Benchmark Problems for WCCI2026 Competition on Dynamic Multiobjective Optimisation

Juan Zou[1], Xiaozhong Yu[1], Zhanglu Hou[1], Hui Bai[1], Xinjie Zhao[1], Yaru Hu[1], Shouyong Jiang[2], Shengxiang Yang[3]

1. Key Laboratory of Intelligent Computing and Information Processing, Ministry of Education of China, Xiangtan University, Xiangtan, Hunan Province, China

2. School of Automation, Central South University, Changsha, Hunan Province, China.

3. De Montfort University, Leicester, LE1 9BH, U.K.

1.Introduction

Over the past decade, there has been rising interest in research on dynamic multiobjective optimisation[1-3]. This is a challenging yet highly important topic that addresses problems with multiobjective and time-varying characteristics. Due to the existence of dynamics, dynamic multiobjective optimization problems (DMOPs) are inherently more complex and demanding than static multiobjective problems, presenting considerable challenges to evolutionary algorithms (EAs) when solving them. Broadly speaking, DMOPs bring about at least three main challenges. First, environmental changes are hard to detect. If they go undetected, they can mislead the search process because nondominated solutions identified for the previous environment may no longer be valid in the current one[4]. Second, Diversity, which acts as the primary driving force of population-based algorithms, is highly sensitive to dynamics. The dynamic nature of DMOPs, which is defined by irregular changes multimodality and discrete Pareto optimal sets (PSs) or fronts (PFs), makes the optimisation process significantly more complex[5]. Finally, algorithms often face tight time constraints when responding to environmental changes. These time limits on DMOPs require algorithms to balance diversity and convergence effectively. This balance allows them to deal with environmental changes promptly and closely track time-varying PSs or PFs[6]. These challenges underline the need to develop more complex and comprehensive test problems. Doing so will encourage the creation of innovative methodologies to overcome these difficulties[7-8].

Benchmark problems are highly significant to algorithm analysis, enabling algorithm designers and practitioners to gain a clearer understanding of the strengths and weaknesses of evolutionary algorithms. In dynamic multi-objective optimisation, several widely used test suites exist, including FDA, dMOP and JY. However, these problem suites oversimplify the complexity of variations in real-world problems, capturing only specific facets of actual scenarios. For instance, the FDA and dMOP functions pose no detection challenges for algorithms. Environmental changes in these problems can be readily identified through a single re-evaluation of a random population member, which is far simpler than real-life environmental shifts. It is also acknowledged that most existing dynamic multi-objective optimisation problems (DMOPs) are direct adaptations of popular static test suites, such as ZDT and DTLZ. Consequently, these DMOPs exhibit considerable similarity in their problem characteristics, making them of limited value for comprehensive algorithm analysis. Furthermore, a concerning feature of most existing DMOPs is that static problem properties dominate over the dynamic elements to an excessive degree. A problem property, such as strong variable dependency, that presents challenges for static multi-objective optimisation may not be well-suited to dynamic multi-objective optimisation. One key reason is that algorithmic underperformance on DMOPs often stems from the static property rather than the presence of dynamics. Using such DMOPs can therefore lead to misleading inferences about algorithm performance. Additionally, most benchmark designs are founded on the premise that environments remain similar before and after a change. In real-world contexts, however, many DMOPs involve erratic environmental shifts. In such cases, the search directions employed by EAs for the current environment may be unsuitable for the new one, particularly when the PS of the new environment diverges substantially from, and in the worst-case scenario, even points in the opposite direction to, that of the current environment. In summary, a suite of diverse and unbiased benchmark test problems is urgently needed in the field to support the systematic investigation of evolutionary algorithms.

In this competition, a total of 19 benchmark functions are introduced[9-11] covering representative types of DMOPs (continuous, and constrained) with diverse properties found in various real-world scenarios. Such as irregular changes of PS or PF, time-dependent PF/PS geometries, disconnectivity, degeneration, detectability, and a changing number of decision variables and/or objective functions. All the algorithms are implemented on MATLAB under the framework of PlatEMO[12]. Through suggesting a set of benchmark functions with a good representation of various real-world scenarios, we aim to promote the research on evolutionary dynamic multiobjective optimisation. All the benchmark functions have been implemented in MATLAB code, which can be downloaded in the following website:

https://github.com/yxz996/WCCI2026DMOP

2. Summary of 19 test problems

The proposed test suite (called RDP and RDC in this competition) has 19 bi-objective problems. The main dynamic characteristics of each problem are briefly summarized in Table 1.

Problems	Objectives	Types

RDP1	2	POS dynamic, POF dynamic
RDP2	2	POS dynamic, POF dynamic
RDP3	2	POS dynamic, POF dynamic
RDP4	2	POS dynamic, POF dynamic
RDP5	2	POS dynamic, POF dynamic
RDP6	2	POS dynamic, POF static
RDP7	2	POS dynamic, POF dynamic
RDP8	2	POS static, POF dynamic
RDP9	2	POS dynamic, POF static
RDP10	2	POS static, POF dynamic
RDC1	2	Dynamic objectives and static constraint
RDC2	2	Dynamic objectives and static constraint
RDC3	2	Dynamic objectives and static constraint
RDC4	2	Static objectives and dynamic constraint
RDC5	2	Static objectives and dynamic constraint
RDC6	2	Static objectives and dynamic constraint
RDC7	2	Dynamic constraint and dynamic objectives
RDC8	2	Dynamic constraint and dynamic objectives
RDC9	2	Dynamic constraint and dynamic objectives

3.Test Problems

Although existing test benchmarks(such as the FDA, JY, and DF) can simulate various environmental dynamic characteristics including linear and nonlinear changes, their PS/PF change trends usually only rely on the driving of uniform time parameters, resulting in relatively single dynamic patterns. However, in practical applications, the evolution trajectory of PS (such as position migration) often shows stronger random characteristics. To bridge this gap, this competition uses a time parameter reconstruction method based on random sequences[13]. By introducing random sequences, the uniform change of the original time parameter t is altered, thereby realizing the random change of PS. Its mathematical definition is as follows.

$$t' = Q\left[\left\lfloor\frac{\tau}{\tau_t}\right\rfloor \% l\right] \cdot \frac{1}{n_t}$$

3.1 Dynamic Multiobjective Optimisation Problems

RDP1

$$\min \begin{cases} f_1(x) = g(x)(x_1 + 0.02\sin(w_t \pi x_1)) \\ f_2(x) = g(x)(1 - x_1 + 0.02\sin(w_t \pi x_1)) \end{cases}$$

with $g(x) = 1 + \sum_{i=2}^{n}(x_i - G(t))^2$,

where $G(t) = \sin(0.5\pi t)$, $w_t = \lfloor 10G(t) \rfloor$,

and the search space is $[0,1] \times [-1,1]^{n-1}$.

RDP2

$$\min \begin{cases} f_1(x) = g(x)(x_1 + 0.1\sin(3\pi x_1))^{\alpha_t} \\ f_2(x) = g(x)(1 - x_1 + 0.1\sin(3\pi x_1))^{\alpha_t} \end{cases}$$

with $g(x) = 1 + \sum_{i=2}^{n}(|G(t)|y_i^2 - 10\cos(2\pi y_i) + 10)$,

where $y_i = x_i - G(t)$, $G(t) = \sin(0.5\pi t)$, $\alpha_t = 0.2 + 2.8|G(t)|$,

and the search space is $[0,1] \times [-1,1]^{n-1}$.

RDP3

$$\min \begin{cases} f_1(x) = g(x)\dfrac{1+t}{x_1} \\ f_2(x) = g(x)\dfrac{x_1}{1+t} \end{cases}$$

with $g(x) = 1 + \sum_{i=2}^{n}(x_i - \frac{1}{1+e^{\alpha_t(x_1-2.5)}})^2$,

where $\alpha_t = 5\cos(0.5\pi t)$,

and the search space is $[1,4] \times [0,1]^{n-1}$.

RDP4

$$\min \begin{cases} f_1(x) = g(x)(x_1 + 0.1\sin(3\pi x_1))^{\alpha_t} \\ f_2(x) = g(x)(1 - x_1 + 0.1\sin(3\pi x_1))^{\alpha_t} \end{cases}$$

with $g(x) = 1 + \sum_{i=2}^{n}(x_i - \frac{G(t)\sin(4\pi x_1^{\beta_t})}{1+|G(t)|})^2$,

where $\alpha_t = 2.25 + 2\cos(2\pi t)$, $\beta_t = 100G^2(t)$, $G(t) = \sin(0.5\pi t)$,

and the search space is $[0,1] \times [-1,1]^{n-1}$.

RDP5

$$\min \begin{cases} f_1(x) = g(x)(x_1 + \max\{0, (\frac{1}{2N_t} + 0.1)\sin(2N_t\pi x_1)\}) \\ f_2(x) = g(x)(1 - x_1 + \max\{0, (\frac{1}{2N_t} + 0.1)\sin(2N_t\pi x_1)\}) \end{cases}$$

with $g(x) = 1 + \sum_{i=2}^{n}(x_i - \cos(4t + x_1 + x_{i-1}))^2$,

where $N_t = 1 + \lfloor 10|\sin(0.5\pi t)|\rfloor$,

and the search space is $[0,1] \times [-1,1]^{n-1}$.

RDP6

$$min \begin{cases} f_1(x) = (1 + g(x))(x_1 + 0.05\sin(6\pi x_1) \\ f_2(x) = (1 + g(x))(1 - x_1 + 0.05sin(6\pi x_1) \end{cases}$$

with $g(x) = \sum_{i=2}^{n} (x_i - G(t))^2$,
where $G(t) = \sin(0.5\pi t)$,
and the search space is $[0,1] \times [-1,1]^{n-1}$

RDP7

$$min \begin{cases} f_1(x) = (1 + g(x))(x_1 + 0.05\sin(w_t\pi x_1) \\ f_2(x) = (1 + g(x))(1 - x_1 + 0.05sin(w_t\pi x_1) \end{cases}$$

with $g(x) = \sum_{i=2}^{n} (x_i - G(t))^2$,
where $G(t) = \sin(0.5\pi t), , w_t = 10^{1+|G(t)|}$,
and the search space is $[0,1] \times [-1,1]^{n-1}$

RDP8

$$min \begin{cases} f_1(x) = (1 + g(x))(x_1 + A_t\sin(\pi x_1) \\ f_2(x) = (1 + g(x))(1 - x_1 + A_t sin(\pi x_1) \end{cases}$$

with $g(x) = \sum_{i=2}^{n} {x_i}^2$,
where $A_t = 0.3\sin(0.5\pi(t - 1)$,

and the search space is $[0,1] \times [-1,1]^{n-1}$

RDP9

$$\min \begin{cases} f_1(x) = (1 + g(x))(x_1 + 0.1\sin(3\pi x_1)) \\ f_2(x) = (1 + g(x))(1 - x_1 + 0.1\sin(3\pi x_1)) \end{cases}$$

with $g(x) = \sum_{i=2}^{n} \left(4y_i^2 - cos(K_t \pi y_i) + 1\right)$,
where $K_t = 2 \cdot \lfloor 10 \cdot |\mathrm{G}(t)| \rfloor, G(t) = \sin(0.5\pi t), y_i = x_i - G(t)$,
and the search space is $[0,1] \times [-1,1]^{n-1}$.

RDP10

$$\min \begin{cases} f_1(x) = (1 + g(x))(x_1 + 0.1sin(3\pi x_1))^{\alpha_t} \\ f_2(x) = \left(1 + g(x)\right)(1 - x_1 + 0.1sin(3\pi x_1))^{\beta_t} \end{cases}$$

with $g(x) = \sum_{xi \in xII} \left(y_i^2 - 10cos(2\pi y_i) + 10\right)$,
where $\alpha t = \beta t = 0.2 + 2.8 \cdot |G(t)|, G(t) = \sin(0.5\pi t), y_i = x_i - G(t)$,
and the search space is $[0,1] \times [-1,1]^{n-1}$.

3.2 Dynamic Constrained Multi-objective Optimization Problems

RDC1

$$\begin{aligned} \min \quad & \begin{cases} f_1 = g x_1 + G \\ f_2 = g(1 - x_1) + G \end{cases} \\ s.t. \quad & c = \cos(-0.15\pi) f_2 - \sin(-0.15\pi) f_1 \geq \\ & \left(2\sin\left(5\pi(\sin(-0.15\pi) f_2 + \cos(-0.15\pi) f_1)\right)\right)^6 \end{aligned}$$

with $g = 1 + \sum_{i=2}^{n} (x_i - G)^2$

where $(G = |\sin(0.5\pi t)|)$,

and the search space is $([0,1]^n)$

RDC2

$$\begin{aligned} \min \quad & \begin{cases} f_1 = g(x_1 + 0.25G\sin(\pi x_1)) \\ f_2 = g(1 - x_1 + 0.25G\sin(\pi x_1)) \end{cases} \\ s.t. \quad & c_1 = (4f_1 + f_2 - 1)(0.3f_1 + f_2 - 0.3) \geq 0 \\ & c_2 = 1.85 - f_1 - f_2 - (0.3\sin(3\pi(f_2 - f_1)))^2 (f_1 + f_2 - 1.3) \leq 0 \end{aligned}$$

with $g = 1 + \sum_{i=2}^{n} ((x_i - G)^2 + \sin(0.5\pi(x_i - G))^2)$

where $(G = \sin(0.5\pi t))$,

and the search space is $([0,1]\times[-1,1]^{n-1})$

RDC3

$$
\begin{aligned}
\min\quad & \begin{cases} f_1 = gx_1 + G^2 \\ f_2 = g(1-x_1) + G^2 \end{cases} \\
s.t.\quad & c_1 = ((f_1-1)^2 + (f_2-0.2)^2 - 0.3^2)((f_1-0.2)^2 + (f_2-1)^2 - 0.3^2) \geq 0 \\
& c_2 = (f_1^2 + f_2^2 - 4^2) \leq 0 \\
& c_3 = (f_1^2 + f_2^2 - (3.1 + 0.2\sin(4\arctan(f_2/f_1))^2)^2)(f_1^2 + f_2^2 - 2.3^2) \geq 0
\end{aligned}
$$

with $g = 1 + \sum_{i=2}^{n}\sqrt{|x_i - G|}$

where $(G = |\sin(0.5\pi t)|)$,

and the search space is $([0,1]^n)$

RDC4

$$
\begin{aligned}
\min\quad & \begin{cases} f_1 = g(x_1 + 0.25\sin(\pi x_1)) \\ f_2 = g(1 - x_1 + 0.25\sin(\pi x_1)) \end{cases} \\
s.t.\quad & c = (f_1^2 + f_2^2 - (1.3 - 0.45\sin(G\arctan(f_2/f_1))^2)^2) \\
& \quad \times (f_1^2 + f_2^2 - (1.5 + 0.4\sin(4\arctan(f_2/f_1))^{16})^2) \geq 0
\end{aligned}
$$

$$
\text{with } g = 1 + \sum_{i=2}^{n}((x_i - 0.5)^2 - \cos(\pi(x_i - 0.5)) + 1)^2
$$

where $(G = 2\lfloor 10|((t+1) \bmod 2) - 1|\rfloor)$,

and the search space is $([0,1]^n)$

RDC5

$$
\begin{aligned}
\min\quad & \begin{cases} f_1 = gx_1 \\ f_2 = g(1-x_1) \end{cases} \\
s.t.\quad & c_1 = ((0.2+G)f_1^2 + f_2 - 2)(0.7f_1^2 + f_2 - 2.5) \geq 0 \\
& c_2 = f_1^2 + f_2^2 - (0.6+G)^2 \geq 0
\end{aligned}
$$

with $g = 1 + \sum_{i=2}^{n}|x_i - 0.5\sin(2\pi x_1)|$

where $(G = 0.5|\sin(0.5\pi t)|)$,

and the search space is $([0,1]\times[-1,1]^{n-1})$

RDC6

$$\min \begin{cases} f_1 = gx_1 \\ f_2 = g(1-x_1) \end{cases}$$
$$s.t. \quad c_1 = f_1 + f_2 - \big(4.5 + 0.08\sin\big(2\pi(f_2 - f_1/1.6)\big)\big) \le 0$$
$$c_2 = \frac{(f_1\cos(-\pi/4) - f_2\sin(-\pi/4))^2}{1.1^2} + \frac{(f_1\sin(-\pi/4) + f_2\cos(-\pi/4))^2}{(0.1+G)^2} \ge (0.1+G)^2$$
$$c_3 = f_1 + f_2 - \big(3.2 - G - 0.08\sin\big(2\pi(f_2 - f_1/1.5)\big)\big) \ge 0$$

with $g = 1 + 6\sum_{i=2}^{n} x_i^2$

where $(G = \sqrt{|\sin(0.5\pi t)|})$,

and the search space is $([0,1] \times [-1,1]^{n-1})$

RDC7

$$\min \begin{cases} f_1 = g(x_1 + 0.02\sin((10 - |\lfloor 10G \rfloor|)\pi x_1)) \\ f_2 = g(1 - x_1 + 0.02\sin((10 - |\lfloor 10G \rfloor|)\pi x_1)) \end{cases}$$
$$s.t. \quad c = f_1 + f_2 - G - \sin(5\pi(f_1 - f_2 + 1))^2 \ge 0$$

with $g = 1 + \sum_{i=2}^{n} (x_i - G)^2$

where $(G = \sin(0.5\pi t))$,

and the search space is $([0,1] \times [-1,1]^{n-1})$

RDC8

$$\min \begin{cases} f_1 = gx_1 \\ f_2 = g(1-x_1) \end{cases}$$
$$s.t. \quad c_1 = \big(\sqrt{f_1} + \sqrt{f_2} - 0.95 - 0.5|G|\big)\big(f_1^{1.5} + f_2^{1.5} - 1.2^{1.5}\big) \ge 0$$
$$c_2 = \big(0.8f_1 + f_2 - (2.5 + 0.08\sin(2\pi(f_2 - f_1)))\big)$$
$$\times \big((0.93 + |G|/3)f_1 + f_2 - (2.7 + 0.5|G| + 0.08\sin(2\pi(f_2 - f_1)))\big) \ge 0$$

with $g = 1 + \sum_{i=2}^{n} \left(|G|(x_i - G)^2 - \cos(\pi(x_i - G)) + 1\right)^2$

where $(G = \sin(0.5\pi t))$,

and the search space is $([0,1] \times [-1,1]^{n-1})$

RDC9

$$\min \begin{cases} f_1 = gx_r \\ f_2 = g(1-x_r) \end{cases}$$
$$s.t. \quad c_1 = (f_1^h + f_2^h - 4^h) \times (f_1^2 + f_2^2 - (0.2+G)^2) \le 0$$
$$c_2 = \left(2.1 - \left(\frac{f_1}{1 + 0.15\cos(6\arctan(f_2/f_1)^3)^{10}}\right)^2 - \left(\frac{f_2}{1 + 0.75\cos(6\arctan(f_2/f_1)^3)^{10}}\right)^2\right) \times (f_1^2 + f_2^2 - 1.6^2) \le 0$$

with $g = 1 + 10\sum_{i=1,\ldots,n\backslash\{r\}}^{n}(x_i - G)^2$

where $(G = |\sin(0.5\pi t)|), (h = 0.75 + 1.25G), (r = 1 + \lfloor(n-1)G\rfloor)$,

and the search space is $([0,1]^n)$

4. Performance assessments

The following experimental settings are encouraged to use when conducting empirical studies on the proposed test suite.

4.1 General settings

- Population size: 100.
- Number of variables: 10.
- Frequency of change (τ_t): 10 (fast changing environments), 20 (slow changing environments).
- Severity of change (n_t): 5 (severe changing environments), 10 (moderate changing environments), 20.
- Number of changes: 30.
- Stopping criterion: a maximum number of 100(30τ_t+50) fitness evaluations, where 500 fitness evaluations are given before the first environmental change occurs.
- Number of independent runs: 20.

4.2 Performance metric

The MIGD is used to evaluate the performance of an optimizer on each DMOP[13]. A smaller MIGD value indicates a better performance of the corresponding optimizer. The MIGD value is calculated as follows.

$$MIGD = \frac{\sum_{i=1}^{T} IGD(POF_t^*, POF_t)}{T} = \frac{\sum_{i=1}^{T} \sum_{i=1}^{POF_t} \frac{dis_t^i}{POF_t}}{T}$$

where dis_t^i is the Euclidean distance between the i-th member in POF_t and its nearest member in POF_t^* .

Moreover, the MHV is used to measure the comprehensive performance of an optimizer on DMOPs[14]. A larger MHV value indicates a better performance of the corresponding optimizer. The MHV value is calculated as follows.

$$MHV = \frac{\sum_{i=1}^{T} HV(POF_t, rp)}{T}$$

where rp is the reference point for calculating the HV.

4.3 Results Format

To submit the result, it is expected to format the submitted competition results in tables as the same as Table 2. More especially, please do make sure that the submitted results are of high readability, and multiple types of results shown in Table are clearly recorded, including the mean and standard deviation of the MIGD/MHV values for each test instance.

For all participants, please also submit the corresponding source code which should allow the generation of reproducible results you're submitted. Besides, it would be nice if you can submit a document that gives a brief illustration to the algorithm and corresponding parameter settings.

Table 2 MIGD and MHV results obtained by your algorithm on RDP or RDC

Problem	(τ_t, n_t)	MIGD(mean(std.))	MHV(mean(std.))
RDP1	10,5	1.234E-2(1.234E-3)	1.234E-2(1.234E-3)
	10,10		
	20,5		
	20,10		
RDP2			
……			
RDP10	10,5		
	10,10		
	20,5		

	20,10

References

[1] Jiang S, Zou J, Yang S, et al. Evolutionary dynamic multi-objective optimisation: A survey[J]. ACM Computing Surveys, 2022, 55(4): 1-47.

[2] Jiang X, Chen Q, Ding J, et al. Dual-Population Evolution Based Dynamic Constrained Multiobjective Optimization With Discontinuous and Irregular Feasible Regions[J]. IEEE Transactions on Emerging Topics in Computational Intelligence, 2025.

[3] Song W, Liu S, Wang X, et al. Learning to guide particle search for dynamic multiobjective optimization[J]. IEEE Transactions on Cybernetics, 2024.

[4] Yu K, Zhang D, Liang J, et al. A correlation-guided layered prediction approach for evolutionary dynamic multiobjective optimization[J]. IEEE transactions on evolutionary computation, 2022, 27(5): 1398-1412.

[5] Hu Y, Zheng J, Jiang S, et al. Handling dynamic multiobjective optimization environments via layered prediction and subspace-based diversity maintenance[J]. IEEE Transactions on Cybernetics, 2021, 53(4): 2572-2585.

[6] Hu Y, Ou J, Suganthan P N, et al. Dynamic multi-objective optimization algorithm guided by recurrent neural network[J]. IEEE Transactions on Evolutionary Computation, 2024.

[7] Zou J, Hou Z, Jiang S, et al. Knowledge Transfer With Mixture Model in Dynamic Multi-Objective Optimization[J]. IEEE Transactions on Evolutionary Computation, 2025.

[8] Gong Q, Xia Y, Zou J, et al. Enhancing Dynamic Constrained Multi-Objective Optimization With Multi-Centers Based Prediction[J]. IEEE Transactions on Evolutionary Computation, 2025.

[9] Jiang S, Yang S, Yao X, et al. Benchmark Functions For The Cec'2018 Competition On Dynamic Multiobjective Optimization[R]. Newcastle University, 2018.

[10] Jiang S, Yang S. Evolutionary Dynamic Multiobjective Optimization: Benchmarks And Algorithm Comparisons[J]. IEEE Transactions On Cybernetics, 2016, 47(1): 198-211.

[11] Chen G, Guo Y, Wang Y, et al. Evolutionary Dynamic Constrained Multiobjective Optimization: Test Suite And Algorithm[J]. IEEE Transactions on Evolutionary Computation, 2023.

[12] Tian Y, Cheng R, Zhang X, Jin Y. PlatEMO: A MATLAB platform for evolutionary multi-objective optimization [educational forum], IEEE Computational Intelligence Magazine, 2017, 12(4): 73-87.

[13] Zou J, Hou Z, Jiang S, et al. Knowledge Transfer With Mixture Model in Dynamic Multi-Objective Optimization[J]. IEEE Transactions on Evolutionary Computation, 2025.

[14] Yu X, Zheng J, Hu Y, et al. An adaptive response algorithm based on dual-space detection for dynamic multiobjective optimization[J]. Swarm and Evolutionary Computation, 2025, 98: 102092.