IEEE WCCI/CEC2020 Competition on Constrained Multiobjective Optimization

Yong Wang ywang@csu.edu.cn
School of Automation, Central South University, Changsha, 410083, China

Zhi-Zhong Liu liuzz3@sustech.edu.cn
Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, 518055, China

Zhongwei Ma mzw_cemo@csu.edu.cn
School of Automation, Central South University, Changsha, 410083, China

Bing-Chuan Wang bcwang@csu.edu.cn
School of Automation, Central South University, Changsha, 410083, China

1 Scope and Topics

Constrained multiobjective optimization problems (CMOPs) are frequently encountered in diverse science/engineering disciplines [1, 2, 3, 4], which can be mathematically defined as:

$$\begin{aligned} \min \quad & \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_m(\mathbf{x}))^T \in \mathbb{F} \\ \text{s.t.} \quad & g_j(\mathbf{x}) \leq 0,\ j = 1, ..., l \\ & h_j(\mathbf{x}) = 0,\ j = l+1, ..., n \\ & \mathbf{x} = (x_1, x_2, ..., x_D)^T \in \mathbb{S} \end{aligned} \tag{1}$$

where $\mathbf{x}$ is a D-dimensional decision vector, $\mathbb{S}$ is the decision space, $\mathbf{F}(\mathbf{x})$ denotes the objective vector including m real-valued objective functions[1], $\mathbb{F}$ refers to the objective space, $f_i(\mathbf{x})$ is the ith objective function, $g_j(\mathbf{x})$ denotes the jth inequality constraint, $h_j(\mathbf{x})$ refers to the $(j-l)$th equality constraint, and l and $(n-l)$ are the numbers of inequality and equality constraints, respectively. CMOPs contain both conflicting objective functions and diverse constraints. Therefore, to properly address a CMOP, not only the tradeoff among objective functions but also the balance between objective functions and constraints should be carefully considered [5]. Undoubtedly, this is not an easy task for current evolutionary algorithms (EAs).

Fortunately, many researchers in the community of evolutionary computation have been devoting to coping with CMOPs; thus, many constrained multiobjective EAs (CMOEAs) have already been proposed during the past two decades [6]. In practice, these CMOEAs are roughly grouped into three classes, i.e., dominance-based CMOEAs, decomposition-based CMOEAs, and indicator-based CMOEAs, in which constraint-handling techniques are combined with dominance-based, decomposition-based, and indicator-based MOEAs, respectively [5, 7, 8]. To assess the performance of these CMOEAs, various artificial CMOPs have been designed including CTP [9], C-DTLZ [10], DAS-CMOPs [11], and NCTPs [12]. These artificial CMOPs are more suitable than real-world ones as benchmark functions. It is because the latter may

[1] Among these m objectives, at least two of them are conflicting with each other

require special hardware or software [13] in the simulation processes. Anyway, these artificial CMOPs can help researchers to analyze and understand the performance of different CMOEAs and encourage users to select the desired ones. At present, most of them have been successfully solved by current peer CMOEAs [5, 14, 15, 16].

To further boost the development of CMOEAs, two novel CMOP test suites (i.e., DOC [17] and MW [18]) were proposed in 2019 and both of them draw the inspirations from the CMOPs in the real-life applications.

- DOC [17] considers the fact that, in the real-world CMOPs, both decision and objective constraints are involved, which are easy to understand in decision space and objective space, respectively. To simulate the real-world scenes better, DOC provides a quite convenient approach to construct a CMOP with both decision and objective constraints. Note that these decision constraints can make the feasible region in the decision space have different properties (e.g., nonlinear, extremely small, and multimodal), while these objective constraints can reduce the feasible region in the objective space and make the Pareto front (PF) have diverse characteristics (e.g., continuous, discrete, mixed, and degenerate). As a result, DOC poses a great challenge to the current CMOEAs to obtain a set of well-distributed and well-converged feasible solutions.

- Due to the presence of constraints, some or all the original Pareto optimal solutions of the unconstrained MOPs may become infeasible, and some solutions on the boundary of the feasible region of a CMOP may become the Pareto optimal solutions. With that in mind, MW [18] designs four different types of CMOPs: 1) Type I—the constrained PF is the same with the unconstrained PF; 2) Type II—the constrained PF is a part of the unconstrained PF; 3) Type III—the constrained PF consists of a part of the unconstrained PF and a part of the boundary of the feasible region; and 4) Type IV—the unconstrained PF is all located outside the feasible region. All these four different types of CMOPs are constructed by making use of a new construction method, in which a global control process and a local adjustment process are included. The CMOPs in MW have complex geometries of constrained Pareto front and controllable size of feasible region in the objective space, which can also set up a lot of obstacles for a CMOEA to obtain promising results.

For convenience, we do not design any new CMOPs, instead, we select several CMOPs in these two test suites as the benchmark test functions for this competition. Specifically, CMOPs 1-6 come from MW5, MW6, MW10, MW13, MW9, and MW11, and CMOPs 7-10 come from DOC-1, DOC-4, DOC-5, and DOC-8. It is believed that the blending of these two test suites has the capability to assess an algorithm's capability for solving CMOPs in the real-life applications.

2 Submission Deadline

All submissions should be sent until March 31st, 23:59 (GMT).

3 Submission Instructions

Please download the source code of the test functions (http://www.escience.cn/people/yongwang1/competition2020.html) and evaluate your algorithms following the performance criteria. Please send the source code of your algorithm, the description of your algorithm, and the experimental results to one of the organizers.

4 Test Instances

4.1 CMOP-1

- *Objective functions:*

$$\min \begin{cases} f_1(\mathbf{x}) = gx_1 \\ f_2(\mathbf{x}) = g\sqrt{1-(f_1/g)^2} \end{cases}$$

where $g = 1 + \sum_{i=2}^{n}\big(1 - \exp\big(-10\big(z_i - 0.5 - \frac{i-1}{2n}\big)^2\big)\big)$, $z_i = x_i^{n-2}$, and n is the number of decision variables and is set to 25.

- *Objective constraints:*

$$\begin{aligned} &C_1 : f_1^2 + f_2^2 - \big(1.7 - 0.2\sin(2l)\big)^2 \le 0, \\ &C_2 : \Big(1 + 0.5\sin\big(6(0.5\pi - 2|l - 0.25\pi|)^3\big)\Big)^2 - f_1^2 - f_2^2 \le 0, \\ &C_3 : \Big(1 - 0.45\sin\big(6(0.5\pi - 2|l - 0.25\pi|)^3\big)\Big)^2 - f_1^2 - f_2^2 \le 0, \end{aligned}$$

where $l = \arctan(f_2/f_1)$.

- The search space is: $0 \le x_i \le 1,\ i = 1, 2, ..., 25$.
- Its Pareto front is illustrated in Fig. 1.

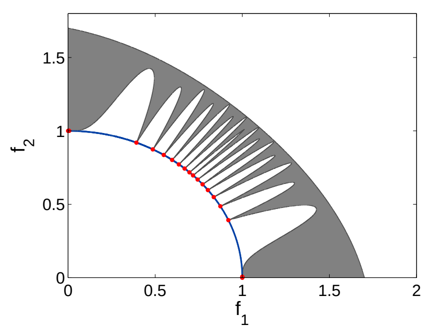

Figure 1: Pareto front of CMOP-1.

4.2 CMOP-2

- *Objective functions:*

$$\min \begin{cases} f_1(\mathbf{x}) = gx_1 \\ f_2(\mathbf{x}) = g\sqrt{1.1^2-(f_1/g)^2} \end{cases}$$

where $g = 1 + \sum_{i=2}^{n}\big(1.5 + \frac{0.1}{n}z_i^2 - 1.5\cos(2\pi z_i)\big)$, $z_i = 1 - \exp\big(-10\big(x_i - \frac{i-1}{n}\big)^2\big)$, and n is the number of decision variables and is set to 25.

- *Objective constraint:*

$$C_1 : \big(f_1/(1+0.15l)\big)^2 + \big(f_2/(1+0.75l)\big)^2 - 1 \leq 0,$$

  where $l = \cos\big(6\arctan(f_2/f_1)^4\big)^{10}$.

- The search space is: $0 \leq x_i \leq 1.1,\ i = 1, 2, ..., 25.$
- Its Pareto front is illustrated in Fig. 2.

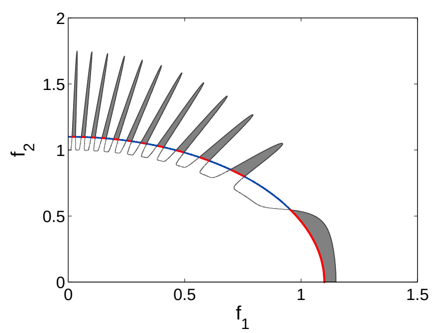

Figure 2: Pareto front of CMOP-2.

4.3 CMOP-3

- *Objective functions:*

$$\min \begin{cases} f_1(\mathbf{x}) = g x_1^n \\ f_2(\mathbf{x}) = g\big(1 - (f_1/g)^2\big) \end{cases}$$

  where $g = 1 + \sum_{i=2}^{n} \big(1.5 + \frac{0.1}{n} z_i^2 - 1.5\cos(2\pi z_i)\big)$, $z_i = 1 - \exp\big(-10\big(x_i - \frac{i-1}{n}\big)^2\big)$, and n is the number of decision variables and is set to 25.

- *Objective constraints:*

$$\begin{aligned} &C_1 : (2 - 4f_1^2 - f_2)(2 - 8f_1^2 - f_2) \geq 0, \\ &C_2 : (2 - 2f_1^2 - f_2)(2 - 16f_1^2 - f_2) \leq 0, \\ &C_3 : (1 - f_1^2 - f_2)(1.2 - 1.2f_1^2 - f_2) \leq 0. \end{aligned}$$

- The search space is: $0 \leq x_i \leq 1,\ i = 1, 2, ..., 25.$
- Its Pareto front is illustrated in Fig. 3.

4.4 CMOP-4

- *Objective functions:*

$$\min \begin{cases} f_1(\mathbf{x}) = g x_1 \\ f_2(\mathbf{x}) = g\big(5 - \exp(f_1/g) - 0.5\big|\sin(3\pi f_1/g)\big|\big) \end{cases}$$

  where $g = 1 + \sum_{i=2}^{n} \big(1.5 + \frac{0.1}{n} z_i^2 - 1.5\cos(2\pi z_i)\big)$, $z_i = 1 - \exp\big(-10\big(x_i - \frac{i-1}{n}\big)^2\big)$, and n is the number of decision variables and is set to 25.

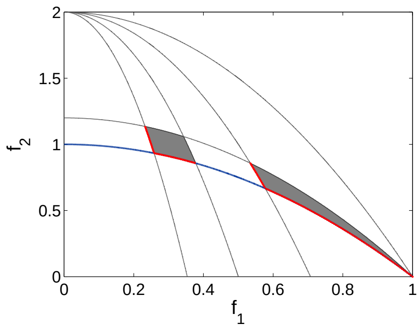

Figure 3: Pareto front of CMOP-3.

- *Objective constraints:*

$$C_1 : \big(5 - \exp(f_1) - 0.5\sin(3\pi f_1) - f_2\big)\big(5 - (1 + 0.4f_1) - 0.5\sin(3\pi f_1) - f_2\big) \le 0,$$
$$C_2 : \big(5 - (1 + f_1 + 0.5f_1^2) - 0.5\sin(3\pi f_1) - f_2\big)\big(5 - (1 + 0.7f_1) - 0.5\sin(3\pi f_1) - f_2\big) \ge 0.$$

- The search space is: $0 \le x_i \le 1.5,\ i = 1, 2, ..., 25.$
- Its Pareto front is illustrated in Fig. 4.

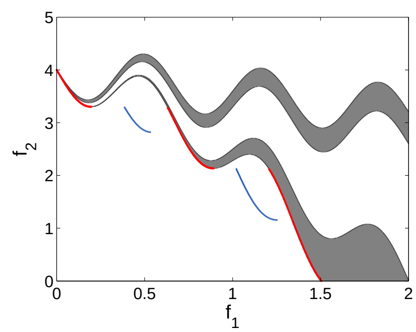

Figure 4: Pareto front of CMOP-4.

4.5 CMOP-5

- *Objective functions:*

$$\min \begin{cases} f_1(\mathbf{x}) = gx_1 \\ f_2(\mathbf{x}) = g\big(1-(f_1/g)^{0.6}\big) \end{cases}$$

where $g = 1 + \sum_{i=2}^{n}\big(1 - \exp\big(-10\big(z_i - 0.5 - \frac{i-1}{2n}\big)^2\big)\big)$, $z_i = x_i^{n-2}$ $(i = 2, ..., n)$, and n is the number of decision variables and is set to 25.

- *Objective constraint:*

$$C_1 : \min\{T_1, T_2\} \le 0,$$

where $T_1 = \big(1 - 0.64f_1^2 - f_2\big)\big(1 - 0.36f_1^2 - f_2\big)$ and $T_2 = \big(1.35^2 - (f_1 + 0.35)^2 - f_2\big)\big(1.15^2 - (f_1 + 0.15)^2 - f_2\big)$.

- The search space is: $0 \leq x_i \leq 1,\ i = 1, 2, ..., 25.$
- Its Pareto front is illustrated in Fig. 5.

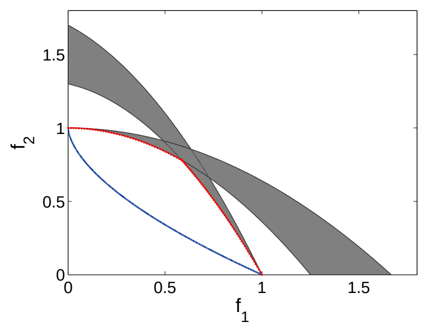

Figure 5: Pareto front of CMOP-5.

4.6 CMOP-6

- *Objective functions:*

$$\min \begin{cases} f_1(\mathbf{x}) = gx_1 \\ f_2(\mathbf{x}) = g\sqrt{2 - (f_1/g)^2} \end{cases}$$

where $g = 1 + 2\sum_{i=2}^{n}\big(x_i + (x_{i-1} - 0.5)^2 - 1\big)^2$, and n is the number of decision variables and is set to 25.

- *Objective constraints:*

$$\begin{aligned} &C_1 : (3 - f_1^2 - f_2)(3 - 2f_1^2 - f_2) \geq 0, \\ &C_2 : (3 - 0.625f_1^2 - f_2)(3 - 7f_1^2 - f_2) \leq 0, \\ &C_3 : (1.62 - 0.18f_1^2 - f_2)(1.125 - 0.125f_1^2 - f_2) \geq 0, \\ &C_4 : (2.07 - 0.23f_1^2 - f_2)(0.63 - 0.07f_1^2 - f_2) \leq 0. \end{aligned}$$

- The search space is: $0 \leq x_i \leq \sqrt{2},\ i = 1, 2, ..., 25.$
- Its Pareto front is illustrated in Fig. 6.

4.7 CMOP-7

- *Objective functions:*

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

where $g = 5.3578547x_4^2 + 0.8356891x_2x_6 + 37.293239x_2 - 10125.6023282166.$

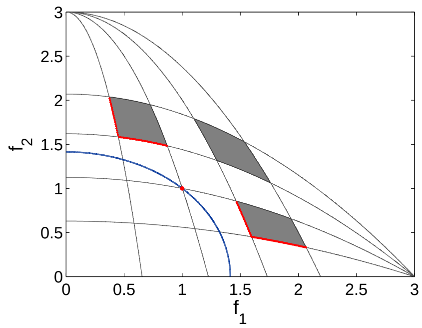

Figure 6: Pareto front of CMOP-6.

- *Objective constraint:*

$$C_1 : f_1^2 + f_2^2 - 1 \geq 0. \tag{2}$$

- *Decision constraints:*

$$\begin{aligned}
&C_2 : 85.334407 + 0.0056858x_3x_6 + 0.0006262x_2x_5 - 0.0022053x_4x_6 \leq 92;\\
&C_3 : -85.334407 - 0.0056858x_3x_6 - 0.0006262x_2x_5 + 0.0022053x_4x_6 \leq 0;\\
&C_4 : 80.51249 + 0.0071317x_3x_6 + 0.0029955x_2x_3 + 0.0021813x_4^2 \leq 110;\\
&C_5 : -80.51249 - 0.0071317x_3x_6 - 0.0029955x_2x_3 - 0.0021813x_4^2 \leq -90;\\
&C_6 : 9.300961 + 0.0047026x_4x_6 + 0.0012547x_2x_4 + 0.0019085x_4x_5 \leq 25;\\
&C_7 : -9.300961 - 0.0047026x_4x_6 - 0.0012547x_2x_4 - 0.0019085x_4x_5 \leq -20.
\end{aligned}$$

- The search space is: $0 \leq x_1 \leq 1$, $78 \leq x_2 \leq 102$, $33 \leq x_3 \leq 45$, and $27 \leq x_4, x_5, x_6 \leq 45$.

- Its Pareto front is illustrated in Fig. 7.

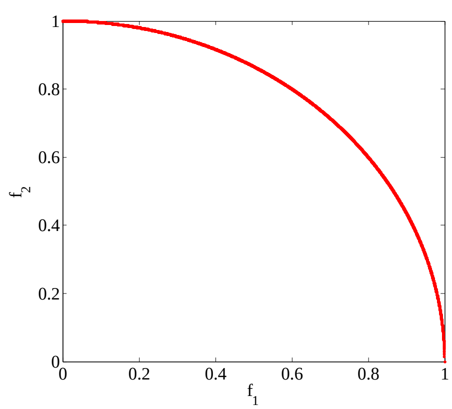

Figure 7: Pareto front of CMOP-7.

4.8 CMOP-8

- *Objective functions:*

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

where $g = (x_2 - 10)^2 + 5(x_3 - 12)^2 + x_4^4 + 3(x_5 - 11)^2 + 10x_6^6 + 7x_7^2 + x_8^4 - 4x_7x_8 - 10x_7 - 8x_8 - 679.6300573745$

- *Objective constraints:*

$$C_1 = f_1 + f_2 - 1 \geq 0;$$
$$C_2 = f_1 + f_2 - 1 - |\sin(10\pi(f_1 - f_2 + 1))| \geq 0.$$

- *Decision constraints:*

$$C_3 = -127 + 2x_2^2 + 3x_3^4 + x_4 + 4x_5^2 + 5x_6 \leq 0;$$
$$C_4 = -282 + 7x_2 + 3x_3 + 10x_4^2 + x_5 - x_6 \leq 0;$$
$$C_5 = -196 + 23x_2 + x_3^2 + 6x_7^2 - 8x_8 \leq 0;$$
$$C_6 = 4x_2^2 + x_3^2 - 3x_2x_3 + 2x_4^2 + 5x_7 - 11x_8 \leq 0.$$

- The search space is: $0 \leq x_1 \leq 1$, and $-10 \leq x_i \leq 10,\ i = 2, 3, ..., 8$.
- Its Pareto front is illustrated in Fig. 8.

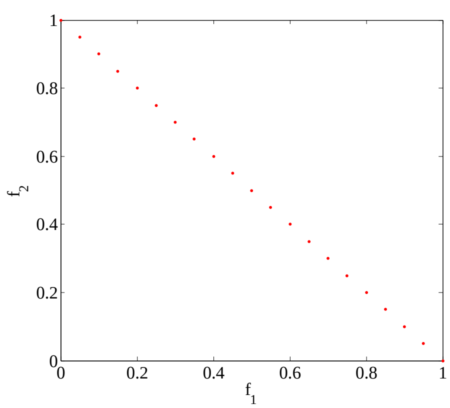

Figure 8: Pareto front of CMOP-8.

4.9 CMOP-9

- *Objective functions:*

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

where $g = x_2 - 192.724510070035$.

- *Objective constraints:*

$$
\begin{aligned}
&C_1 : f_1 + f_2 - 1 \geq 0;\\
&C_2 : f_1 + f_2 - 1 - |sin(10\pi(f_1 - f_2 + 1))| \geq 0;\\
&C_3 : (f_1 - 0.8)(f_2 - 0.6) \leq 0.
\end{aligned}
$$

- *Decision constraints:*

$$
\begin{aligned}
&C_4 : -x_2 + 35x_3^{0.6} + 35x_4^{0.6} \leq 0;\\
&C_5 : -300x_4 + 7500x_6 - 7500x_7 - 25x_5x_6 + 25x_5x_7 + x_4x_5 = 0;\\
&C_6 : 100x_3 + 155.365x_5 + 2500x_8 - x_3x_5 - 25x_5x_8 - 15536.5 = 0;\\
&C_7 : -x_6 + \log(-x_5 + 900) = 0;\\
&C_8 : -x_7 + \log(x_5 + 300) = 0;\\
&C_9 : -x_8 + \log(-2x_5 + 700) = 0.
\end{aligned}
$$

- The search space is: $0 \leq x_1 \leq 1$, $0 \leq x_2 \leq 1000$, $0 \leq x_3, x_4 \leq 40$, $100 \leq x_5 \leq 300$, $6.3 \leq x_6 \leq 6.7$, $5.9 \leq x_7 \leq 6.4$, and $4.5 \leq x_8 \leq 6.25$.

- Its Pareto front is illustrated in Fig. 9.

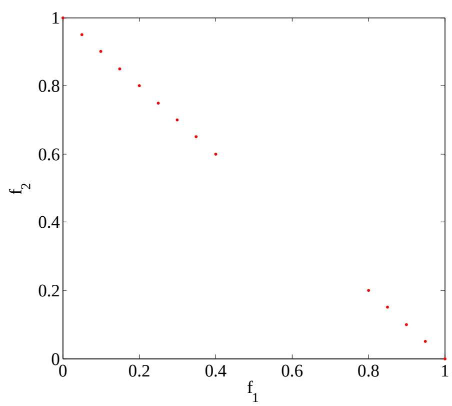

Figure 9: Pareto front of CMOP-9.

4.10 CMOP-10

- *Objective functions:*

$$
\min \begin{cases} f_1(\mathbf{x}) = gx_1x_2 \\ f_2(\mathbf{x}) = gx_1(1 - x_2) \\ f_3(\mathbf{x}) = g(1 - x_1) \end{cases}
$$

where $g = x_3 + x_4 + x_5 - 7048.2480205286$.

- *Objective constraints:*

$$C_1 : (f_3 - 0.4)(f_3 - 0.6) \geq 0.$$

- *Decision constraints:*

$$
\begin{aligned}
&C_2 : -1 + 0.0025(x_6 + x_8) \le 0;\\
&C_3 : -1 + 0.0025(x_7 + x_9 - x_6) \le 0;\\
&C_4 : -1 + 0.01(x_{10} - x_7) \le 0;\\
&C_5 : -x_3x_8 + 833.33252x_6 + 100x_3 - 83333.333 \le 0;\\
&C_6 : -x_4x_9 + 1250x_7 + x_4x_6 - 1250x_6 \le 0;\\
&C_7 : -x_5x_{10} + 1250000 + x_5x_7 - 2500x_7 \le 0.
\end{aligned}
$$

- The search space is: $0 \le x_1 \le 1$, $0 \le x_2 \le 1$, $500 \le x_3 \le 1000$, $1000 \le x_4 \le 2000$, $5000 \le x_5 \le 6000$, and $100 \le x_i \le 500, i = 6, 7, ..., 10$.

- Its Pareto front is illustrated in Fig. 10.

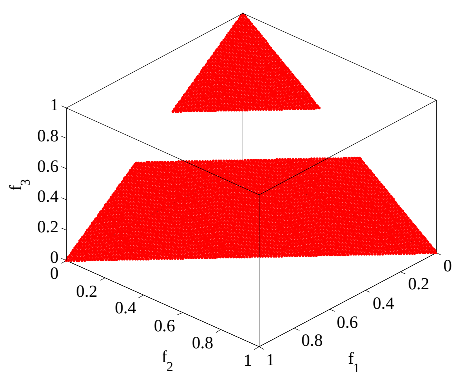

Figure 10: Pareto front of CMOP-10.

5 Evaluation Criteria

For each instance, 30 independent runs should be implemented. For CMOP-1-6 and CMOP-7-10, the maximum number of fitness evaluations is set to 100,000 and 200,000, respectively. For each algorithm, the population size is set to 100.

According to the suggestion in [17], three evaluation criteria are used to compare the performance of different CMOEAs:

- *Feability Rate (FR):* Suppose that $FeasibleRuns$ denotes the number of runs where a CMOEA can find at least one feasible solution in the final population, and $TotalRuns$ denotes the number of total runs. Then, FR is defined as:

$$FR = \frac{FeasibleRuns}{TotalRuns}. \tag{3}$$

  The value of FR ranges from 0% to 100%, and the larger the value of FR, the higher the probability of a CMOEA to enter the feasible region.

- *Inverted Generational Distance (IGD) [19]:* IGD has been widely used to evaluate a MOEA's performance. However, IGD may lose its effectiveness to evaluate a CMOEA owing to the existence of the infeasible solutions. Herein, we only keep the feasible solutions and compute the IGD value of them. Specifically, suppose that $\mathcal{P}$ is the set of images of the feasible solutions, and $\mathcal{P}^*$ is a set of nondominated points uniformly distributed on the PF. Then, the IGD metric is calculated as:

$$IGD(\mathcal{P}) = \frac{1}{|\mathcal{P}^*|} \sum_{z^* \in \mathcal{P}^*} distance(z^*, \mathcal{P}), \tag{4}$$

where $distance(z^*, \mathcal{P})$ is the minimum Euclidean distance between z^* and all the feasible solutions in $\mathcal{P}$, and $|\mathcal{P}^*|$ is the cardinality of $\mathcal{P}^*$. The smaller the IGD value, the better the performance of a CMOEA.

- *Hypervolume (HV) [20]:* Similarly, the infeasible solutions should be deleted before the calculation of HV. Then, HV measures the volume enclosed by $\mathcal{P}$ and a specified reference point in the objective space [21]. HV has the capability to assess both convergence and diversity of $\mathcal{P}$. Usually, the larger the HV value, the better the performance of a CMOEA. In our experiments, the HV value is calculated by using the reference point which is set to 1.1 times of the upper bounds of the PF.

References

[1] Xiao Long Zheng and Ling Wang. A collaborative multiobjective fruit fly optimization algorithm for the resource constrained unrelated parallel machine green scheduling problem. *IEEE Transactions on Systems Man and Cybernetics: Systems*, 48(5):790–800, 2018.

[2] Chiu Hung Chen and Jyh Horng Chou. Multiobjective optimization of airline crew roster recovery problems under disruption conditions. *IEEE Transactions on Systems Man and Cybernetics: Systems*, 47(1):133–144, 2017.

[3] Boxiong Tan, Hui Ma, Yi Mei, and Mengjie Zhang. Evolutionary multi-objective optimization for web service location allocation problem. *IEEE Transactions on Services Computing*, 2019, in press, DOI: 10.1109/TSC.2018.2793266.

[4] N. G. Paterakis, M. Gibescu, A. G. Bakirtzis, and J. P. S. Catalão. A multi-objective optimization approach to risk-constrained energy and reserve procurement using demand response. *IEEE Transactions on Power Systems*, 33(4):3940–3954, July 2018.

[5] Z. Liu, Y. Wang, and B. Wang. Indicator-based constrained multiobjective evolutionary algorithms. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2019, in press. DOI:10.1109/TSMC.2019.2954491.

[6] Zhun Fan, Yi Fang, Wenji Li, Jiewei Lu, Xinye Cai, and Caimin Wei. A comparative study of constrained multi-objective evolutionary algorithms on constrained multi-objective optimization problems. In *2017 IEEE Congress on Evolutionary Computation (CEC)*, pages 209–216. IEEE, 2017.

[7] Aimin Zhou, Bo-Yang Qu, Hui Li, Shi-Zheng Zhao, Ponnuthurai Nagaratnam Suganthan, and Qingfu Zhang. Multiobjective evolutionary algorithms: A survey of the state of the art. *Swarm and Evolutionary Computation*, 1(1):32–49, 2011.

[8] Z. Fan, H. Li, Caimin Wei, W. Li, Han Huang, X. Cai, and Z. Cai. An improved epsilon constraint handling method embedded in moea/d for constrained multi-objective optimization problems. *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1–8, Dec 2016.

[9] Kalyanmoy Deb, Amrit Pratap, and T Meyarivan. Constrained test problems for multi-objective evolutionary optimization. In *First International Conference on Evolutionary Multi-Criterion Optimization (EMO 2001)*, pages 284–298, 2000.

[10] Himanshu Jain and Kalyanmoy Deb. An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, Part II: Handling constraints and extending to an adaptive approach. *IEEE Transactions on Evolutionary Computation*, 18(4):602–622, 2014.

[11] Zhun Fan, Wenji Li, Xinye Cai, Hui Li, Caimin Wei, Qingfu Zhang, Kalyanmoy Deb, and Erik D Goodman. Difficulty adjustable and scalable constrained multi-objective test problem toolkit. *Evolutionary Computation*, pages 1–40, 2019.

[12] Jia-Peng Li, Yong Wang, Shengxiang Yang, and Zixing Cai. A comparative study of constraint-handling techniques in evolutionary constrained multiobjective optimization. In *2016 IEEE Congress on Evolutionary Computation (CEC)*, pages 4175–4182. IEEE, 2016.

[13] Ryoji Tanabe, Akira Oyama, Ryoji Tanabe, and Akira Oyama. A note on constrained multi-objective optimization benchmark problems. In *IEEE Congress on Evolutionary Computation*, pages 1127–1134, 2017.

[14] Zhun Fan, Wenji Li, Xinye Cai, Hui Li, Caimin Wei, Qingfu Zhang, Kalyanmoy Deb, and Erik D Goodman. Push and pull search for solving constrained multi-objective optimization problems. *arXiv preprint arXiv:1709.05915*, 2017.

[15] K. Li, R. Chen, G. Fu, and X. Yao. Two-archive evolutionary algorithm for constrained multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 23(2):303–315, April 2019.

[16] Jiahai Wang, Weiwei Zhang, and Jun Zhang. Cooperative differential evolution with multiple populations for multiobjective optimization. *IEEE Transactions on Cybernetics*, 46(12):2848–2861, 2016.

[17] Zhizhong Liu and Yong Wang. Handling constrained multiobjective optimization problems with constraints in both the decision and objective spaces. *IEEE Transactions on Evolutionary Computation*, 23(5):870–884, 2019.

[18] Z. Ma and Y. Wang. Evolutionary constrained multiobjective optimization: Test suite construction and performance comparisons. *IEEE Transactions on Evolutionary Computation*, 23(6):972–986, Dec 2019.

[19] Carlos A Coello Coello, Gary B Lamont, David A Van Veldhuizen, et al. *Evolutionary Algorithms for Solving Multi-objective Problems*, volume 5. Springer, 2007.

[20] Eckart Zitzler and Lothar Thiele. Multiobjective optimization using evolutionary algorithms – A comparative case study. In *International Conference on Parallel Problem Solving from Nature*, pages 292–301. Springer, 1998.

[21] Michael Emmerich, Nicola Beume, and Boris Naujoks. An EMO algorithm using the hypervolume measure as selection criterion. In *EMO*, volume 3410, pages 62–76. Springer, 2005.