Advances in Constrained Multi-objective Evolutionary Algorithms (Postprint)

Authors: Zhu Yawen, Zhou Hongbiao, Li Yang, XU Haoyuan

Date: 2022-05-10T00:00:00+00:00

Abstract

Constrained multi-objective evolutionary algorithms (CMOEAs), capable of simultaneously handling multiple conflicting objective functions and constraints while guiding the population toward optimal solutions in the feasible region, have garnered widespread attention from researchers. First, the relevant definitions of constrained multi-objective optimization problems (CMOPs) and three classifications of multi-objective evolutionary algorithms (MOEAs) are introduced. Second, the constraint handling mechanisms in current CMOEAs are systematically analyzed, distilling four main constraint handling approaches. Then, a comprehensive review of the research advances in CMOEAs is presented from three perspectives: dominance-based, indicator-based, and decomposition-based. Finally, the existing challenges and future research directions for CMOEAs are identified.

Full Text

Preamble

Research Progress of Constrained Multi-Objective Evolutionary Algorithms

Zhu Yawen, Zhou Hongbiao†, Li Yang, Xu Haoyuan

(Faculty of Automation, Huaiyin Institute of Technology, Huai'an, Jiangsu 223003, China)

Abstract: Constrained multi-objective evolutionary algorithms (CMOEAs) can simultaneously handle multiple conflicting objective functions and constraint conditions, guiding the population toward optimal solutions in the feasible domain, which has received extensive attention from researchers. This paper first introduces the relevant definitions of constrained multi-objective optimization problems (CMOPs) and three classifications of multi-objective evolutionary algorithms (MOEAs). Second, it systematically analyzes current constraint handling mechanisms in CMOEAs, distilling four main constraint handling methods. Then, it provides a detailed review of research progress on CMOEAs from three perspectives: dominance-based, indicator-based, and decomposition-based. Finally, it identifies existing challenges and future research directions for CMOEAs.

Keywords: constrained multi-objective optimization problems; constrained multi-objective evolutionary algorithms; dominance-based; indicator-based; decomposition-based

0 Introduction

Real-world optimization problems involve not only multiple conflicting objective functions but also various constraint conditions [1]. For example, in multi-objective optimization of wastewater treatment, energy consumption and penalty costs are common objectives, while compliance limits for water quality indicators such as effluent ammonia nitrogen and total phosphorus, along with upper and lower bounds for control variables, serve as inequality constraints [2]. Similar constrained multi-objective optimization problems with equality or inequality constraints exist in fields such as job shop scheduling [3], traffic route optimization [4], and financial investment [5], collectively termed constrained multi-objective optimization problems (CMOPs) [6,7].

Research on CMOPs primarily addresses two issues: first, finding a well-converged, uniformly distributed, and comprehensively covered approximate Pareto solution set; second, handling constraint conditions to facilitate population convergence toward optimal solutions in the feasible domain. However, due to the nonlinearity, high-order characteristics, and uncertainty of practical engineering problems [8], traditional algorithms struggle to approximate the true Pareto front. Moreover, the presence of constraints makes the structure of feasible regions in the search space more complex, hindering traditional optimization algorithms from effectively discovering potential feasible domains [9].

Multi-objective evolutionary algorithms (MOEAs) can obtain a set of Pareto optimal solutions for MOPs in a single run, attracting widespread research attention. Based on different selection strategies, MOEAs can be broadly classified into three categories: 1) **Dominance-based frameworks**, which employ Pareto dominance relationships to drive population evolution toward the true Pareto front, with typical representatives including NSGA-II [10], SPEA2 [11], and MOPSO [12,13]; 2) **Indicator-based frameworks**, which use performance metrics such as hypervolume (HV) to evaluate solution convergence and diversity, with typical representatives including IBEA [14], SIBEA [15], and HypE [16]; 3) **Decomposition-based frameworks**, which collaboratively decompose an MOP into a series of single-objective optimization subproblems

[17], with typical representatives including C-MOGA [18], MOGLS [19], and MOEA/D [20].

The aforementioned MOEAs and their variants were developed for unconstrained optimization problems. Since practical problems generally involve constraints, researchers naturally integrated constraint handling techniques into MOEAs, forming constrained multi-objective evolutionary algorithms (CMOEAs). Currently, commonly used constraint handling techniques include penalty function methods [21], constraint and objective separation methods [22], multi-objective optimization methods [23], and hybrid methods [24]. The organic integration of these techniques with MOEAs not only balances feasibility, convergence, and distribution but also significantly improves algorithm search efficiency. However, these methods have certain drawbacks. For instance, penalty function methods suffer from difficult parameter settings [25], constraint and objective separation methods require manual specification of different values, multi-objective optimization methods exhibit exponential growth in computational complexity with the number of constraints, and hybrid methods lack effective theoretical guidance on integrating different mechanisms.

In recent years, researchers have proposed innovative approaches such as penalty function mechanisms based on fuzzy rules [26], dynamic fusion mechanisms for unconstrained and constrained search [27], and constraint handling mechanisms based on dual collaborative archives [28], greatly enhancing the ability to find optimal solutions in feasible domains. Evidently, constraint handling techniques critically impact the efficient solution of CMOPs. Therefore, organizing and analyzing research achievements in the CMOEAs field, summarizing existing problems, and forecasting future challenges constitute important and necessary research work.

This paper first introduces the definitions of CMOPs and classifications of MOEAs. Second, it analyzes the advantages and disadvantages of four commonly used constraint handling mechanisms. Then, it provides a detailed review of CMOEAs research progress from dominance-based, indicator-based, and decomposition-based perspectives. Finally, it identifies challenges and future research directions for CMOEAs.

---

1 Preliminaries

1.1 Mathematical Formulation of CMOPs

Without loss of generality, considering minimization problems, CMOPs can be defined as follows:

$$
\begin{aligned}
\min \quad & F(x) = (f_1(x), f_2(x), \ldots, f_m(x)) \\
\text{s.t.} \quad & g_i(x) \leq 0, \quad i = 1, 2, \ldots, r \\
& h_i(x) = 0, \quad i = r+1, \ldots, q \\
& x \in \Omega \subset \mathbb{R}^n
\end{aligned}
$$

where $\Omega \subset \mathbb{R}^n$ is the decision space, $f_1(x), f_2(x), \ldots, f_m(x)$ are real-valued functions, m is the number of objectives, $g_i(x)$ represents r inequality constraints, and $h_i(x)$ represents $q - r$ equality constraints.

In practice, equality constraints are typically converted to inequality constraints as shown in equation (2):

$$|h_i(x)| - \delta \geq 0$$

where $\delta > 0$ is called the tolerance parameter for inequality constraints.

The overall constraint violation degree of an individual x can be calculated as:

$$v(x) = \sum_{i=1}^{m} G_i(x)$$

where $G_i(x)$ represents the distance from individual x to the i-th constraint, and $G(x)$ represents the distance from individual x to the feasible domain, reflecting the degree of constraint violation.

1.2 Definitions Related to CMOPs

Definition 1 (Feasible Solution). An individual x is called a feasible solution of CMOPs if and only if $g_i(x) \leq 0$ for $i \in \{1, \ldots, r\}$ and $h_i(x) = 0$ for $i \in \{r+1, \ldots, q\}$.

Definition 2 (Infeasible Solution). A solution x that cannot simultaneously satisfy $g_i(x) \leq 0$ for $i \in \{1, \ldots, r\}$ and $h_i(x) = 0$ for $i \in \{r+1, \ldots, q\}$ is called an infeasible solution.

Definition 3 (Feasible Region). The set composed of all feasible solutions is called the feasible region Ω of CMOPs, as shown in [Figure 1: see original paper].

Definition 4 (Infeasible Region). The set composed of all infeasible solutions is called the infeasible region of CMOPs.

1.3 Classification of MOEAs

1) Dominance-based MOEAs

Dominance-based MOEAs divide the population into multiple layers using Pareto dominance relationships, selecting solutions layer by layer to ensure convergence. Within the same layer, solutions with smaller crowding distance values are preferred for inheritance to the next generation to maintain distribution. [Figure 2: see original paper] illustrates the non-dominated sorting process in NSGA-II. In the figure, all individuals in classification subsets F_1 and F_2 enter the new population P_{t+1}, but only some individuals from subset F_3 can enter P_{t+1}.

2) Indicator-based MOEAs

Indicator-based MOEAs employ performance evaluation metrics to assess solution quality and measure individual convergence and diversity in the population. [Figure 3: see original paper] shows a schematic diagram of the $I_{\varepsilon+}$ indicator. In [Figure 3: see original paper], A and B are two solution sets each containing a single individual. In Figure 3: see original paper, the individuals in sets A and B are non-dominated, with $I_{\varepsilon+}(B, A) > 0$ and $I_{\varepsilon+}(A, B) > 0$. In Figure 3: see original paper, the individual in set B dominates the individual in set A, with $I_{\varepsilon+}(B, A) < 0$ and $I_{\varepsilon+}(A, B) > 0$.

3) Decomposition-based MOEAs

In 2002, Murata et al. first proposed a decomposition-based MOEA called the cellular multi-objective genetic algorithm (C-MOGA) [29]. C-MOGA uses the weighted sum (WS) approach as an aggregation function for problem decomposition, where generated offspring are only compared with the current individual. In 2007, Zhang and Li proposed the MOEA/D algorithm, which decomposes a multi-objective optimization problem into multiple single-objective subproblems through aggregation functions and optimizes these subproblems collaboratively. Compared with C-MOGA, MOEA/D employs not only the WS aggregation function but also investigates the Tchebycheff (TCH) and penalty-based boundary intersection (PBI) aggregation functions. Additionally, it introduces the concept of neighborhoods to improve population evolution efficiency. [Figure 4: see original paper] illustrates the TCH decomposition method.

---

2 Constraint Handling Mechanisms

After nearly two decades of development, CMOEAs have evolved various mature constraint handling mechanisms. This section introduces four commonly used constraint handling mechanisms.

1) Penalty Function Methods

Penalty function methods penalize individuals violating constraints according to their violation degree, reducing their selection probability, as shown in equations (5) and (6):

$$G(x) = \sum_{i=1}^{m} G_i(x)$$

Due to their simplicity and ease of application, penalty function methods are the most popular constraint handling approach in the CMOEAs domain. Several variants have emerged, including death penalty [30], static penalty, dynamic penalty [31], and adaptive penalty [32]. In death penalty methods, any solution violating constraints is considered infeasible and rejected from algorithm evolution. In static penalty methods, penalty factors remain constant throughout evolution. In dynamic penalty methods, penalty factors typically vary with generation number, using smaller values in early stages to expand search regions and improve diversity, and larger values in later stages to focus search within feasible regions and improve convergence. Adaptive penalty methods generally extract constraint violation information from the evolutionary process and use this feedback to adaptively adjust penalty factors.

However, penalty function methods are parameter-dependent; excessively large or small parameters affect algorithm performance, and parameter tuning is extremely time-consuming [33]. Therefore, the primary challenge lies in designing adaptive strategies to efficiently determine appropriate penalty factors.

2) Constraint and Objective Separation Methods

These methods treat objective values and constraint violation degrees separately. Deb et al. [34] proposed three feasibility criteria for individual comparison during replacement to maintain balance between feasible and infeasible solutions: 1) If both individuals are feasible, evaluate them based on objective function values; 2) If one is feasible and the other infeasible, select the feasible individual; 3) If both are infeasible, select the individual with lower constraint violation.

Feasibility criteria are simple to implement and parameter-free, making it easier to find feasible solutions. However, they overemphasize feasible solutions while ignoring potentially useful information carried by infeasible solutions, risking premature convergence to local optima. Other representative methods include the constraint dominance principle (CDP) [34], stochastic ranking (SR) [35], and -constraint handling [36].

The CDP method establishes the following criteria to evaluate solutions x and y for replacement operations: 1) If x is feasible and y is infeasible, do not replace x with y; 2) If x is infeasible and y is feasible, replace x with y; 3) If both x and y are infeasible, replace the solution with higher constraint violation with the one having lower violation; 4) If both are feasible, replace the solution with poorer performance metric with the better one.

The SR method uses a probability parameter p_f to determine whether the algorithm evaluates solution quality based on objective function values or constraint violation degrees. Algorithm 1 presents the SR pseudocode. The SR method effectively balances constraint violation and fitness values during search, maximizing useful information from feasible solutions while 挖掘 ing potential information from infeasible solutions, strongly driving the algorithm toward optimal values in feasible domains.

In the -constraint handling method, a piecewise function controls the value. Individuals are divided into different regions based on total constraint violation, with different evaluation criteria applied:

$$\varepsilon(t) = \begin{cases} \varepsilon(0)\left(1 - \frac{t}{T_c}\right)^{cp}, & 0 < t < T_c \\ 0, & t \geq T_c \end{cases}$$

where t is the current generation, T_c is the control generation, cp controls the reduction rate, and $v(x_\theta)$ represents the constraint violation degree of the θ-th individual in the population.

The -level comparison $\prec_\varepsilon$ is defined based on the ordering relationship on the $(f(x), v(x))$ set. Let $f(x_1), f(x_2)$ and $v(x_1), v(x_2)$ be the objective values and constraint violation degrees of solutions x_1 and x_2, respectively. For any $\varepsilon \geq 0$, the -level comparison is defined as:

$$x_1 \prec_\varepsilon x_2 \Leftrightarrow \begin{cases} f(x_1) \leq f(x_2), & \text{if } v(x_1), v(x_2) \leq \varepsilon \\ v(x_1) < v(x_2), & \text{if } v(x_1) \leq \varepsilon < v(x_2) \\ v(x_1) < v(x_2), & \text{otherwise} \end{cases}$$

3) Multi-Objective Optimization Methods

The core idea is to transform constraints into objective functions, converting CMOPs into MOPs. Two approaches exist: first, transforming constrained optimization problems into bi-objective problems of the form $\min F(x) = (f(x), G(x))$, where $f(x)$ is the original objective function and $G(x)$ is the total violation degree of $(r + q)$ constraints. Second, transforming constrained problems into MOPs with $(r + q + 1)$ objectives. Although multi-objective optimization methods demonstrate good convergence when handling CMOPs, computational complexity increases exponentially.

4) Hybrid Methods

Hybrid methods typically employ three operational paradigms: 1) Multi-mechanism strategies using two or more constraint handling mechanisms; 2) Multi-stage strategies dividing the search process into phases with different mechanisms in each phase; 3) Cooperative coevolution strategies where multiple mechanisms collaborate to effectively solve complex constrained optimization problems.

3 Research on CMOEAs Classification

Based on different selection mechanisms, this section elaborates on CMOEAs from three major categories: dominance-based, indicator-based, and decomposition-based.

3.1 Dominance-Based CMOEAs

This subsection first introduces the application of penalty function methods in dominance-based CMOEAs. Zhou et al. [37] proposed a hybrid multi-objective bare-bones particle swarm optimization algorithm (HBBMOPSO) based on Pareto dominance and decomposition. This method constructed constrained multi-objective functions including energy consumption and penalties using an adaptive penalty function approach. To improve the convergence and diversity of the Pareto optimal solution set, the method utilized decomposition strategies to select individual leaders and employed Pareto dominance and crowding distance methods to maintain external archives. Azzouz et al. [38] proposed the dynamic nondominated sorting genetic algorithm II (DNSGA-II) to handle dynamic constraint problems where objective functions, constraints, or problem parameters change over time. This method adaptively adjusted penalty terms based on the proportion of feasible solutions in the current population to handle dynamic constraints. Experimental results showed that DNSGA-II outperformed the original algorithm in convergence and diversity on most test problems.

Beyond parameterized penalty function methods, some parameter-free approaches exist. Jadaan et al. [39] proposed a constrained multi-objective evolutionary algorithm combining nondominated ranked genetic algorithm (NRGA) with a parameter-free penalty function method (PP-NRGA). This method used a parameter-free penalty coefficient r_g requiring no prior problem knowledge and introduced ranks in the fitness function to facilitate convergence to the global Pareto front. Experimental results demonstrated that PP-NRGA improved convergence speed while maintaining diversity. Ning et al. [40] proposed a novel constrained nondominated sorting (CNS) constraint handling method. This method assigned a constrained nondomination rank to each individual in the population using both constraint violation degree and Pareto rank, achieving balance between feasibility and optimality. To effectively solve CMOPs, CNS was embedded into MOEA/D-M2M to form the competitive cMOEA/H algorithm.

Due to difficulties in setting penalty parameters, the performance of CMOEAs with penalty function-based constraint handling mechanisms is hard to guarantee. Consequently, researchers have focused more on feasibility criterion methods within constraint and objective separation approaches.

Zhang et al. [41] proposed a constraint handling method based on NSGA-II

(NSGA-II-TRA) to address constraints in optimal testing resource allocation problems (OTRAPs). This method used heuristic algorithms to repair element values causing constraint violations and employed z-score-based Euclidean distance to evaluate differences between individuals. Sensitivity analysis results showed that NSGA-II-TRA exhibited good robustness. Jiao et al. [42] proposed an improved NSGA-II algorithm with a feasible solution guiding strategy. This method modified individual objective function values using constraint violation values and true objective function values, maintained algorithm balance using the feasible solution ratio from the current population, and determined feasible directions for infeasible solutions under the guidance of feasible individuals, enabling search for Pareto optimal solutions from both feasible and infeasible spaces. Experimental results demonstrated superiority in convergence and diversity. Fan et al. [43] proposed a novel dominance-based clustering method using infeasible solutions to handle CMOPs and utilized differential evolution crossover operators for rapid convergence to the Pareto front.

In constraint and objective separation methods, Ray et al. [44] proposed a nondomination-based CMOEA that eliminated scaling and aggregation issues in penalty function methods by separately processing constraints and objectives using Pareto sorting concepts in both objective and constraint domains. Experimental results showed the algorithm could obtain a diverse set of Pareto optimal solutions. Elarbi et al. [45] proposed the isolated solution-based constrained NSGA-III algorithm (ISC-NSGA-III), which improved diversity and convergence by selecting infeasible solutions related to isolated subregions (i.e., weight vectors or reference points). Yang et al. [46] proposed a multi-objective differential evolution algorithm based on dominance and -constraint handling switching mechanisms (MODE-CHS) to balance feasibility, convergence, and distribution in CMOPs. This algorithm accelerated population convergence while obtaining feasible solutions through -constraint techniques. Experimental results showed MODE-CHS was highly competitive in solving CMOPs. Cuate et al. [47] proposed a two-stage hybrid evolutionary algorithm (-NSGA-II/PT) combining NSGA-II with a Pareto tracer (PT) to solve equality-constrained CMOPs. The PT strategy could track optimal solutions along the Pareto front and handle constraints. Experimental results demonstrated -NSGA-II/PT's superiority over existing methods.

In multi-objective optimization methods, Isaacs et al. [48] proposed a constraint handling evolutionary algorithm (CHEA) based on improved NSGA-II. This method transformed the original constrained optimization problem with k objectives into an unconstrained problem with $k + 1$ objectives, using a custom parameter α to control the number of infeasible solutions retained in the population and leveraging infeasible space search to increase solution diversity. Experimental results showed CHEA could rapidly search for feasible domain optimal solution sets.

In hybrid method applications, Deb et al. [49] proposed a CMOPs handling method combining bi-objective evolutionary algorithms with penalty function methods. This approach treated constraint violation minimization as an additional objective combined with bi-objective evolutionary optimization strategies, used the obtained Pareto front to estimate penalty parameter R, and employed classical local search methods to solve penalty functions, promoting algorithm convergence. Experimental results showed the evolutionary algorithm with penalty function-based constraint handling could quickly and accurately find optimal solutions for constrained problems. Venkatraman et al. [50] proposed a two-stage framework based on genetic algorithms for CMOPs. In the first stage, constrained optimization problems were treated as unconstrained problems to find feasible solutions; in the second stage, objective functions and constraints were considered as a bi-objective optimization problem. Experimental results proved the algorithm was problem-independent and highly robust. Woldesenbet et al. [51] proposed a hybrid constraint handling technique based on adaptive penalty functions and distance metrics. This simple, parameter-free method demonstrated better handling results.

Most practical engineering problems exhibit dynamic characteristics where objective functions, decision parameters, and constraints change over time. For such dynamic CMOPs, Azzouz et al. [52] combined adaptive penalty functions with feasibility-driven strategies to propose a dynamic constrained multi-objective evolutionary algorithm (DC-MOEA) based on NSGA-II. In DC-MOEA, the feasibility-driven strategy guides search convergence toward new feasible solutions according to environmental changes. summarizes research on dominance-based CMOEAs.

Additional studies apply CMOEAs to practical engineering problems. Ji et al. [53] developed an improved NSGA-II algorithm by converting constraint violations into objective functions and applied it to the continuous berth allocation problem (BAPC). Gadhvi et al. [54] used NSGA-II, SPEA2, and PESA-II with constraint handling mechanisms for constrained multi-objective optimization of vehicle passive suspension systems. Sorkhabi [55] proposed a novel constraint handling method based on NSGA-II using penalty functions and constraint programming (CHCP) to balance local and global exploration, improving optimization efficiency for multi-objective wind farm layout optimization with constraints. Abul'Wafa et al. [56] proposed a controlled elitist NSGA-II algorithm for multi-objective economic/environmental dispatch problems. Bora et al. [57] used constrained NSGA-II and reinforcement learning for multi-objective wind power dispatch. Song et al. [58] designed an improved NSGA-II algorithm using constraint steering rules for multi-objective land allocation spatial optimization. Rathnayake [59,60] improved NSGA-II's constraint handling capability using binary tournament constraint handling techniques for multi-objective control of sewer systems, with experimental results showing better performance than other techniques. Zhou et al. [61] proposed an intelligent optimal control method for wastewater treatment processes based on constrained multi-objective bare-bones particle swarm optimization. summarizes engineering applications of dominance-based CMOEAs.

3.2 Indicator-Based CMOEAs

Compared with dominance-based CMOEAs, research on indicator-based CMOEAs is relatively limited. Key contributions include:

García et al. [62] proposed an indicator-based MOEA that uses reference sets to guide population search and IGD values to drive population convergence, ultimately achieving the goal of obtaining constraint boundary feasible solutions through performance indicators.

In constraint and objective separation methods, Scimemi et al. [63] proposed a hypervolume indicator-based constraint handling technique that extends the SR method from single-objective to multi-objective optimization, with experimental results demonstrating applicability.

In hybrid methods, Said et al. [64] proposed an indicator-based co-evolutionary migration based algorithm (IB-CEMBA) for solving combinatorial bi-level multi-objective optimization problems. Bi-level problems consist of upper-level (leader) and lower-level (follower) components, where the lower-level problem appears as a constraint for the upper-level. IB-CEMBA uses indicator-based methods to measure whether solutions sent from the lower to upper level have optimal indicator contributions. Guo et al. [65] proposed a hybrid MOEA combining indicator-based IBEA with satisfiability modulo theories (SMT) called SMTIBEA for solving real-world software product line configuration optimization problems. Antonio et al. [66] proposed a new cooperative coevolution MOEA based on hypervolume indicators (IBMOEA) for CMOPs.

Some achievements combine multiple optimization algorithms for constrained problems. BenMansour et al. [67] proposed a combined algorithm (IW-BLS/ACO) based on ant colony optimization and neighborhood methods (local search) for multi-objective multi-dimensional knapsack problems. This method uses ant colony optimization to create initial solutions and local search with -indicator for spatial search, achieving better trade-offs between exploration and exploitation. Mansour et al. [68] proposed an indicator-based ant colony optimization algorithm for multi-objective knapsack constrained problems.

To prevent populations from falling into local optima when handling CMOPs, Yuan et al. [69] proposed an indicator-based constrained multi-objective algorithm (ICMA). This method first introduces distance metrics to measure crowding between solutions in objective space, guiding populations to explore different regions thoroughly, then uses this metric and overall constraint violation to design evaluation indicators for feasible and infeasible solutions, effectively guiding populations toward promising regions and preventing local optima. Experimental results showed ICMA outperformed other CMOEAs. Liu et al. [70] developed indicator-based CMOEAs (HypE, IBEA, and ISDE) combined with three constraint handling techniques (feasibility criterion, SR, -constraint method). Experimental results demonstrated that indicator-based CMOEAs were highly competitive for CMOPs. Wang et al. [71] proposed an improved SATIBEA+

method based on SATIBEA, which maps feature selection problems to SAT problems by introducing binary variables for each feature and mapping constraints to formulas. The method defines minimizing the number of invalid constraints as a search objective, replacing invalid configurations with valid or "better" configurations (with fewer constraint violations) during search to reduce feature violations and increase the percentage of valid configurations. Experimental results showed SATIBEA+ outperformed SATIBEA.

Shi et al. [72] proposed an indicator-based evolutionary algorithm parallel portfolio approach (IBEAPORT) that composes three algorithm variants (SATIBEAv1, SATIBEAv2, and SATIBEAv3) by merging SAT solving into IBEA in different ways and executes them using parallelization techniques. Experimental results showed IBEAPORT leveraged the exploration capabilities of different algorithms and improved optimality within limited time budgets. summarizes research on indicator-based CMOEAs.

In engineering applications, Li et al. [73] proposed a parallel bi-criterion evolution indicator-based evolutionary algorithm (PBCE-IBEA) to balance economy and security in hybrid AC/VSC-MTDC systems. Experimental results demonstrated PBCE-IBEA could effectively trade off economy and security. Tangpattanakul et al. [74] proposed an indicator-based multi-objective local search (IBMOLS) for CMOPs related to Earth observation satellite selection and scheduling. lists engineering applications of indicator-based CMOEAs.

3.3 Decomposition-Based CMOEAs

This subsection primarily uses MOEA/D as an example to introduce research status on decomposition-based CMOEAs.

Some methods utilize feasible or infeasible solutions when handling CMOPs. Miyakawa et al. [75] proposed a constrained MOEA/D with directed mating and archive (CMOEA/D-DMA). This method employs directed mating to select useful infeasible solutions with better scalar function values than feasible solutions as parents and stores them in an archive. Tests on continuous mCDTLZ and multi-objective discrete knapsack problems showed improved search performance. Elarbi et al. [76] proposed an isolated solution-based constrained MOEA/D (ISC-MOEA/D) for constrained many-objective optimization problems (CMaOPs) with various complex features. The method first handles CMaOPs with complex features through an isolated solution-based update procedure, then selects infeasible solutions related to isolated subregions with smaller constraint violations. Experimental results showed ISC-MOEA/D was more competitive.

In constraint and objective separation methods, Fan et al. [77] proposed MOEA/D with angle-based constrained dominance principle (MOEA/D-ACDP). ACDP uses angle information between two solutions and the proportion of feasible solutions to adjust dominance relationships, simultaneously maintaining good convergence, diversity, and feasibility. Asafuddoula et al. [78] proposed an adaptive constraint handling approach for MOEA/D. This method includes an adaptive constraint violation threshold setting mechanism where infeasible solutions with violation values below the threshold are treated equivalently to feasible solutions. Fan et al. [79] applied an improved -constraint handling mechanism in the MOEA/D framework, with experimental results showing effectiveness in both convergence and diversity. Martinez et al. [80] proposed a novel selection mechanism based on -constraint handling that uses MOEA/D neighborhood information to obtain solutions minimizing objective functions within feasible regions. Results showed the new -constraint method achieved optimal balance between feasible solution generation and accelerated convergence toward the Pareto optimal front.

Moreover, constraints may cause algorithms to fall into two stagnation states: 1) Since feasible regions of CMOPs generally consist of disconnected feasible subregions, algorithms may become trapped in subregions without global Pareto optimal solutions; 2) A constraint violation function may contain many non-zero minima, trapping search in infeasible regions. To address these issues, Zhu et al. [81] designed a detect-and-escape (DAE) strategy based on MOEA/D to help constraint handling methods escape infeasible regions and local optimal Pareto fronts, with the constraint handling mechanism flow shown in [Figure 5: see original paper].

In multi-objective optimization-based constraint handling methods, Zapotecas-Martínez et al. [82] proposed a multi-objective constraint handling method based on MOEA/D. This method constructs a bi-objective problem with scalar functions and constraint violation degrees as objectives, selecting optimal solutions through scalar functions to enhance the possibility of using infeasible solutions to approximate CMOPs' Pareto fronts, maintaining balance among convergence, diversity, and feasibility. Peng et al. [83] also proposed a multi-objective constraint handling method based on MOEA/D. This method transforms constraint violation degree into an objective function and uses a portion of weights distributed in infeasible regions to select infeasible nondominated individuals. These weights dynamically change during evolution to find infeasible nondominated individuals with better objective values and smaller constraints, improving population convergence and diversity. Wang et al. [84] proposed a weight vector adjustment strategy based on MOEA/D for multi-objective handling of constraints.

Regarding the issue that common decomposition methods in MOEA/D may cause diversity loss or sensitivity to PF shape, Cai et al. [85] proposed a constrained decomposition with grids (CDG) method called CDG-MOEA. CDG can be viewed as an extension of -constraint methods with inherent characteristics reflecting neighborhood structure information among solutions. In CDG, one objective function is selected for optimization while all other objectives are converted to constraints by setting upper and lower bounds. This method outperformed comparison algorithms in both convergence and diversity and showed robustness to Pareto front shapes. Building on CDG, Cai et al. [86] proposed a dynamic constrained decomposition with grids (DCDG) framework for multi-objective memetic algorithms (DCDG-MOMA) for combinatorial CMOPs with dynamic population sizes. Additionally, DCDG can dynamically increase grid numbers to obtain more nondominated solutions and improve diversity. Experimental comparisons showed clear superiority over other algorithms. compares these two methods.

In hybrid methods for solving CMOPs, Chen et al. [87] proposed a decomposition-based MOEA with the -constraint framework (DMOEA- C). This method incorporates -constraint into decomposition strategies, selecting one objective as primary while converting others to constraints, and uses feasibility criteria to optimize subproblems. Experimental comparisons showed clear superiority and significant advantages in solving 0-1 knapsack problems. Liu et al. [88] proposed a MOEA/D evolutionary algorithm with boundary search and archiving for CMOPs. summarizes decomposition-based CMOEAs research.

In engineering applications, Zhu et al. [89] used MOEA/D with penalty function constraint handling for economic emission dispatch in wind-thermal power systems. Biswas et al. [90] used MOEA/D with superiority of feasible solutions (SF) constraint handling for constrained multi-objective optimal power flow. Zhou et al. [91] used MOEA/D with feasibility criterion constraint handling for constrained multi-objective optimal control of wastewater treatment processes. Li et al. [92] used improved MOEA/D for MEMS layout optimization. Hu et al. [93] proposed a MOEA based on MOEA/D and constraint programming for multi-objective team orienteering problems with time windows (MOTOPTW). Li et al. [94] proposed a problem-specific heuristics MOEA/D (PH-MOEAD) for hybrid flowshop lot-streaming problems. lists engineering applications of decomposition-based CMOEAs.

---

4 Conclusion

This paper summarized CMOEAs research achievements from dominance-based, indicator-based, and decomposition-based perspectives, and analyzed the specific applications of four typical constraint handling mechanisms: penalty function methods, constraint and objective separation methods, multi-objective optimization methods, and hybrid methods. Reviewing research trends over the past decade, CMOEAs theory has become richer, constraint handling effects more prominent, and engineering applications continuously expanding. However, several challenges remain:

1) Handling Complex Constraints

To find feasible solutions, equality constraints are often converted to inequality constraints. However, for constrained optimization problems with complex equality constraints that are nonlinear or discrete, evolutionary algorithms struggle to find feasible solutions. In discrete constrained optimization, optimal solutions easily fall into local optima in feasible regions. Therefore, how to reasonably utilize complex constraints to find feasible solutions while preventing algorithms from falling into local optima remains an open problem. Future research could consider comprehensively applying existing constraint handling techniques to design hybrid mechanisms for complex search environments, or dividing populations into multiple subpopulations for collaborative evolution to achieve constraint handling at different evolutionary stages.

2) Imbalance Between Constraints and Objectives

When solving CMOPs, overemphasizing constraints may miss Pareto optimal solutions, while overemphasizing objective functions may deviate populations from feasible regions. For example, in wastewater treatment multi-objective optimization, strictly requiring all water quality parameters to meet standards would result in extremely high electricity costs; conversely, pursuing minimum energy consumption alone would fail to guarantee water quality constraints, causing significant environmental harm. Therefore, balancing constraints and objective functions according to engineering problem characteristics is crucial. Future research could consider obtaining Pareto fronts of interest based on decision-maker preferences or identifying urgently needed solutions that violate constraints.

3) Many-Objective Problems

Real-life problems often require considering numerous factors. When the number of objectives exceeds three, traditional Pareto dominance relationships lose discriminative power, causing convergence speed degradation and inability to converge well to the Pareto front. Therefore, designing evolutionary mechanisms with synchronized selection pressure and constraint handling for constrained many-objective optimization problems is extremely challenging. Future research could consider phased processing of convergence and diversity, such as using weak dominance relations in the first stage to promote convergence, strong dominance relations in the second stage to enhance diversity, and adopting dual-archive strategies for constraint handling while providing strong selection pressure.

4) Test Functions

Real-world problems are complex and variable. Existing constrained test function sets for CMOEAs performance verification can no longer meet practical engineering needs. Therefore, designing benchmark test functions that reflect real engineering problem characteristics is necessary. Future research could consider designing test function generators with controllable parameters.

In summary, CMOEAs still face numerous challenges when handling CMOPs. With the development of science, technology, and intelligent evolutionary algorithms, there remains room for improvement in CMOEAs' constraint handling effectiveness and optimization performance. Most current achievements focus on performance improvements to original algorithms, while new algorithm theories and constraint handling techniques require further research. Real-world problems are typically dynamic constrained optimization problems interacting with time or environment, where optimal solutions change dynamically. Therefore, research on dynamic constrained multi-objective optimization problems 仍需从算法框架、参数控制、处理机制等方面进行研究. Designing adaptive learning mechanisms meeting exploration and exploitation needs, or combining with advanced machine learning theories (such as deep learning, transfer learning, reinforcement learning) to design knowledge-driven constrained multi-objective evolutionary algorithms for improving population self-organizing learning capabilities, are also promising research directions.

References

[1] Qu B Y, Suganthan P N. Constrained multi-objective optimization algorithm with an ensemble of constraint handling methods [J]. Engineering Optimization, 2011, 43 (4): 403-416.

[2] Zhou Hongbiao, Qiao Junfei. Multi-objective optimal control for wastewater treatment process using adaptive MOEA/D [J], Applied Intelligence, 2019, 49 (3): 1098-1126.

[3] Khalilpourazari S, Pasandideh S H R. Multi-objective optimization of multi-item EOQ model with partial backordering and defective batches and stochastic constraints using MOWCA and MOGWO [J]. Operational Research, 2020, 20 (3): 1729-1761.

[4] Zhu Siying, Zhu Feng. Multi-objective bike-way network design problem with space-time accessibility constraint [J]. Transportation, 2020, 47 (5): 2479-2503.

[5] El-Abbasy M S, Elazouni A, Zayed T. Finance-based scheduling multi-objective optimization: benchmarking of evolutionary algorithms [J]. Automation in Construction, 2020, 120: Article ID 103392.

[6] Michalewicz Z, Schoenauer M. Evolutionary algorithms for constrained parameter optimization problems [J]. Evolutionary Computation, 1996, 4 (1): 1-32.

[7] Coello C A C. Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art [J]. Computer Methods in Applied Mechanics and Engineering, 2002, 191 (11-12): 1245-1287.

[8] Chen Zhiwang, Bai Xin, Yang Qi, et al. Decision space constraint, domination and same order solution selection strategy in interval multi-objective optimization [J]. Acta Automatica Sinica, 2015, 41 (12): 2115-2124.

[9] Wang Luping, Zhang Qingfu, Zhou Aimin, et al. Constrained subproblems in a decomposition-based multi-objective evolutionary algorithm [J]. IEEE Trans on Evolutionary Computation, 2015, 20 (3): 2953−2965.

[10] Deb K, Pratap A, Agarwal S, et al. A fast and elitist multi-objective genetic algorithm: NSGA-II [J]. IEEE Trans on Evolutionary Computation, 2002, 6 (2): 182-197.

[11] Zitzler E, Laumanns M, Thiele L. SPEA2: improving the strength pareto evolutionary algorithm [J/OL]. Technical Report Gloriastrasse, 2001, 103. (2001-07-13) [2022-03-31]. doi: 10.3929/ETHZ-A-004284029.

[12] Coello C A C, Lechuga M S. MOPSO: a proposal for multiple objective particle swarm optimization [J]. IEEE Service Center, 2002, 2: 1051-1056.

[13] Qiao Junfei, Zhou Hongbiao, Yang Cuili. Bare-bones multi-objective particle swarm optimization based on parallel cell balanceable fitness estimation [J]. IEEE Access, 2018, 6 (6): 32493-32506.

[14] Zitzler E, Künzli S. Indicator-based selection in multi-objective search [J]. Lecture Notes in Computer Science, 2004, 3242: 832-842.

[15] Brockhoff D, Zitzler E. Improving hypervolume-based multi-objective evolutionary algorithms by using objective reduction methods [C]// Proc of IEEE Congress on Evolutionary Computation. IEEE, 2007: 2086-2093.

[16] Bader J, Zitzler E. HypE: an algorithm for fast hypervolume-based many-objective optimization [J]. Evolutionary Computation, 2011, 19 (1): 45-76.

[17] Qiao Junfei, Zhou Hongbiao, Yang Cuili, Yang Shengxiang. A decomposition-based multi-objective evolutionary algorithm with angle-based adaptive penalty [J]. Applied Soft Computing, 2019, 74 (1): 190-202.

[18] Murata T, Gen M. Cellular genetic algorithm for multi-objective optimization [C]// Proc of the 4th Asian Fuzzy System Symposium. 2002: 103.

[19] Ishibuchi H, Murata T. A multi-objective genetic local search algorithm and its application to flowshop scheduling [J]. IEEE Trans on Systems, Man, and Cybernetics, Part C: Applications and Reviews, 1998, 28 (3): 392-403.

[20] Li Hui, Zhang Qingfu. Multi-objective optimization problems with complicated pareto sets, MOEA/D and NSGA-II [J]. IEEE Trans on Evolutionary Computation, 2008, 13 (2): 284-302.

[21] Homaifar A, Qi C X, Lai S H. Constrained optimization via genetic algorithms [J]. Simulation, 1994, 62 (4): 242-253.

[22] Jain H, Deb K. An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part II: handling constraints and extending to an adaptive approach [J]. IEEE Trans on Evolutionary Computation, 2013, 18 (4): 602-622.

[23] Cai Zixing, Wang Yong. A multi-objective optimization-based evolutionary algorithm for constrained optimization [J]. IEEE Trans on Evolutionary Computation, 2006, 10 (6): 658-675.

[24] Wang Yong, Cai Zixing, Zhou Yuren, et al. An adaptive tradeoff model for constrained evolutionary optimization [J]. IEEE Trans on Evolutionary Computation, 2008, 12 (1): 80-92.

[25] Shang Ronghua, Jiao Licheng, Ma Wenping. Immune clonal multi-objective optimization algorithm for solving constrained optimization problems [J]. Journal of Software, 2008 (11): 2943-2956.

[26] Saha C, Das S, Pal K, et al. A fuzzy rule-based penalty function approach for constrained evolutionary optimization [J]. IEEE Trans on Cybernetics, 2014, 46 (12): 2953−2965.

[27] Yang Yongkuan, Liu Jianchang, Tan Shubin. A constrained multi-objective evolutionary algorithm based on decomposition and dynamic constraint-handling mechanism [J]. Applied Soft Computing, 2020, 89: Article ID 106104.

[28] Li Ke, Chen Renzhi, Fu Guangtao, et al. Two-archive evolutionary algorithm for constrained multi-objective optimization [J]. IEEE Trans on Evolutionary Computation, 2018, 23 (2): 303-315.

[29] Murata T, Gen M. Cellular genetic algorithm for multi-objective optimization [C]// Proc of the 4th Asian Fuzzy System Symposium. 2002: 103.

[30] Bäck T, Hoffmeister F, Schwefel H P. A survey of evolution strategies [C/OL]/ Proc of the 4th International Conference on Genetic Algorithms. (1994) [2022-03-31]. https://www.researchgate.net/publication/2611416.

[31] Joines J A, Houck C R. On the use of non-stationary penalty functions to solve nonlinear constrained optimization problems with GA's [C]// Proc of the 1st IEEE Conference on Evolutionary Computation. IEEE World Congress on Computational Intelligence. 1994: 579-584.

[32] Coit D W, Smith A E, Tate D M. Adaptive penalty methods for genetic optimization of constrained combinatorial problems [J]. INFORMS Journal on Computing, 1996, 8 (2): 173-182.

[33] Liu Sanyang, Jin Anzhao. Co-evolutionary teaching and learning optimization algorithms for solving constrained optimization problems [J]. Acta Automatica Sinica, 2018, 44 (09): 1690-1697.

[34] Deb K. An efficient constraint handling method for genetic algorithms [J]. Computer Methods in Applied Mechanics and Engineering, 2000, 186 (2-4): 311-338.

[35] Runarsson T P, Yao X. Stochastic ranking for constrained evolutionary optimization [J]. IEEE Trans on Evolutionary Computation, 2000, 4 (3): 284-294.

[36] Takahama T, Sakai S. Constrained optimization by constrained particle swarm optimizer with -level control [J]. Springer Berlin Heidelberg, 2005, 29: 1019-1029.

[37] Zhou Hongbiao, Qiao Junfei. Application of hybrid multi-objective backbone particle swarm optimization algorithm in optimal control of wastewater treatment process [J]. Journal of Chemical Industry and Engineering, 2017, 68 (9): 3511-3521.

[38] Azzouz R, Bechikh S, Ben Said L. Multi-objective optimization with dynamic constraints and objectives: new challenges for evolutionary algorithms [C]// Proc of Annual Conference on Genetic and Evolutionary Computation. 2015: 615-622.

[39] Jadaan O A L, Rajamani L, Rao C R. Non-dominated ranked genetic algorithm for solving constrained multi-objective optimization problems [J]. Journal of Theoretical & Applied Information Technology, 2009, 5 (5): 60-67.

[40] Ning Weikang, Guo Baolong, Yan Yunyi, et al. Constrained multi-objective optimization using constrained non-dominated sorting combined with an improved hybrid multi-objective evolutionary algorithm [J]. Engineering Optimization, 2017, 49 (10): 1645-1664.

[41] Zhang Guofu, Su Zhaopin, Li Miqing, et al. Constraint handling in NSGA-II for solving optimal testing resource allocation problems [J]. IEEE Trans on Reliability, 2017, 66 (4): 1193-1212.

[42] Jiao Licheng, Luo Juanjuan, Shang Ronghua, et al. A modified objective function method with feasible-guiding strategy to solve constrained multi-objective optimization problems [J]. Applied Soft Computing, 2014, 14: 363-380.

[43] Fan Lei, Liu Xiyang. An enhanced domination based evolutionary algorithm for multi-objective problems [C]// Proc of the 9th International Conference on Computational Intelligence and Security. 2013: 95-99.

[44] Ray T, Tai K, Seow K C. Multi-objective design optimization by an evolutionary algorithm [J]. Engineering Optimization, 2001, 33 (4): 399-424.

[45] Elarbi M, Bechikh S, Said L B. On the importance of isolated infeasible solutions in the many-objective constrained NSGA-III [J]. Knowledge-Based Systems, 2021, 227: Article ID 104335.

[46] Yang Yongkuan, Liu Jianchang, Tan Shubin, et al. A multi-objective differential evolution algorithm based on domination and constraint-handling switching [J]. Information Sciences, 2021, 579: 796-813.

[47] Cuate O, Ponsich A, Uribe L, et al. A new hybrid evolutionary algorithm for the treatment of equality constrained MOPs [J]. Mathematics, 2020, 8 (1): 7.

[48] Isaacs A, Ray T, Smith W. Blessings of maintaining infeasible solutions for constrained multi-objective optimization problems [C]// Proc of IEEE Congress on Evolutionary Computation. IEEE, 2008: 2780-2787.

[49] Deb K, Datta R. A fast and accurate solution of constrained optimization problems using a hybrid bi-objective and penalty function approach [C]// Proc

of IEEE Congress on Evolutionary Computation. IEEE, 2010: 1-8.

[50] Venkatraman S, Yen G G. A generic framework for constrained optimization using genetic algorithms [J]. IEEE Trans on Evolutionary Computation, 2005, 9 (4): 424-435.

[51] Woldesenbet Y G, Yen G G, Tessema B G. Constraint handling in multi-objective evolutionary optimization [J]. IEEE Trans on Evolutionary Computation, 2009, 13 (3): 514-525.

[52] Azzouz R, Bechikh S, Said L B, et al. Handling time-varying constraints and objectives in dynamic evolutionary multi-objective optimization [J]. Swarm and Evolutionary Computation, 2018, 39: 222-248.

[53] Ji B, Yuan X, Yuan Y. Modified NSGA-II for solving continuous berth allocation problem: using multi-objective constraint-handling strategy [J]. IEEE Trans on Cybernetics, 2017, 47 (9): 2885-2895.

[54] Gadhvi B, Savsani V, Patel V. Multi-objective optimization of vehicle passive suspension system using NSGA-II, SPEA2 and PESA-II [J]. Procedia Technology, 2016, 23: 361-368.

[55] Sorkhabi S Y D, Romero D A, Beck J C, et al. Constrained multi-objective wind farm layout optimization: novel constraint handling approach based on constraint programming [J]. Renewable Energy, 2018, 126: 341-353.

[56] Abul'Wafa A R. Optimization of economic/emission load dispatch for hybrid generating systems using controlled elitist NSGA-II [J]. Electric Power Systems Research, 2013, 105: 142-151.

[57] Bora T C, Mariani V C, dos Santos Coelho L. Multi-objective optimization environmental-economic dispatch with reinforcement learning based on non-dominated sorting genetic algorithm [J]. Applied Thermal Engineering, 2019, 146: 688-700.

[58] Song M, Chen D. An improved knowledge-informed NSGA-II for multi-objective land allocation (MOLA) [J]. Geo-spatial Information Science, 2018, 21 (4): 273-287.

[59] Rathnayake U. Review of binary tournament constraint handling technique in NSGA II for optimal control of combined sewer systems [J]. Journal of Information and Optimization Sciences, 2016, 37 (1): 37-49.

[60] Rathnayake U S, Tanyimboh T T. Optimal control of combined sewer systems using SWMM 5.0 [J]. Transactions on the Built Environment, 2012, 122: 87-96.

[61] Zhou Hongbiao. Optimal control of wastewater treatment process based on constrained multi-objective backbone particle swarm [J]. Journal of Electronic Measurement and Instrument, 2017, 31 (9): 1488-1498.

[62] García J L L, Monroy R, Hernández V A S, et al. COARSE-EMOA: An indicator-based evolutionary algorithm for solving equality constrained multi-objective optimization problems [J]. Swarm and Evolutionary Computation, 2021, 67: Article ID 100983.

[63] Scimemi G F, Rizzo S. An hypervolume based constraint handling technique for multi-objective optimization problems [C/OL]/ Proc of the 20th Congr. Ital. Assoc. Theor. Appl. Mech, 2011. (2011) [2022-03-31]. http://hdl.handle.net/10447/77182.

[64] Said R, Bechikh S, Louati A, et al. Solving combinatorial multi-objective bi-level optimization problems using multiple populations and migration schemes [J]. IEEE Access, 2020, 8: 141674-141695.

[65] Guo Jianmei, Liang Jiahui, Shi Kai, et al. SMTIBEA: a hybrid multi-objective optimization algorithm for configuring large constrained software product lines [J]. Software & Systems Modeling, 2019, 18 (2): 1019-1029.

[66] Antonio L M, Coello C A C. Indicator-based cooperative coevolution for multi-objective optimization [C]// Proc of IEEE Congress on Evolutionary Computation (CEC). IEEE, 2016: 991-998.

[67] BenMansour I, Alaya I, Tagina M. Indicator weighted based multi-objective approach using self-adaptive neighborhood operator [J]. Procedia Computer Science, 2021, 192: 338-347.

[68] Mansour I B, Alaya I. Indicator based ant colony optimization for multi-objective knapsack problem [J]. Procedia Computer Science, 2015, 60: 203-212.

[69] Yuan Jiawei, Liu Hailin, Ong Y S, et al. Indicator-based evolutionary algorithm for solving constrained multi-objective optimization problems [J]. IEEE Trans on Evolutionary Computation, 2021: 1.

[70] Liu Zhizhong, Wang Yong, Wang Bingchuan. Indicator-based constrained multi-objective evolutionary algorithms [J]. IEEE Trans on Systems, Man, and Cybernetics: Systems, 2019, 51 (9): 5414-5426.

[71] Wang Yibo, Hotz L. Optimal feature selection via evolutionary algorithms and constraint solving [C]// Proc of 18th International Configuration Workshop. 2016: 97.

[72] Shi Kai, Yu Huiqun, Guo Jianmei, et al. A parallel portfolio approach to configuration optimization for large software product lines [J]. Software: Practice and Experience, 2018, 48 (9): 1588-1606.

[73] Li Yahui, Li Yang. Security-constrained multi-objective optimal power flow for a hybrid AC/VSC-MTDC system with lasso-based contingency filtering [J]. IEEE Access, 2019, 8: 6801-6811.

[74] Tangpattanakul P, Jozefowiez N, Lopez P. A multi-objective local search heuristic for scheduling earth observations taken by an agile satellite [J]. European Journal of Operational Research, 2015, 245 (2): 542-554.

[75] Miyakawa M, Sato H, Sato Y. Directed mating in decomposition-based MOEA for constrained many-objective optimization [C]// Proc of Genetic and Evolutionary Computation Conference. 2018: 721-728.

[76] Elarbi M, Bechikh S, Said L B. On the importance of isolated solutions in constrained decomposition-based many-objective optimization [C]// Proc of Genetic and Evolutionary Computation Conference. 2017: 561-568.

[77] Fan Zhun, Fang Yi, Li Wenji, et al. MOEA/D with angle-based constrained dominance principle for constrained multi-objective optimization problems [J]. Applied Soft Computing, 2019, 74: 621-633.

[78] Asafuddoula M, Ray T, Sarker R, et al. An adaptive constraint handling approach embedded MOEA/D [C]// Proc of IEEE Congress on Evolutionary Computation. IEEE, 2012: 1-8.

[79] Fan Zhun, Li Hui, Wei Caimin, et al. An improved epsilon constraint handling method embedded in MOEA/D for constrained multi-objective optimization problems [C]// Proc of IEEE Symposium Series on Computational Intelligence (SSCI). IEEE, 2016: 1-8.

[80] Martinez S Z, Coello C A C. A multi-objective evolutionary algorithm based on decomposition for constrained multi-objective optimization [C]// Proc of IEEE Congress on Evolutionary Computation (CEC). IEEE, 2014: 429-436.

[81] Zhu Qingling, Zhang Qingfu, Lin Qiuzhen. A constrained multi-objective evolutionary algorithm with detect-and-escape strategy [J]. IEEE Trans on Evolutionary Computation, 2020, 24 (5): 938-947.

[82] Zapotecas-Martínez S, Ponsich A. Constraint handling within MOEA/D through an additional scalarizing function [C]// Proc of Genetic and Evolutionary Computation Conference. 2020: 595-602.

[83] Peng Chaoda, Liu Hailin. A decomposition-based constraint-handling technique for constrained multi-objective optimization [C]// Proc of IEEE/WIC/ACM International Conference on Web Intelligence Workshops. ACM, 2016: 129-134.

[84] Wang Bingchuan, Li Hanxiong, Zhang Qingfu, et al. Decomposition-based multi-objective optimization for constrained evolutionary optimization [J]. IEEE Trans on Systems, Man, and Cybernetics: Systems, 2018, 51 (1): 574-587.

[85] Cai Xinye, Mei Zhiwei, Fan Zhun, et al. A constrained decomposition approach with grids for evolutionary multi-objective optimization [J]. IEEE Trans on Evolutionary Computation, 2017, 22 (4): 564-577.

[86] Cai Xinye, Xia Chao, Zhang Qingfu, et al. The collaborative local search based on dynamic-constrained decomposition with grids [J]. Swarm and Evolutionary Computation, 2020, 52: Article ID 100600.

[87] Chen Jie, Li Juan, Bin Xin. DMOEA- C: Decomposition-based multi-objective evolutionary algorithm with the -constraint framework [J]. IEEE

Trans. Evolutionary Computation. 2017, 21 (5): 714-730.

[88] Liu Hailin, Peng Chaoda, Gu Fangqing, et al. A constrained multi-objective evolutionary algorithm based on boundary search and archive [J]. International Journal of Pattern Recognition and Artificial Intelligence, 2016, 30 (01): Article ID 1659002.

[89] Zhu Yongsheng, Wang Jie, Qu Boyang. Multi-objective economic emission dispatch considering wind power using evolutionary algorithm based on decomposition [J]. International Journal of Electrical Power & Energy Systems, 2014, 63: 434-445.

[90] Biswas P P, Suganthan P N, Mallipeddi R, et al. Multi-objective optimal power flow solutions using a constraint handling technique of evolutionary algorithms [J]. Soft Computing, 2020, 24 (4): 2999-3023.

[91] Zhou Hongbiao, Qiao Junfei. Multi-objective optimal control for wastewater treatment process using adaptive MOEA/D [J], Applied Intelligence, 2019, 49 (3): 1098-1126.

[92] Li Wenji, Fan Zhun, Cai Xinye, et al. Design optimization of MEMS using constrained multi-objective evolutionary algorithm [C]// Proc of Companion Publication of the Annual Conference on Genetic and Evolutionary Computation. 2014: 1175-1180.

[93] Hu Wanzhe, Fathi M, Pardalos P M. A multi-objective evolutionary algorithm based on decomposition and constraint programming for the multi-objective team orienteering problem with time windows [J]. Applied Soft Computing, 2018, 73: 383-393.

[94] Li Junqing, Tao Xinrui, Jia Baoxian, et al. Efficient multi-objective algorithm for the lot-streaming hybrid flowshop with variable sub-lots [J]. IEEE Trans on Cybernetics, 2019, 51 (5): 2639-2650.

Note: Figure translations are in progress. See original paper for figures.