# Benchmark Problems for CEC2025 Competition on Dynamic Multiobjective Optimisation

Juan Zou[1], Shouyong Jiang[2], Zhanglu Hou[1], Xiaozhong Yu[1], Yaru Hu[1], 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

The past decade has witnessed a growing amount of research interest in dynamic multiobjective optimisation[1-3], a challenging yet very important topic that deals with problems with multi-objective and time-varying properties. Due to the presence of dynamics, DMOPs are inherently more complex and challenging than static multiobjective problems, posing significant difficulties for evolutionary algorithms (EAs) in solving them. Generally speaking, DMOPs pose at least three main challenges. First, environmental changes are difficult to detect and, if undetected, can mislead the search process, as nondominated solutions found for the previous environment may no longer be valid in the current one[4]. Second, diversity, the key driving force of population-based algorithms, is highly sensitive to dynamics. The dynamic nature of DMOPs, characterized by irregular changes, multimodality, and discrete Pareto optimal sets (PSs) or fronts (PFs), significantly complicates the optimisation process[5]. Finally, the response time for environmental changes is often tight for algorithms. The time constraints on DMOPs require algorithms to strike a balance between diversity and convergence, enabling them to promptly handle environmental changes and closely track time-varying PSs or PFs[6]. These challenges highlight the imperative to introduce more complex and comprehensive test problems, thereby fostering the development of innovative methodologies to tackle them[7][8].

Benchmark problems are of great importance to algorithm analysis, which helps algorithm designers and practitioners to better understand the strengths and weaknesses of evolutionary algorithms. In dynamic multi-objective optimisation, there exist several widely used test suites, including FDA, dMOP and JY. However, these problem suites simplify the complexity of variations in real-world problems and only represent certain aspects of actual scenarios. For example, the FDA and dMOP functions have no detection difficulty for algorithms. Environmental changes involved in these problems can be easily detected with one re-evaluation of a random population member. Real-life environmental changes should not be so simple. It has also been recognised that most existing DMOPs are a direct modification of popular static test suites, e.g. ZDT and DTLZ. As a result, the DMOPs are more or less the same regarding their problem properties, and therefore are of limited use when a comprehensive algorithm analysis is pursued. Furthermore, another worrying characteristic of most existing DMOPs is that static problem properties overweigh too much dynamics. A problem property (e.g. strong variable dependency) that is challenging for static multiobjective optimisation may not be a good candidate for dynamic multiobjective optimisation. One reason for this is that a failure of algorithms for DMOPs is not due to the presence of dynamics, but rather the static property. It is therefore likely to get a misleading conclusion on the performance of algorithms when such DMOPs are used. Additionally, most benchmark designs are based on the assumption that the environments before and after the change are similar. However, in real-world scenarios, many DMOPs involve irregular environmental changes. In such cases, the search directions used by EAs for the current environment may not be suitable for the new environment, especially when the true PS of the new environment significantly deviates from, and in the worst case even points in the opposite direction to, that of the current environment. In a nutshell, a set of diverse and unbiased benchmark test problems for a systematic study of evolutionary algorithms are greatly needed in the area.

In this competition, a total of 20 benchmark functions are introduced[9-12], 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. 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/CEC2025DMOP

# 2. Summary of 20 test problems

The test suite (called DP/DC in this competition) has 20 bi-objective problems. The main dynamic characteristics of each problem are briefly summarized in Table 1.

Table 1: Main characteristics of the 20 test problems

| Problems | Objectives | Types |
|---|---|---|
| DP1 | 2 | POS dynamic, POF dynamic |
| DP2 | 2 | POS dynamic, POF dynamic |
| DP3 | 2 | POS dynamic, POF dynamic |
| DP4 | 2 | POS dynamic, POF static |
| DP5 | 2 | POS dynamic, POF dynamic |
| DP6 | 2 | POS dynamic, POF dynamic |

| | | |
|---|---|---|
| DP7 | 2 | POS dynamic, POF dynamic |
| DP8 | 2 | POS dynamic, POF dynamic |
| DP9 | 2 | POS dynamic, POF dynamic |
| DP10 | 2 | POS dynamic, POF dynamic |
| DC1 | 2 | Static objectives and dynamic constraint |
| DC2 | 2 | Static constraint and dynamic objectives |
| DC3 | 2 | dynamic constraint and dynamic objectives |
| DC4 | 2 | dynamic constraint and dynamic objectives |
| DC5 | 2 | Static constraint and dynamic objectives |
| DC6 | 2 | Static constraint and dynamic objectives |
| DC7 | 2 | Static constraint and dynamic objectives |
| DC8 | 2 | Static constraint and dynamic objectives |
| DC9 | 2 | dynamic constraint and dynamic objectives |
| DC10 | 2 | dynamic constraint and dynamic objectives |

# 3.Test Problems

# 3.1 Dynamic Multiobjective Optimisation Problems

# DP1

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

with $g(x) = 1 + \sum_{i=2}^{n} \left(|G(t)|y_i^2 - 10\cos(2\pi y_i) + 10\right)$,
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 $[0,1] \times [-1,1]^{n-1}$.

# DP2

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

with $g(x) = \sum_{i=2}^{n} \left(4y_i^2 - cos(2p_t\pi y_i) + 1\right)$,
where $y_i = x_i - cos(t), p_t = \lfloor 5|\sin(\pi t)| \rfloor$,
and the search space is $[0,1] \times [-1,1]^{n-1}$.

# DP3

$$min \begin{cases} f_1(x) = \left(1 + g(x)\right)cos(0.5\pi x_1) \\ f_2(x) = \begin{cases} \left(1 + g(x)\right)\left|k_t\left(cos(0.5\pi x_1) - cos(0.5\pi x\alpha_t)\right) + sin(0.5\pi\alpha_t)\right| & x_1 > \alpha_t \\ \left(1 + g(x)\right)sin(0.5\pi x_1) & x_1 \le \alpha_t \end{cases} \end{cases}$$

with $g(x)=\sum_{i=2}^{n}(x_i-0.5)^2(1+|\cos(8\pi x_i)|)$,
where $k_t=10\cos(2.5\pi t), \alpha_t=0.5|sin(\pi t)|$,
and the search space is $[0,1]$.

# DP4

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

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 $[0,1]\times[-1,1]^{n-1}$.

# DP5

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

with $g(x)=\sum_{i=2}^{n}\left(y_i^2-10cos(2\pi y_i)+10\right)$,
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 $[0,1]\times[-1,1]^{n-1}$.

# DP6

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

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 $[0,1]\times[-1,1]^{n-1}$.

# DP7

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

with $g(x)=1+\sum_{i=2}^{n}\left(x_i-\frac{G(t)\sin(4\pi x_1\beta_t)}{1+|G(t)|}\right)^2$,
where $\alpha_t=2.25+2cos(2\pi t), \beta_t=100G^2(t), G(t)=\sin(0.5\pi t)$,
and the search space is $[0,1]\times[-1,1]^{n-1}$.

# DP8

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

with $g(x)=1+\sum_{i=2}^{n}(x_i-\cos(4t+x_1+x_{i-1}))^2$,
where $N_t=1+\lfloor 10|G(t)|\rfloor, G(t)=\sin(0.5\pi t)$,

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

# DP9

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

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 $[0,1]\times[-1,1]^{n-1}$

# DP10

$$min\begin{cases} f_1(x) = (1+g(x)\cdot cos^2(0.5\pi x_1)+G(t) \\ f_2(x) = [sin^2(0.5\pi x_1)+sin(0.5\pi x_1)cos2(p_t\pi x_1)]+G(t) \end{cases}$$

with $g(x)=\sum_{i=2}^{n}(x_i-\frac{1}{\pi}|\arctan(\cot(3\pi t^2))|)^2$

where $G(t)=|\sin(0.5\pi t)|, p_t=\lfloor 6G(t)\rfloor$,
and the search space is $[0,1]$.

# 3.2 Dynamic Constrained Multiobjective Optimisation Problems

# DC1

$$min\begin{cases} f_1(x) = g(x)\cdot x_1 \\ f_2(x) = g(x)\sqrt{(1-x_1^2)} \end{cases}$$

$$s.t.\mathrm{c}=T_1\cdot T_2\le 0$$

$$T_1=3-G-\exp(f_1)-0.3\sin(4\pi f_1)-f_2$$

$$T_2=4.1-(1+f_1+0.3f_1^2)-0.3\sin(4\pi f_1)-f_2$$

With $g(x)=1+\sum_{i=2}^{n}x_i^2$
where $G=\cos(\pi t)\,, and\ the\ search\ space\ is\ \ x\,\epsilon[0,1]\times[-1,1]^{n-1}$.

# DC2

$$min\begin{cases} f_1(x) = g(x)(x_1+0.2Gsin(\pi x_1)) \\ f_2(x) = \ \ g(x)(1-x_1+0.2Gsin(\pi x_1)) \end{cases}$$

$$s.t.c_1=(f_1+2f_2-1)(f_1+0.5f_2-0.5)\ge 0$$

$$c_2=(f_1^2+f_2^2-1.4)\le 0$$

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

# DC3

$$min\begin{cases} f_1(x) = g(x)\cdot x_1 \\ f_2(x) = \ g(x)\sqrt{(1-x_1^2)} \end{cases}$$

$$s.t. c = 1.1 - \left((f_1/(0.9+(0.1+0.7|G|)W)\right)^2 - (f_2/(0.9+(0.8-0.7|G|)W))^2 \geq 0$$

$$W = cos(5arctan((f_2/f_1)^{\mathrm{H}})^4)^6$$

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

where $G = \sin(0.5\pi t)\,, H = \begin{cases} 1 & G \geq 0 \\ -1 & else \end{cases}$ *and the search space is* $x\,\epsilon[0,1]\times[-1,1]^{n-1}$.

# DC4

$$min\begin{cases} f_1(x) = g(x)\cdot x_1 \\ f_2(x) = \ g(x)(1-x_1) \end{cases}$$

$$s.t. c_1 = T_1 \cdot T_2 \geq 0$$

$$c_2 = T_3 \cdot T_4 \geq 0$$

$$T_1 = \frac{sinW - cosW}{sinW + cosW} f_1 - f_2 + \frac{-2.2(1-cosW)+1.3}{sinW + cosW}$$

$$T_2 = \frac{sinW - cosW}{sinW + cosW} f_1 - f_2 + \frac{-2.2(1-cosW)+1.8}{sinW + cosW}$$

$$T_3 = \frac{sinW - cosW}{sinW + cosW} f_1 - f_2 + \frac{-2.2(1-cosW)+2.6}{sinW + cosW}$$

$$T_1 = \frac{sinW - cosW}{sinW + cosW} f_1 - f_2 + \frac{-2.2(1-cosW)+3.1}{sinW + cosW}$$

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

where $G = \sin(0.5\pi t)\,, W = 0.5\pi t,$ *and the search space is* $x\,\epsilon[0,1]\times[-1,1]^{n-1}$

# DC5

$$min\begin{cases} f_1(x) = g(x)\cdot x_1 + G \\ f_2(x) = \ g(x)(1-x_1) + G \end{cases}$$

$$s.t. c = \cos(-0.15\pi)\, f_2 - \sin(-0.15\pi)\, f_1 \geq$$

$$(2\sin\,(5\pi(\sin(-0.15\pi) f_2 + cos(-0.15\pi) f_1)))^6$$

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

where $G = |\sin(0.5\pi t)|\,,$ *and the search space is* $x\,\epsilon[0,1]^n$.

# DC6

$$min\begin{cases} f_1(x) = g(x)\cdot(x_1 + 0.25Gsin(\pi x_1)) \\ f_2(x) = \ g(x)\big(1 - x_1 + 0.25Gsin(\pi x_1)\big) \end{cases}$$

$$s.t. c_1 = (4f_1 + f_2 - 1)(0.3f_1 + f_2 - 0.3) \geq 0$$

$$c_2 = (1.85 - f_1 - f_2 - (0.3\sin\,(3\pi(f_2 - f_1)))^2)(f_1 + f_2 - 1.3) \leq 0$$

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

where $G = \sin(0.5\pi t)$, *and the search space is* $x \epsilon [0,1] \times [-1,1]^{n-1}$.

# DC7

$$min \begin{cases} f_1(x) = g(x)(x_1 + 0.25\sin(\pi x_1)) \\ f_2(x) = g(x)(1 - x_1 + 0.25\sin(\pi x_1)) \end{cases}$$

$$s.t. c = (f_1^2 + f_2^2 - (1.3 - 0.45\sin(Garctan(f_2/f_1))^2)^2)$$

$$(f_1^2 + f_2^2 - (1.5 + 0.4\sin(4\arctan(f_2/f_1))^{16})^2) \geq 0$$

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

where $G = 2\lfloor 10|((\mathrm{t}+1)\mathrm{mod}\ 2) - 1|\ \rfloor$, *and the search space is* $x \epsilon [0,1]^n$.

# DC8

$$min \begin{cases} f_1(x) = g(x) \cdot x_1 \\ f_2(x) = g(x)(1 - x_1) \end{cases}$$

$$s.t. c_1 = ((0.2 + \mathrm{G})f_1^2 + f_2 - 2)(0.7f_1^2 + f_2 - 2.5) \geq 0$$

$$c_2 = f_1^2 + f_2^2 - (0.6 + \mathrm{G})^2 \geq 0$$

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

where $G = 0.5|\sin(0.5\pi t)|$, *and the search space is* $x \epsilon [0,1] \times [-1,1]^{n-1}$.

# DC9

$$min \begin{cases} f_1(x) = g(x)(x_1 + 0.02\sin((10 - |\lfloor 10\mathrm{G} \rfloor|)\pi x_1)) \\ f_2(x) = g(x)(1 - x_1 + 0.02\sin((10 - |\lfloor 10\mathrm{G} \rfloor|)\pi x_1)) \end{cases}$$

$$s.t. c = f_1 + f_2 - G - sin(5\,\pi(f_1 - f_2 + 1))^2 \geq 0$$

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

where $G = \sin(0.5\pi t)$, *and the search space is* $x \epsilon [0,1] \times [-1,1]^{n-1}$.

# DC10

$$min \begin{cases} f_1(x) = g(x) \cdot x_1 \\ f_2(x) = g(x)(1 - x_1) \end{cases}$$

$$s.t. c_1 = (\sqrt{f_1} + \sqrt{f_2} - 0.95 - 0.5|G|)(f_1^{1.5} + f_2^{1.5} - 1.2^{1.5}) \geq 0$$

$$c_2 = (0.8f_1 + f_2 - (2.5 + 0.08\sin(2\,\pi(f_2 - f_1))))$$

$$((0.93 + \frac{|G|}{3})f_1 + f_2 - (2.7 + 0.5|G| + 0.08\sin(2\pi(f_2 - f_1)))) \geq 0$$

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

where $G = \sin(0.5\pi t)$, *and the search space is* $x \epsilon [0,1] \times [-1,1]^{n-1}$.

# 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 ($\tau_t$): 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(30$\tau_t$+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 DP orDC

| Problem | $(\tau_t, n_t)$ | MIGD(mean(std.)) | MHV(mean(std.)) |
|---|---|---|---|
| DP1 | 10,5 | 1.234E-2(1.234E-3) | 1.234E-2(1.234E-3) |
| | 10,10 | | |
| | 20,5 | | |
| | 20,10 | | |
| DP2 | | | |
| …… | | | |
| DP10 | 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, Kaiser M, Yang S, et al. A Scalable Test Suite For Continuous Dynamic Multiobjective Optimization[J]. IEEE Transactions On Cybernetics, 2019, 50(6): 2814-2826.

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

[12] Guo Y, Chen G, Yue C, et al. Benchmark problems for CEC2023 competition on dynamic constrained multiobjective optimization[J]. Proc. CEC2023 Competition, 2023: 1-12.

[13] Hu Y, Zou J, Zheng J, et al. A new framework of change response for dynamic multi-objective optimization[J]. Expert Systems with Applications, 2024, 248: 123344.

[14] Lin Q, Ye Y, Ma L, et al. Dynamic multiobjective evolutionary optimization via knowledge transfer and maintenance[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2023, 54(2): 936-949.