -	5.9351e−1 (2.44e−1) -	**3.5200e−1 (7.00e−2)**
	200	6.1006e−1 (3.62e−1) -	5.6758e−1 (3.46e−1) -	5.3938e−1 (2.68e−1) -	**3.6078e−1 (1.08e−2)**
	500	4.3357e−1 (2.43e−1) -	4.7164e−1 (2.91e−1) -	4.1407e−1 (3.24e−2) -	**2.2569e−1 (1.67e−1)**
+/-/=		1/14/12	1/16/10	0/24/3	

best performance with 21 best results on the 27 instances, followed by the DLMOEA-DLS/B with 4 instances, and the DLMOEA-DLS/A and the DLMOEA-DLS/C have achieved only one best results respectively. It can be seen from the table that the DLMOEA-DLS achieves better performance with the number of decision variables increasing.

Fig. 12 plots the evolutionary curves of the logarithm of IGD values obtained by compared algorithms on LSMOP5 with 200 decision variables and LSMOP9 with 500 decision variables. Obviously, the DLMOEA-DLS achieves the best performance in the later stage of the evolution process. The boxplots of IGD values are shown in Fig. 13. The final population obtained by the DLMOEA-DLS is more stable, which means that the DLMOEA-DLS achieves robust performance. The experimental results from the DLMOEA-DLS and its variants verify the effectiveness of the DLMOEA-DLS.

4.11. The effectiveness of dynamic learning strategy

In the DLMOEA-DLS, the DLS plays a significant role in enhancing the convergence and the diversity of the population. To verify the effectiveness of the DLS, a series of experiments were performed on tri-objective LSMOP1-LSMOP9 with 100, 200 and 500 decision variables. The algorithms with modified learning strategies in the diversity stage are termed as the DLMOEA-Fix, the DLMOEA-Random and the DLMOEA-Null, respectively. In the DLMOEA-Fix, the control parameter F was initialized between 0 and 1 and fixed during the evolution. The DLMOEA-Random randomly reinitialized the parameter F at each iteration. The parameter F was set as 0 in the DLMOEA-Null, and the convergence stage was still performed. All the experiments were implemented 20 times independently, and the termination criterion was 15,000 * D. The experiments results are shown in Table 12. The results demonstrate and validate that the DLMOEA-DLS performs significantly better than other algorithms on 20 LSMOP benchmark instances, and the DLS is effective for addressing the LSMOPs.

5. Conclusion

In this paper, a new LSMOEA has been proposed to solve the LSMOPs, named DLMOEA-DLS. The DLMOEA-DLS involves the convergence and the diversity stages. A generation operator is employed to evolve the convergence-related and diversity-related decision variables for approximating PF in the convergence stage, and a DLS is implemented in the diversity stage to diversify the population along the PF, in which a dynamically adjusting strategy for the control parameter is designed to generate reasonable value. To verify the effectiveness of the DLMOEA-DLS, five state-of-the-art LSMOEAs are used for comparison on a set of LSMOPs benchmark instances. The experimental results show and validate that the DLMOEA-DLS is effective to deal with the LSMOPs.

The DLMOEA-DLS has shown potential ability in addressing the LSMOPs. In future work, more adaptively adjusting learning strategies are expected in the DLMOEA-DLS to address various complicated LSMOPs, such as large-scale many-objective optimization problems and the LSMOPs with complex landscapes.

CRediT authorship contribution statement

Jie Cao: Conceptualization, Project administration, Funding acquisition, Supervision. **Kaiyue Guo:** Conceptualization, Methodology, Data curation, Writing – original draft. **Jianlin Zhang:** Writing – review & editing, Supervision. **Zuohan Chen:** Visualization, Validation, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgement

This work was financially supported by the National Key Research and Development Plan under grant number 2020YFB1713600.

References

Alcalá-Fdez, J., Sánchez, L., Garcia, S., del Jesus, M. J., Ventura, S., Garrell, J. M., … Rivas, V. M. (2009). KEEL: A software tool to assess evolutionary algorithms for data mining problems. *Soft Computing, 13*, 307–318.

Antonio, L. M., & Coello, C. A. C. (2013). Use of cooperative coevolution for solving large scale multiobjective optimization problems. In *2013 IEEE Congress on Evolutionary Computation* (pp. 2758–2765). IEEE: Cancun, Mexico.

Brest, J., Greiner, S., Boskovic, B., Mernik, M., & Zumer, V. (2006). Self-Adapting Control Parameters in Differential Evolution: A Comparative Study on Numerical Benchmark Problems. *IEEE Transactions on Evolutionary Computation, 10*, 646–657.

Cao, J., Zhang, J., Zhao, F., & Chen, Z. (2021). A two-stage evolutionary strategy based MOEA/D to multi-objective problems. *Expert Systems with Applications, 185*, Article 115654.

Cao, B., Zhao, J., Gu, Y., Ling, Y., & Ma, X. (2020). Applying graph-based differential grouping for multiobjective large-scale optimization. *Swarm and Evolutionary Computation, 53*, Article 100626.

Cao, B., Zhao, J., Lv, Z., & Liu, X. (2017). A Distributed Parallel Cooperative Coevolutionary Multiobjective Evolutionary Algorithm for Large-Scale Optimization. *IEEE Transactions on Industrial Informatics, 13*, 2030–2038.

Cheng, R., & Jin, Y. (2015). A Competitive Swarm Optimizer for Large Scale Optimization. *IEEE Transactions on Cybernetics, 45*, 191–204.

Cheng, R., Jin, Y., & Olhofer, M. (2017). Test Problems for Large-Scale Multiobjective and Many-Objective Optimization. *IEEE Transactions on Cybernetics, 47*, 4108–4121.

Cheng, J., Pan, Z., Liang, H., Gao, Z., & Gao, J. (2021). Differential Evolution Algorithm with Fitness and Diversity Ranking-Based Mutation Operator. *Swarm and Evolutionary Computation, 61*, Article 100816.

Czyżżak, P., & Jaszkiewicz, A. (1998). Pareto simulated annealing—a metaheuristic technique for multiple-objective combinatorial optimization. *Journal of Multi-Criteria Decision Analysis, 7*, 34–47.

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation, 6*, 182–197.

Dong, J., Gong, W., Ming, F., & Wang, L. (2022). A two-stage evolutionary algorithm based on three indicators for constrained multi-objective optimization. *Expert Systems with Applications, 195*, Article 116499.

Fan, Q., Wang, W., & Yan, X. (2019). Differential evolution algorithm with strategy adaptation and knowledge-based control parameters. *Artificial Intelligence Review, 51*, 219–253.

He, C., Cheng, R., & Yazdani, D. (2022). Adaptive Offspring Generation for Evolutionary Large-Scale Multiobjective Optimization. *IEEE Transactions on Systems, Man, and Cybernetics: Systems, 52*, 786–798.

He, C., Cheng, R., Zhang, C., Tian, Y., Chen, Q., & Yao, X. (2020). Evolutionary large-scale multiobjective optimization for ratio error estimation of voltage transformers. *IEEE Transactions on Evolutionary Computation, 24*, 868–881.

He, C., Li, L., Tian, Y., Zhang, X., Cheng, R., Jin, Y., & Yao, X. (2019). Accelerating large-scale multiobjective optimization via problem reformulation. *IEEE Transactions on Evolutionary Computation, 23*, 949–961.

Iorio, A. W., & Li, X. (2004). A Cooperative Coevolutionary Multiobjective Algorithm Using Non-dominated Sorting. In *Genetic and Evolutionary Computation Conference* (pp. 537–548). Berlin, Heidelberg: Springer.

Jinlu, Z., Lixin, W., Rui, F., Hao, S., & Ziyu, H. (2022). Solve large-scale many-objective optimization problems based on dual analysis of objective space and decision space. *Swarm and Evolutionary Computation, 70*, Article 101045.

Junhao, Z., Lingjie, L., Qiuzhen, L., & Zhong, M. (2021). An Improved Weighted Optimization-based Framework for Large-scale MOPs. In *2021 IEEE Congress on Evolutionary Computation (CEC)* (pp. 2156–2163). Kraków, Poland: IEEE.

Kukkonen, S., & Lampinen, J. (2005). GDE3: The third evolution step of generalized differential evolution. In *2005 IEEE Congress on Evolutionary Computation* (pp. 443–450). Edinburgh, UK: IEEE.

Li, M., & Wei, J. (2018). A cooperative co-evolutionary algorithm for large-scale multi-objective optimization problems. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion (GECCO)* (pp. 1716-1721). New York, NY, USA: Association for Computing Machinery.

Li, Z., Shi, L., Yue, C., Shang, Z., & Qu, B. (2019). Differential evolution based on reinforcement learning with fitness ranking for solving multimodal multiobjective problems. *Swarm and Evolutionary Computation, 49*, 234–244.

Li, M., Yang, S., & Liu, X. (2014). Shift-Based Density Estimation for Pareto-Based Algorithms in Many-Objective Optimization. *IEEE Transactions on Evolutionary Computation, 18*, 348–365.

Li, L., Yen, G. G., Sahoo, A., Chang, L., & Gu, T. (2021). On the Estimation of Pareto Front and Dimensional Similarity in Many-objective Evolutionary Algorithm. *Information Sciences, 563*, 375–400.

Lin, Q., Liu, S., Zhu, Q., Tang, C., Song, R., Chen, J., … Zhang, J. (2016). Particle Swarm Optimization With a Balanceable Fitness Estimation for Many-Objective Optimization Problems. *IEEE Transactions on Evolutionary Computation, 22*, 32–46.

Liu, S., Lin, Q., Wong, K.-C., Li, Q., & Tan, K. C. (2021c). Evolutionary Large-Scale Multiobjective Optimization: Benchmarks and Algorithms. *IEEE Transactions on Evolutionary Computation*, 1-1.

Liu, S., Lin, Q., Li, Q., & Tan, K. C. (2021). A Comprehensive Competitive Swarm Optimizer for Large-Scale Multiobjective Optimization. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 1–14.

Liu, S., Lin, Q., Tian, Y., & Tan, K. C. (2021). A Variable Importance-Based Differential Evolution for Large-Scale Multiobjective Optimization. *IEEE Transactions on Cybernetics*, 1–15.

Liu, S., Lin, Q., Wong, K.-C., Ma, L., Coello, C. A. C., & Gong, D. (2019). A novel multi-objective evolutionary algorithm with dynamic decomposition strategy. *Swarm and Evolutionary Computation, 48*, 182–200.

Liu, R., Liu, J., Li, Y., & Liu, J. (2020). A random dynamic grouping based weight optimization framework for large-scale multi-objective optimization problems. *Swarm and Evolutionary Computation, 55*, Article 100684.

Ma, X., Liu, F., Qi, Y., Wang, X., Li, L., Jiao, L., … Gong, M. (2016). A Multiobjective Evolutionary Algorithm Based on Decision Variable Analyses for Multiobjective Optimization Problems With Large-Scale Variables. *IEEE Transactions on Evolutionary Computation, 20*, 275–298.

Ma, H., Wei, H., Tian, Y., Cheng, R., & Zhang, X. (2021). A multi-stage evolutionary algorithm for multi-objective optimization with complex constraints. *Information Sciences, 560*, 68–91.

Miguel Antonio, L., & Coello Coello, C. A. (2016). Decomposition-Based Approach for Solving Large Scale Multi-objective Problems. In *International Conference on Parallel Problem Solving from Nature* (Vol. 9921, pp. 525–534). Cham: Springer.

Ming, F., Gong, W., Zhen, H., Li, S., Wang, L., & Liao, Z. (2021). A simple two-stage evolutionary algorithm for constrained multi-objective optimization. *Knowledge-Based Systems, 228*, Article 107263.

Nebro, A. J., Durillo, J. J., Garcia-Nieto, J., Coello, C. C., Luna, F., & Alba, E. (2009). SMPSO: A new PSO-based metaheuristic for multi-objective optimization. In *2009 IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making (MCDM)* (pp. 66–73). Nashville, TN, USA: IEEE.

Ning, Y., Peng, Z., Dai, Y., Bi, D., & Wang, J. (2019). Enhanced particle swarm optimization with multi-swarm and multi-velocity for optimizing high-dimensional problems. *Applied Intelligence, 49*, 335–351.

Omidvar, M. N., Li, X., Mei, Y., & Yao, X. (2014). Cooperative Co-Evolution With Differential Grouping for Large Scale Optimization. *IEEE Transactions on Evolutionary Computation, 18*, 378–393.

Omidvar, M. N., Li, X., Yang, Z., & Yao, X. (2010). Cooperative co-evolution for large scale optimization through more frequent random grouping. In *IEEE Congress on Evolutionary Computation* (pp. 1–8). IEEE.

Qin, A. K., Huang, V. L., & Suganthan, P. N. (2008). Differential evolution algorithm with strategy adaptation for global numerical optimization. *IEEE Transactions on Evolutionary Computation, 13*, 398–417.

Qin, S., Sun, C., Jin, Y., Tan, Y., & Fieldsend, J. (2021). Large-scale evolutionary multiobjective optimization assisted by directed sampling. *IEEE Transactions on Evolutionary Computation, 25*, 724–738.

Shen, J., Wang, P., Dong, H., Li, J., & Wang, W. (2022). A Multistage Evolutionary Algorithm for Many-Objective Optimization. *Information Sciences, 589*, 531–549.

Song, Z., Wang, H., & Xu, H. (2022). A framework for expensive many-objective optimization with Pareto-based bi-indicator infill sampling criterion. *Memetic Computing, 14*, 179–191.

Song, A., Yang, Q., Chen, W.-N., & Zhang, J. (2016). A random-based dynamic grouping strategy for large scale multi-objective optimization. In *2016 IEEE Congress on Evolutionary Computation (CEC)* (pp. 468–475). Vancouver, BC, Canada: IEEE.

Tan, K. C., Feng, L., & Jiang, M. (2021). Evolutionary Transfer Optimization - A New Frontier in Evolutionary Computation Research. *IEEE Computational Intelligence Magazine, 16*, 22–33.

Tanabe, R., & Fukunaga, A. (2013). Success-history based parameter adaptation for Differential Evolution. In *2013 IEEE Congress on Evolutionary Computation* (pp. 71–78). Cancun, Mexico: IEEE.

Tang, H., Yu, F., Zou, J., Yang, S., & Zheng, J. (2022). A constrained multi-objective evolutionary strategy based on population state detection. *Swarm and Evolutionary Computation, 68*, Article 100978.
Tian, Y., Cheng, R., Zhang, X., Cheng, F., & Jin, Y. (2017). An Indicator Based Multi-Objective Evolutionary Algorithm with Reference Point Adaptation for Better Versatility. *IEEE Transactions on Evolutionary Computation, 22*, 609–622.
Tian, Y., Cheng, R., Zhang, X., & Jin, Y. (2017). PlatEMO: A MATLAB Platform for Evolutionary Multi-Objective Optimization [Educational Forum]. *IEEE Computational Intelligence Magazine, 12*, 73–87.
Tian, Y., Xiang, X., Zhang, X., Cheng, R., & Jin, Y. (2018). Sampling reference points on the Pareto fronts of benchmark multi-objective optimization problems. In *Proceedings of the 2018 IEEE World Congress on Computational Intelligence (WCCI 2018)*: University of Surrey.
Tian, Y., Si, L., Zhang, X., Cheng, R., He, C., Tan, K. C., & Jin, Y. (2022). Evolutionary Large-Scale Multi-Objective Optimization: A Survey. *ACM Computing Surveys, 54*, 1–34.
Tian, Y., Yang, S., Zhang, L., Duan, F., & Zhang, X. (2019). A Surrogate-Assisted Multiobjective Evolutionary Algorithm for Large-Scale Task-Oriented Pattern Mining. *IEEE Transactions on Emerging Topics in Computational Intelligence, 3*, 106–116.
Tian, Y., Zhang, X., Cheng, R., & Jin, Y. (2016). A multi-objective evolutionary algorithm based on an enhanced inverted generational distance metric. In *2016 IEEE Congress on Evolutionary Computation (CEC)* (pp. 5222–5229). Vancouver, BC, Canada: IEEE.
Tian, Y., Zheng, X., Zhang, X., & Jin, Y. (2020). Efficient Large-Scale Multiobjective Optimization Based on a Competitive Swarm Optimizer. *IEEE Transactions on Cybernetics, 50*, 3696–3708.
Wang, X., Zhang, K., Wang, J., & Jin, Y. (2021a). An enhanced competitive swarm optimizer with strongly convex sparse operator for large-scale multi-objective optimization. *IEEE Transactions on Evolutionary Computation*, 1-1.
Wang, Z., Zhan, Z., Kwong, S., Jin, H., & Zhang, J. (2021). Adaptive Granularity Learning Distributed Particle Swarm Optimization for Large-Scale Optimization. *IEEE Transactions on Cybernetics, 51*, 1175–1188.
Yang, X., Zou, J., Yang, S., Zheng, J., & Liu, Y. (2021). A Fuzzy Decision Variables Framework for Large-scale Multiobjective Optimization. *IEEE Transactions on Evolutionary Computation*, 1-1.
Zhang, Q., & Li, H. (2008). MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. *IEEE Transactions on Evolutionary Computation, 11*, 712–731.
Zhang, X., Tian, Y., Cheng, R., & Jin, Y. (2015). An Efficient Approach to Nondominated Sorting for Evolutionary Multiobjective Optimization. *IEEE Transactions on Evolutionary Computation, 19*, 201–213.
Zhang, X., Tian, Y., Cheng, R., & Jin, Y. (2018). A Decision Variable Clustering-Based Evolutionary Algorithm for Large-Scale Many-Objective Optimization. *IEEE Transactions on Evolutionary Computation, 22*, 97–112.
Zhang, X., Zheng, X., Cheng, R., Qiu, J., & Jin, Y. (2018). A competitive mechanism based multi-objective particle swarm optimizer with fast convergence. *Information Sciences, 427*, 63–76.
Zhou, J., Fang, W., Wu, X., Sun, J., & Cheng, S. (2016). An opposition-based learning competitive particle swarm optimizer. In *2016 IEEE Congress on Evolutionary Computation (CEC)* (pp. 515–521). Vancouver, BC, Canada: IEEE.
Zille, H., Ishibuchi, H., Mostaghim, S., & Nojima, Y. (2018). A Framework for Large-Scale Multiobjective Optimization Based on Problem Transformation. *IEEE Transactions on Evolutionary Computation, 22*, 260–275.
Zitzler, E., & Thiele, L. (1998). Multiobjective optimization using evolutionary algorithms — A comparative case study. In A. E. Eiben, T. Bäck, M. Schoenauer, & H.-P. Schwefel (Eds.), *Parallel Problem Solving from Nature — PPSN V* (pp. 292–301). Berlin, Heidelberg: Springer, Berlin Heidelberg.