Benchmark Problems for IEEE WCCI2024 Competition on Dynamic Constrained Multiobjective Optimization

Yinan Guo[1,2], Guoyu Chen[2], Caitong Yue[3], Jing Liang[3], Yong Wang[4], Shengxiang Yang[5]
1. School of Mechanical Electronic and Information Engineering, China University of Mining and Technology (Beijing), Beijing, China
2. School of Information and Control Engineering, China University of Mining and Technology, Xuzhou, China
3. School of Electrical Engineering, Zhengzhou University, Zhengzhou, China
4. School of Automation, Central South University, Changsha, China
5. School of Computer Science and Informatics, De Montfort University, Leicester LE1 9BH, U.K.

1. Introduction

In the past decade, dynamic constrained multiobjective optimization has attracted the increasing research interest [1][2]. To the best of our knowledge, the problem is widely-spread in real-world applications, such as scheduling optimization, and resource allocation, which involves time-varying multiobjective and constraints [3]-[5]. More especially, the corresponding dynamic constrained multiobjective optimization problems (DCMOPs) contain more complex characteristics and special difficulties than dynamic multiobjective optimization or constrained multiobjective optimization ones [6]-[9]. To solve this kind of problem, traditional multiobjective evolutionary algorithms mainly face three difficulties. First, environmental changes can be described as various dynamics, forming different levels of difficulties to algorithms. Thus, there is no change response strategy that can deal with all kinds of dynamics. Second, different types of constraints may appear under dynamic environments, which pose a challenge to achieve good versatility in handling various constraints for any static optimizer. Finally, the response time for environmental changes is generally tight for algorithms. Concerning the above-mentioned analysis, there is a significant need for new mechanisms in solving DCMOPs. More especially, a set of diverse and unbiased test problems is a great demand to systematically study dynamic constrained multiobjective evolutionary algorithms (DCMOEAs) in the field [10][11].

In the competition, 10 benchmark functions are developed, covering diverse characteristics which exactly represent different real-world scenarios, for example, continuity-disconnection, time-dependent PF/PS geometries, dynamic infeasible region, small feasible region, and so on. Based on the test suite with various characteristics, researchers can better understand the strengths and weaknesses of DCMOEAs, stimulating the research on dynamic constrained multiobjective optimization [12][13]. All the benchmark functions have been implemented in MATLAB code based on the codes provided by [14], which can be downloaded in the following website.
https://github.com/gychen94/DCMO

Table 1: Main characteristics of the 10 test problems

Problems	Objectives	Types	Remarks
DCF1	2	Static objectives and dynamic constraint	Dynamic constraints lead to that the true PF changes from the unconstrained PF to the partial boundary of feasible region.
DCF2	2	Dynamic objectives and static constraint	Under disconnected feasible regions, the switch of the position-related variable can significantly cause severe diversity loss to population.
DCF3	2	Dynamic objectives and constraint	Dynamic constraint can sometimes arise the convergence pressure of population, and sometimes shows no effect.
DCF4	2	Static objectives and dynamic constraint	In small feasible region, dynamic constraint changes the true PF from unconstrained PF to the boundary of feasible region.
DCF5	2	Dynamic objectives and static constraints	The mixed convexity-concavity of PF is affected by the boundary of feasible region under the small feasible region.
DCF6	2	Dynamic objectives and constraint	In small feasible region, the range and preference of disconnect PF change over time.
DCF7	2	Dynamic objectives and static constraint	PF shifts in the objective space over time, and changes from disconnect to continuity.
DCF8	2	Dynamic objectives and constraints	True PF is the boundary of feasible region, and oscillates over time.
DCF9	2	Dynamic objectives and constraints	The infeasible regions rotate along (1.1, 1.1) over time, changing from convergence pressure to distribution pressure.
DCF10	2	Dynamic objectives and constraint	Dynamic constraint makes the true PF that changes from several segments to continuous one of the unconstrained PF.

2. Summary of 10 test problems

The proposed test suite (called DCF in this competition) has 10 bi-objective problems. The main dynamic characteristics of each problem are briefly summarized in Table 1.

3. Problem definitions

The following definitions are widely used in each problem definition:

- m : the number of objectives
- n : the number of decision variables
- N : population size
- x_i : the *i*-th decision variable
- f_j : the *j*-th objective function
- c_k : the *k*-th function of constraint

- τ : generation counter
- τ_t : frequency of change
- n_t : severity of change
- $t = 1/n_t \cdot \lfloor \tau/\tau_t \rfloor$: time index

3.1 DCF1

$$\min \begin{cases} f_1 = g x_1 \\ f_2 = g(1-x_1) \end{cases}$$
$$\text{s.t.} \;\; c = f_1^2 + f_2^2 - (0.7+G)^2 \geq 0$$

with

$$g = 1 + \sum_{i=\{2,\ldots,n\}} (x_i - 0.5)^2$$

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

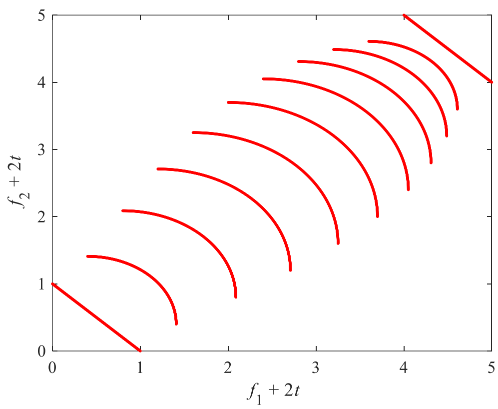

Fig. 1 Illustration of the true PF of DCF1.

Remark: DCF1 has the static objective functions and a dynamic constraint, forming the true PF changing from the unconstrained PF to the partial boundary of feasible region. This problem is used to assess the ability of algorithms in increasing diversity.

3.2 DCF2

$$\min \begin{cases} f_1 = g x_r \\ f_2 = g(1-x_r^2) \end{cases}$$
$$\text{s.t.} \;\; c = \cos(-0.15\pi) f_2 - \sin(-0.15\pi) f_1 - (2\sin(4\pi(\sin(-0.15\pi) f_2 + \cos(-0.15\pi) f_1)))^6 \leq 0$$

with

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

where $G=|\sin(0.5\pi t)|$, $r=1+\lfloor (n-1)G \rfloor$, and the search space is $x\in[0,1]^n$.

Remark: DCF2 has dynamic objective functions and a static constraint. Under disconnected feasible regions, the switch of the position-related variable can significantly cause severe diversity loss to population. Therefore, it is a significant challenge in maintaining or increasing diversity for solving this problem.

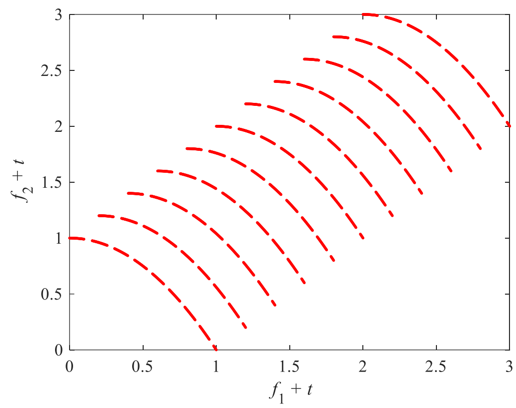

Fig. 2 Illustration of the true PF of DCF2.

3.3 DCF3

$$\min\begin{cases} f_1 = g x_1 \\ f_2 = g\sqrt{1-x_1^2} \end{cases}$$
$$\text{s.t. } c = T_1\cdot T_2 \geq 0$$
$$T_1 = f_1^2+f_2^2-2$$
$$T_2 = f_1^2+f_2^2-(2-(0.5-0.4G)\sin(10\arctan(f_2/f_1)))$$

with

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

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

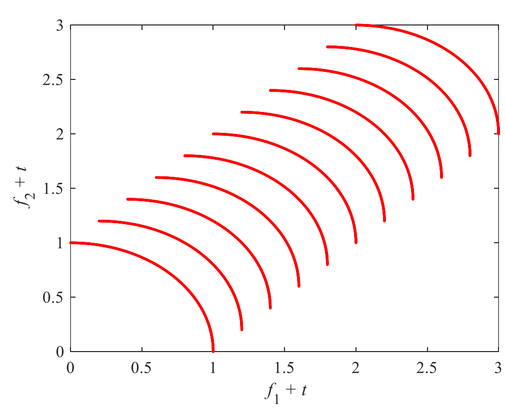

Fig. 3 Illustration of the true PF of DCF3.

Remark: DCF3 has simple dynamics on objective functions and constraint. Regarding the constraint, it sometimes arises the convergence pressure of population, and sometimes shows hardly no effect.

3.4 DCF4

$$\min\begin{cases} f_1 = g x_1 \\ f_2 = g\sqrt{1-x_1^2} \end{cases}$$
$$\text{s.t. } c = T_1 \cdot T_2 \le 0$$
$$T_1 = 3 - G - \exp(f_1) - 0.3\sin(4\pi f_1) - f_2$$
$$T_2 = 4.1 - (1 + f_1 + 0.3 f_1^2) - 0.3\sin(4\pi f_1) - f_2$$

with

$$g = 1 + \sum_{i=\{2,\ldots,n\}} x_i^2$$

where $G = \cos(\pi t)$, and the search space is $x \in [0,1] \times [\text{-}1,1]^{n-1}$.

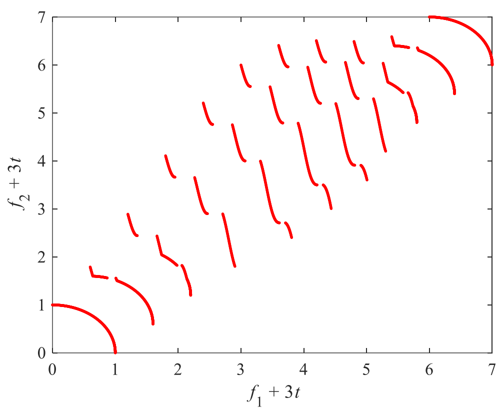

Fig. 4 Illustration of the true PF of DCF4.

Remark: DCF4 shows dynamics on the small feasible region. The true PF changes from unconstrained PF to the boundary of feasible region, and the smaller feasible region makes it difficult to track.

3.5 DCF5

$$\min\begin{cases} f_1 = g(x_1 + 0.2G\sin(\pi x_1)) \\ f_2 = g(1 - x_1 + 0.2G\sin(\pi x_1)) \end{cases}$$
$$\text{s.t. } c_1 = \left(f_1 + 2f_2 - 1\right)\left(f_1 + 0.5f_2 - 0.5\right) \ge 0$$
$$c_2 = \left(f_1^2 + f_2^2 - 1.4\right) \le 0$$

with

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

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

Remark: DCF5 has the mixed and small feasible region, and the unconstrained PF geometry changes from concavity to convexity, or vice versa. To be more specific, the true PF is sometimes affected by the boundary of feasible region.

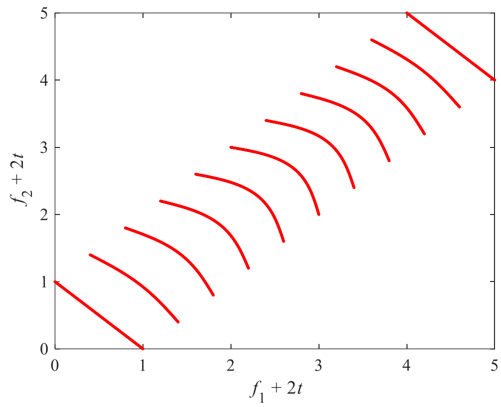

Fig. 5 Illustration of the true PF of DCF5.

3.6 DCF6

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

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

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

with

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

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

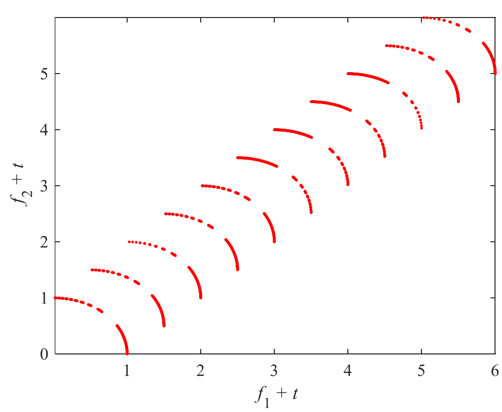

Fig. 6 Illustration of the true PF of DCF6.

Remark: DCF6 has both of dynamic objective functions and constraint, and the constraint changes the objective preference of PF over time. Therefore, problem is used to see what performance of algorithms once they are confronted with the dynamic range and bias of disconnect PF.

3.7 DCF7

$$\min \begin{cases} f_1 = gx_1 + |G| \\ f_2 = g(1-x_1) + |G| \end{cases}$$
$$\text{s.t. } c = f_1 + f_2 - 1 - |\sin(5\pi(f_1 - f_2 + 1))| \geq 0$$

with

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

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

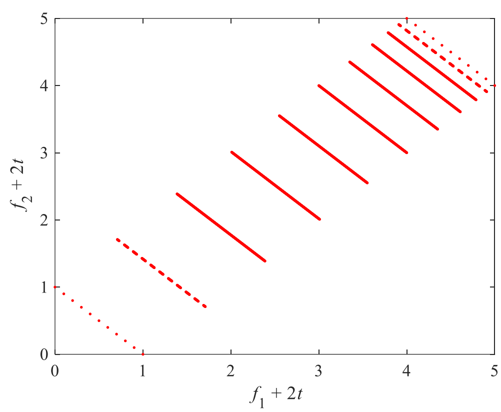

Fig. 7 Illustration of the true PF of DCF7.

Remark: The constraint in DCF7 is always static, and unconstrained PF moves in objective space over time. More especially, the true PF changes from disconnect to continuity over time.

3.8 DCF8

$$\min \begin{cases} f_1 = gx_1 \\ f_2 = g(1-x_1) \end{cases}$$
$$\text{s.t. } c_1 = f_1 + f_2 - 1.2 - 0.03\sin(W\pi f_1) \geq 0$$
$$c_2 = (f_1 + f_2 - 3.5)(f_1 + f_2 - 2.2 - G^2) \geq 0$$

with

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

where $G = \sin(0.5\pi t)$, $W = \lfloor 10G \rfloor$, and the search space is $x \in [0,1] \times [-1,1]^{n-1}$.

Remark: The true PF of DCF8 is always the boundary of feasible region, and oscillates over time. In addition, the convergence pressure is time-varying caused by dynamic constraints.

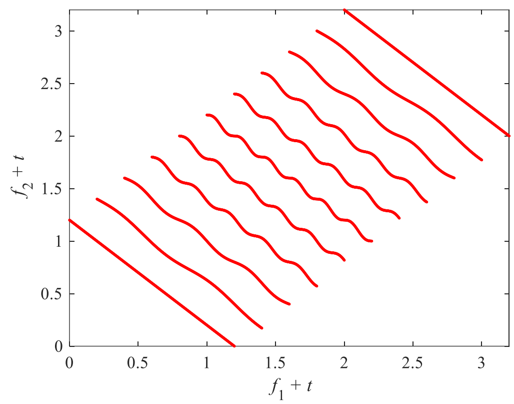

Fig. 8 Illustration of the true PF of DCF8.

3.9 DCF9

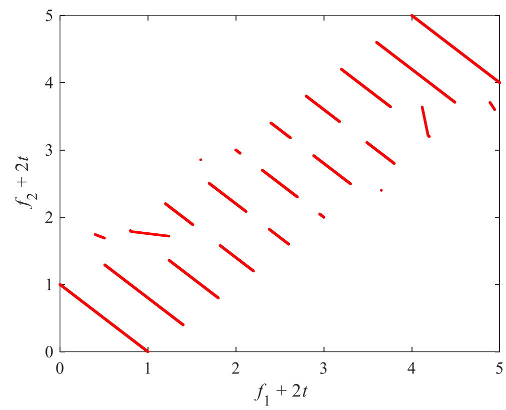

Fig. 9 Illustration of the true PF of DCF9.

$$
\begin{aligned}
&\min \begin{cases} f_1 = g(x_1) \\ f_2 = g(1-x_1) \end{cases} \\
&\text{s.t. } \; c_1 = T_1 \cdot T_2 \geq 0 \\
&\qquad c_2 = T_3 \cdot T_4 \geq 0 \\
&\qquad T_1 = \frac{\sin W - \cos W}{\sin W + \cos W} f_1 - f_2 + \frac{-2.2(1-\cos W)+1.3}{\sin W + \cos W} \\
&\qquad T_2 = \frac{\sin W - \cos W}{\sin W + \cos W} f_1 - f_2 + \frac{-2.2(1-\cos W)+1.8}{\sin W + \cos W} \\
&\qquad T_3 = \frac{\sin W - \cos W}{\sin W + \cos W} f_1 - f_2 + \frac{-2.2(1-\cos W)+2.6}{\sin W + \cos W} \\
&\qquad T_4 = \frac{\sin W - \cos W}{\sin W + \cos W} f_1 - f_2 + \frac{-2.2(1-\cos W)+3.1}{\sin W + \cos W}
\end{aligned}
$$

with

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

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

Remark: The objective functions of DCF9 are rather simple, and the infeasible regions rotate along (1.1, 1.1) over time, changing the pressure in managing convergence to distribution.

3.10 DCF10

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

$$\text{s.t. } c = T_1 \cdot T_2 \geq 0$$

$$T_1 = f_1^2 + f_2^2 - (1.4 + 0.5|G| + (1 - |G|)\sin(8\arctan(f_2 / f_1))^{12})^2$$

$$T_2 = f_1^2 + f_2^2 - (1.4 - (1 - |G|)\sin(8\arctan(f_2 / f_1))^{12})^2$$

with

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

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

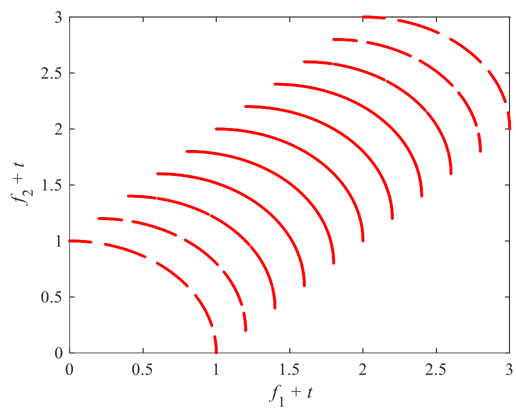

Fig. 10 Illustration of the true PF of DCF10.

Remark: The objective functions and constraints of DCF10 varies over time, and dynamic constraint makes the true PF that changes from several segments to continuous one of the unconstrained PF. This problem is used to assess the solving ability of diversity or convergence

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), 30 (slow changing environments).
- Severity of change (n_t): 5 (severe changing environments), 10 (moderate changing environments).
- Number of changes: 60.
- Stopping criterion: a maximum number of $100(60\tau_t + 60)$ fitness evaluations, where 6000 fitness evaluations are given before the first environmental change occurs.
- Number of independent runs: 20.

4.2 Performance metric

1) MIGD:

MIGD is the modified IGD [15], which can comprehensively reflect the diversity and convergence of Pareto-optimal solutions obtained under the total environments. Assuming that a set of uniformly distributed samples of true PF(t) are denoted as $R(t)$, the MIGD value is calculated as follows.

$$\text{MIGD} = \frac{1}{T}\sum_{t=1}^{T}\text{IGD}(P(t), R(t))$$

with

$$\text{IGD}(\text{P}(t), R(t)) = \frac{\sum_{i=1}^{|R(t)|} dis(P(t), r_i)}{|R(t)|}$$

where T is the number of environments that occurred in a run. $dis(P(t), r_i)$ denotes the minimum Euclidean distance in the objective space between ith point in $R(t)$ and the $P(t)$ obtained by an algorithm. Here, a set of around 1000 points uniformly sampled from the true PF(t) is expected to use for the calculation of MIGD.

2) MHV:

The MHV is a modification of the HV [16] that computes the hypervolume of the area dominated by the obtained $P(t)$:

$$\text{MHV} = \frac{1}{T}\sum_{t=1}^{T}\text{HV}(P(t))$$

with

$$\text{HV}(P(t)) = \bigcup_{i=1}^{|P(t)|}\{\text{vol}_i\}$$

where vol_i denotes the volume of a hypercube formed by reference point z_{ref} and ith solution in $P(t)$. Before the calculation, all the objective values of $P(t)$ are normalized by the nadir point and ideal point of the true PF(t), and the reference point z_{ref} is set to (1.1, …, 1.1).

Table 2: MIGD and MHV results obtained by your algorithm on DCF

Problem	(τ_t, n_t)	MIGD (mean(std.))	MHV (mean(std.))
DCF1	10, 5	1.1111E-2(1.1111E-3)	1.1111E-2(1.1111E-3)
	10, 10		
	30, 5		
	30, 10		
…			
DCF10	10, 5		
	10, 10		
	30, 5		
	30, 10		

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.

References

[1] S. Biswas, S. Das, P. N. Suganthan, and C. A. C. Coello, "Evolutionary multiobjective optimization in dynamic environments: A set of novel benchmark functions," in *Proc. 2014 IEEE Congr. Evol. Comput.*, 2014, pp. 3192–3199.

[2] G. Chen, Y. Guo, M. Huang, D. Gong, and Z. Yu, "A domain adaptation learning strategy for dynamic multiobjective optimization," *Inf. Sci.*, vol. 606, no. 4, pp. 328–349, 2022.

[3] R. Azzouz, S. Bechikh, and L. Ben Said, "Dynamic Multi-objective Optimization Using Evolutionary Algorithms: A Survey," in *Adaptation, Learning, and Optimization*, 2017, pp. 31–70.

[4] S. Jiang and S. Yang, "Evolutionary dynamic multi-objective optimization: Benchmarks and algorithm comparisons," *IEEE Trans. Cybern.*, vol. 47, no. 1, pp. 198–211, 2017.

[5] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable Test Problems for Evolutionary Multiobjective Optimization," in *Evolutionary Multiobjective Optimization*, London: Springer-Verlag, 2005, pp. 105–145.

[6] R. Azzouz, S. Bechikh, and L. Ben Said, "Multi-objective Optimization with Dynamic Constraints and Objectives," in *Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation*, Jul. 2015, 615–622.

[7] J.-J. Ji, Y.-N. Guo, X.-Z. Gao, D.-W. Gong, and Y.-P. Wang, "Q-Learning-Based Hyperheuristic Evolutionary

Algorithm for Dynamic Task Allocation of Crowdsensing," *IEEE Trans. Cybern.*, pp. 1–14, 2021.

[8] Z. Ma and Y. Wang, "Evolutionary Constrained Multiobjective Optimization: Test Suite Construction and Performance Comparisons," *IEEE Trans. Evol. Comput.*, vol. 23, no. 6, pp. 972–986, 2019.

[9] T. T. Nguyen and X. Yao, "Continuous Dynamic Constrained Optimization—The Challenges," *IEEE Trans. Evol. Comput.*, vol. 16, no. 6, pp. 769–786, 2012.

[10] T. T. Nguyen and Xin Yao, "Benchmarking and solving dynamic constrained problems," in *2009 IEEE Congress on Evolutionary Computation*, 2009, pp. 690–697.

[11] R. Azzouz, S. Bechikh, L. Ben Said, and W. Trabelsi, "Handling time-varying constraints and objectives in dynamic evolutionary multi-objective optimization," *Swarm Evol. Comput.*, vol. 39, pp. 222–248, 2018.

[12] Q. Chen, J. Ding, S. Yang, and T. Chai, "A Novel Evolutionary Algorithm for Dynamic Constrained Multiobjective Optimization Problems," *IEEE Trans. Evol. Comput.*, vol. 24, no. 4, pp. 792–806, 2020.

[13] S. Jiang, S. Yang, X. Yao, K. TAN, M. Kaiser, and N. Krasnogor, "Benchmark problems for cec2018 competition on dynamic multiobjective optimisation," *CEC2018 Competition*, 2018.

[14] Y. Tian, R. Cheng, X. Zhang, and Y. Jin, "PlatEMO: A MATLAB Platform for Evolutionary Multi-Objective Optimization [Educational Forum]," *IEEE Comput. Intell. Mag.*, vol. 12, no. 4, pp. 73–87, 2017.

[15] Aimin Zhou, Yaochu Jin, and Qingfu Zhang, "A Population Prediction Strategy for Evolutionary Dynamic Multiobjective Optimization," *IEEE Trans. Cybern.*, vol. 44, no. 1, pp. 40–53, 2014.

[16] L. While, P. Hingston, L. Barone, and S. Huband, "A faster algorithm for calculating hypervolume," *IEEE Trans. Evol. Comput.*, vol. 10, no. 1, pp. 29–38, 2006.