Available on IEEE Xplore
西南大學 SOUTHWEST UNIVERSITY

IEEE/SWU Artificial Intelligence Science and Engineering

https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=10899809

# Decomposition for Large-Scale Optimization Problems: An Overview

Thai Doan CHUONG[1], Chen LIU[2], Xinghuo YU[2]

[1] Department of Mathematics, Brunel University of London, London UB8 3PH, United Kingdom
[2] School of Engineering, RMIT University, Melbourne 3000, Australia

**ARTICLE INFO**

***Keywords:***
decomposition methods; nonlinear optimization; large-scale problems; computational intelligence

***Citation:***
T D CHUONG, C LIU, X H YU. Decomposition for Large-Scale Optimization Problems: An Overview [J]. *Artificial Intelligence Science and Engineering*, 2025, 1(3): 157-174.

***Publishing Editor:***
Mengyuan ZHOU (Email: zhoumengyuan@swu.edu.cn)

**ABSTRACT**

Formalizing complex processes and phenomena of a real-world problem may require a large number of variables and constraints, resulting in what is termed a large-scale optimization problem. Nowadays, such large-scale optimization problems are solved using computing machines, leading to an enormous computational time being required, which may delay deriving timely solutions. Decomposition methods, which partition a large-scale optimization problem into lower-dimensional subproblems, represent a key approach to addressing time-efficiency issues. There has been significant progress in both applied mathematics and emerging artificial intelligence approaches on this front. This work aims at providing an overview of the decomposition methods from both the mathematics and computer science points of view. We also remark on the state-of-the-art developments and recent applications of the decomposition methods, and discuss the future research and development perspectives.

In practice, when formalizing an optimization problem of real-world complex processes and phenomena, one may need to use a large number of variables and constraints for their description, resulting in the so-called large-scale optimization problems, such as large-scale linear programming problems or mixed integer linear optimization problems (*e.g.*, Refs.[1-2]). Large-scale optimization problems are usually solved using computing machines. Because of the large number of variables (and thus dimensions), solving large-scale optimization problems requires enormous computational memory, which hinders the timely delivery of solutions. Therefore, "Divide and Conquer" is one of the crucial schemes and techniques for handling large-scale optimization problems[3]. The developments of decomposition methods or how to partition a large-scale optimization problem into subproblems of lower-dimension that allow one to derive a timely solution in a reasonable time have emerged as an important research topic in both applied mathematics and computer science. This overview will look into their separate developments, pinpoint their strengths and weaknesses, identify mutual benefits, and foreshadow their future perspectives.

Decomposition methods are an efficient way to solve a large-scale programming problem by dividing the problem into subproblems which are easier to implement and/or are reduced in size. There has been extensive research done in applied mathematics and operations research, which will be summarized in the following sections.

A general optimization problem can be stated as

$$\max_{\boldsymbol{x}\in\mathbb{R}^n}\{f(\boldsymbol{x})\,|\,\boldsymbol{x}\in\Omega,\ g_i(\boldsymbol{x})\leqslant 0,\ i=1,\cdots,p,\ h_j(\boldsymbol{x})=0,\ j=1,\cdots,q\} \tag{1}$$

where $\boldsymbol{x}$ is the decision vector, $f:\mathbb{R}^n\to\mathbb{R}$ is the objective function, $g_i:\mathbb{R}^n\to\mathbb{R}, i=1,\cdots,p$ and $h_j:\mathbb{R}^n\to\mathbb{R}, j=1,\cdots,q$ are the constraint functions, and

**Received:** 2025-07-25; **Received in Revised Form:** 2025-08-20; **Accepted:** 2025-08-20.
**Corresponding Author:** Xinghuo YU.
**Financial Support:** The Australian Research Council (DP200101197, DP230101107).
**Conflict of Interest Disclosure:** The authors report there are no competing interests to declare. 
**DOI:** 10.23919/AISE.2025.000012

$\Omega \subset \mathbb{R}^n$ is a geometric constraint set. In large-scale optimization, the main challenge is maintaining algorithmic efficiency and manageable memory usage as the dimension $n$ and the number (and complexity) of constraints increase.

Having established the motivation and relevance of large-scale optimization in contemporary applications, the discussion now turns to a structured examination of decomposition methods, beginning with their fundamental principles and classical theoretical underpinnings.

# 1 Linear and Nonlinear Optimization

## 1.1 Linear Optimization Problems

The decomposition method can be dated back to the work of DANTZIG and WOLFE in 1960[4], where the authors proposed a decomposition approach to solve the special optimization problem of (1), which is a linear programming problem defined by

$$\max_{t\in\mathbb{R},\, \boldsymbol{x}\in\mathbb{R}^n,\, \boldsymbol{y}\in\mathbb{R}^m} \{t \mid \boldsymbol{P}_0 t + \bar{\boldsymbol{A}}_1\boldsymbol{x} + \bar{\boldsymbol{A}}_2\boldsymbol{y} = \bar{\boldsymbol{b}},\ \boldsymbol{A}_1\boldsymbol{x} = \boldsymbol{b}_1, \boldsymbol{A}_2\boldsymbol{y} = \boldsymbol{b}_2,\ \boldsymbol{x} \geqslant 0, \boldsymbol{y} \geqslant 0\} \tag{2}$$

where $\bar{\boldsymbol{A}}_i, \boldsymbol{A}_i, i = 1,2$ are matrices and $\boldsymbol{P}_0, \bar{\boldsymbol{b}}, \boldsymbol{b}_i, i = 1,2$ are vectors.

At that time, most linear programs could be solved using the revised simplex algorithm, and so Dantzig-Wolfe decomposition is based on column generation (*e.g.*, Ref.[5]), which involves the master problem and a restricted master problem with fewer variables. In this method, at each step, most columns/variables are not in the basis. Hence, a master problem containing at least the currently active columns (the basis) uses subproblems to generate columns for entry into the basis so that their inclusion improves the objective function.

## 1.2 Column Generation Approaches

Column generation (see, *e.g.*, Ref.[5]) is an effective technique for solving a large-scale linear programming problem by decomposing it into the master problem and the restricted master problem with a reduced number of variables. The restricted master problem is heuristically constructed by selecting a subset of columns from the original problem or introducing artificial columns. This restricted master problem is solved and its dual solutions are used to define a subproblem, which is often referred to as the oracle. The oracle is then solved to identify new columns with negative reduced costs, which are added to the restricted master problem. This iterative process continues until a predefined stopping criterion is met, at which point the solution of the latest restricted master problem provides an optimal or near-optimal solution to the original problem[6]. Column generation enables one to find solutions to large-scale linear programs by iteratively solving smaller subproblems, thereby reducing the number of variables or constraints to be considered at each step. This makes it a promising approach for tackling large-scale linear and mixed-integer programming problems. For a brief review of the column generation method and its variants, the reader is referred to Ref.[5].

Recent studies illustrate the versatility and effectiveness of column generation and related decomposition methods in various application domains. For instance, GAMBOA *et al*.[7] investigated decomposition techniques for two-stage Wasserstein-based distributionally robust optimization problems by using Multi-cut Benders decompositions and regularized versions on unit commitment problems with high-dimensional uncertainty vectors. In the context of home care scheduling, GRENOUILLEAU *et al*.[8] proposed a pattern-based logic-based Benders decomposition by integrating it with matheuristics based on large neighborhood search to maximize patient acceptance and maintain service consistency. Similarly, LEAO *et al*.[9] addressed the integrated one-dimensional cutting-stock and lot-sizing problem in paper manufacturing. They applied Dantzig-Wolfe decomposition and improved the column generation procedure by incorporating an adaptive large neighborhood search and demonstrated effectiveness in real-world instances.

The authors in Ref.[10] developed a structured stabilized Dantzig-Wolfe decomposition technique for large-scale linear programs to improve computational performance, particularly in multicommodity capacitated network design. In the area of electricity markets, SAGASTIZABAL[11] explored uncertainty and strategic interactions in capacity investment planning by employing Lagrangian relaxation and Benders decomposition variants to model these complex dynamics. Linear programming based decomposition was also prominent in the work of KUNNUMKAL *et al*.[12], who addressed inventory distribution problems by decomposing a dynamic programming model, while ZHANG *et al*.[13] developed an exact column generation algorithm for scheduling in seru production systems by using Dantzig-Wolfe decomposition to solve linear relaxations of the model efficiently.

In a distributed optimization setting, TONBARI *et al*.[14] introduced a fully decentralized Dantzig-Wolfe decomposition algorithm based on the consensus alternative direction method of multipliers, and YAZDANI *et al*.[15] proposed a decomposition-based coevolutionary framework to address the scalability of dynamic optimization. FLORES *et al*.[16] considered long-term capacity planning and short-term operational decisions in power-intensive industrial plants. They extended a multiscale process network model and used a column generation approach to solve the resulting large-scale mixed integer linear programs without the need for branching.

In Ref. [17], the authors proposed a column generation-based heuristic for pricing and extreme point placement, which outperformed existing approaches and provided new lower bounds for the non-rotational variant. FARHAM *et al.*[18] developed a branch-and-price algorithm for the location-routing problem with time windows, combining set-partitioning models and dynamic programming with acceleration strategies. Their approach achieved strong computational results on both benchmark and large-scale instances. Similarly, RIERA *et al.*[19] reformulated the team orienteering arc routing problem as a set-partitioning problem using a customer-on-vertex representation. Their column generation algorithms proved particularly effective in cases, where traditional branch-and-cut methods failed to provide tight dual bounds due to knapsack-type constraints.

### 1.3 Nonlinear Optimization Problems

Benders introduced a decomposition in 1962[20] to solve a mixed-variable programming problem, which is a particular nonlinear problem of (1) defined by

$$\max_{\boldsymbol{x}\in\mathbb{R}^p,\,\boldsymbol{y}\in\mathbb{R}^q} \{\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} + f(\boldsymbol{y}) \mid \boldsymbol{A}\boldsymbol{x} + F(\boldsymbol{y}) \leqslant \boldsymbol{b}, \boldsymbol{y} \in S\} \tag{3}$$

where $S \subset \mathbb{R}^q$, $\boldsymbol{A}$ is an $(m \times p)$ matrix, $f$ is a scalar function and $F$ is a vector function on $S$, $\boldsymbol{c} \in \mathbb{R}^p$ and $\boldsymbol{b} \in \mathbb{R}^m$.

The Benders approach was extended by GEOFFRION in 1972[21] to address a more general problem of the form:

$$\max_{\boldsymbol{x}\in\mathbb{R}^p,\,\boldsymbol{y}\in\mathbb{R}^q} \{f(\boldsymbol{x},\boldsymbol{y}) \mid G(\boldsymbol{x},\boldsymbol{y}) \geqslant 0, \boldsymbol{x} \in S_1, \boldsymbol{y} \in S_2\} \tag{4}$$

where $\boldsymbol{y}$ is a vector of complicating variables in the sense that (4) is an easier optimization problem in $\boldsymbol{x}$ when $\boldsymbol{y}$ is temporarily held fixed, and $f$ is a scalar function and $G$ is a vector function on $S_1 \times S_1$ with $S_1 \subset \mathbb{R}^p, S_2 \subset \mathbb{R}^q$.

The projection (or partitioning) on $\boldsymbol{y}$ is defined by

$$\max_{\boldsymbol{y}\in\mathbb{R}^q} \{v(\boldsymbol{y}) \mid \boldsymbol{y} \in S_2 \cap V\} \tag{5}$$

where $V := \{\boldsymbol{y} \in \mathbb{R}^q \mid \exists \boldsymbol{x} \in S_1, G(\boldsymbol{x},\boldsymbol{y}) \geqslant 0\}$ and

$$v(\boldsymbol{y}) := \sup_{\boldsymbol{x}\in\mathbb{R}^p} \{f(\boldsymbol{x},\boldsymbol{y}) \mid G(\boldsymbol{x},\boldsymbol{y}) \geqslant 0, \boldsymbol{x} \in S_1\} \tag{6}$$

Note that $v(\boldsymbol{y})$ is the optimal value of problem (4) for fixed $\boldsymbol{y}$ and that, by the designation of $\boldsymbol{y}$ as complicating variables, evaluating $v(\boldsymbol{y})$ is much easier than solving problem (4) itself.

### 1.4 Solution Relations Between Problem (4) and Problem (5)

Theorem 2.1 of Ref. [21] states that if $(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}})$ is an optimal solution of the original problem (4), then $\bar{\boldsymbol{y}}$ is an optimal solution of the partitioning problem (5). Conversely, if $\bar{\boldsymbol{y}}$ is an optimal solution of the partitioning problem (5) and $\bar{\boldsymbol{x}}$ achieves the supremum in Eq.(6) with $\boldsymbol{y} = \bar{\boldsymbol{y}}$, then $(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}})$ is an optimal solution of the original problem (4).

A price-and-verify algorithm based on the Dantzig-Wolfe decomposition was proposed in Ref. [22] to reformulate and solve a recursive circle packing problem within the logistics operations of the tube industry. In Ref. [23], the Benders decomposition algorithm was applied to solve the problem of locating items in carousel systems, where a carousel storage problem is reformulated as a mixed-integer program. An extension of sparse group Lasso regularization was proposed in Ref. [24] to calculate clusters of generalized linear models in the area of proportional hazards problems with right-censoring. The authors in Ref. [25] developed a simultaneous Magnanti-Wong method[26] to accelerate Benders decomposition[20] and handle a metropolitan container transportation problem that helps leverage transportation effectively from a least-cost perspective.

Recent studies highlight the adaptability of decomposition methods in solving nonlinear and mixed-integer nonlinear programming (MINLP) problems across diverse application domains. OLSEN *et al.*[27] investigated a multicommodity flow formulation of the vehicle scheduling problem and proposed a three-phase solution approach. This approach comprises an exact method for solving the base problem without range constraints, innovative flow decomposition techniques, and a novel algorithm to manage electric vehicle charging constraints, recently developed in Ref. [28] for electric bus scheduling. Similarly, VASQUEZ *et al.*[29] addressed a traveling salesman problem with drone by decomposing truck and drone operations. Their Benders-type algorithm incorporated valid inequalities and novel optimality cuts such as t-shortcut and t-reduction to achieve strong computational results.

Logic-based Benders decomposition has proven particularly effective in complex scheduling and routing contexts. BRUNI *et al.*[30] formulated a mixed-integer linear program for drone-assisted package delivery and developed a logic-based Benders decomposition enhanced with relaxations. ZHANG *et al.*[31] considered a scheduling problem in seru production with learning effects and sequence-dependent setups. They applied logic-based Benders decomposition to reformulate the problem into a set-partitioning model. Similarly, MICHELS *et al.*[32] solved the type-2 multi-manned assembly line balancing problem by combining decomposition techniques with combinatorial Benders cuts to find optimal solutions for large real-life instances in automotive manufacturing.

Benders decomposition has also been adapted for resilient infrastructure and cyber-physical systems. SHELAR *et al.*[33] proposed a bilevel mixed-integer second-order cone program to assess the resilience of electricity distribution networks against cyber-physical attacks. To mitigate computational complexity and balance accuracy and efficiency, they improved classical generalized Benders decomposition by modifying Benders cuts based on selected dual variables. MOKHTAR *et al.*[34] studied the uncapacitated 2-allocation p-hub median problem to enhance network survivability. They used a modified Benders decomposition algorithm that transforms subproblems into minimum-cost network flow models to locate efficient solutions of large-scale instances. This approach was recently extended in Ref. [35] to assess the viability of regional connectivity strategies.

In the setting of stochastic programming, the decomposition has played a central role. The authors in Ref. [36] solved large-scale lot-sizing problems under uncertain demand using Benders decomposition integrated with stochastic linear programming. Their method accelerated subproblem solutions and achieved linear scaling with the number of scenarios. In Ref. [37], the authors studied enhancements to dual decomposition for two-stage stochastic mixed-integer programs, including Benders-like cuts and a new interior-point cutting-plane method with proven finite convergence to optimal dual solutions. GUIGUES[38] developed sampling-based decomposition algorithms for multistage stochastic convex programs and derived cutting-plane formulas for efficient stochastic dual dynamic programming. The approach guaranteed convergence even under interstage dependent stochastic processes.

Decomposition techniques have also been employed to solve nonlinear and conic problems. In Ref. [39], the authors proposed a variable partitioning strategy for Benders decomposition, targeting a class of MINLPs including fixed-charge multicommodity network design with congestion effects. Their method simplified branch-and-bound procedures by transforming each node into a conic quadratic subproblem with only continuous variables. SCHMIDT *et al.*[40] presented global optimization methods for mixed-integer problems with Lipschitz nonlinear constraints, including settings of inexact function evaluations or uncertain Lipschitz constants. Their algorithms demonstrated robust convergence and performance on gas transport networks and academic benchmarks.

To address common challenges in duality and decomposition, VUJANIC *et al.*[41] proposed modifications to the primal problem to ensure feasibility of dual solutions in mixed-integer optimization. Their approach showed improved results in large-scale power systems problems. BECK *et al.*[42] focused on the minimization of strongly convex functions combined with nonsmooth convex terms. They derived primal versions of dual-based block descent algorithms, linked convergence rates, and validated the approach through total variation-based image denoising problems. In Ref. [43], the authors introduced a decomposition branching method that integrates branch-and-bound with decompositions. Their results on weighted set covering and regionalized p-median problems showed superior performance compared to some commercial solvers and automatic Dantzig-Wolfe decomposition.

The decomposition methods have proven essential for solving large-scale optimization problems in both linear and nonlinear domains including MINLP and conic problems. Techniques like Dantzig-Wolfe decomposition, Benders decomposition, column generation, and hybrid variants continue to deliver scalable solutions across diverse fields such as energy, logistics, healthcare, and manufacturing.

However, challenges remain, particularly in scalability, convergence performance in non-convex settings, and sensitivity to algorithmic details and parameter tuning. In the next section, we will examine the effectiveness of decomposition methods in more structured optimization frameworks, including piecewise-linear and convex problems.

## 2 Alternative Structured Optimization Problems

### 2.1 Structured Convex and Piecewise-Linear Decomposition Methods

This section explores the decomposition procedures for other structured optimization problems that can be reformulated and cast into the problem (1). The notes in Ref. [44] provided some decomposition approaches for unconstrained and constrained optimization models with structured functions as follows:

$$\min_{\boldsymbol{x}_1\in\mathbb{R}^{p_1}, \boldsymbol{x}_2\in\mathbb{R}^{p_2}, \boldsymbol{y}\in\mathbb{R}^q} \{f_1(\boldsymbol{x}_1,\boldsymbol{y})+f_2(\boldsymbol{x}_2,\boldsymbol{y})\} \tag{7}$$

where $f_1$ and $f_2$ are piecewise-linear convex functions and $\boldsymbol{x}_1, \boldsymbol{x}_2$ and $\boldsymbol{y}$ are variables. Here, $\boldsymbol{y}$ is called a complicating variable as when it is fixed, the problem splits into two subproblems independently. In this case, the original problem (7) is equivalent to the following master problem:

$$\min_{\boldsymbol{y}\in\mathbb{R}^q} \{\phi_1(\boldsymbol{y})+\phi_2(\boldsymbol{y})\} \tag{8}$$

where $\phi_1(\boldsymbol{y}) := \inf_{\boldsymbol{x}_1\in\mathbb{R}^{p_1}} f_1(\boldsymbol{x}_1,\boldsymbol{y})$ and $\phi_2(\boldsymbol{y}) := \inf_{\boldsymbol{x}_2\in\mathbb{R}^{p_2}} f_1(\boldsymbol{x}_2,\boldsymbol{y})$ are the subproblems. In this way, the authors extended the corresponding decomposition methods to a more general constrained problem

$$\min_{\boldsymbol{x}_i\in\mathbb{R}^{p_i}, \boldsymbol{y}_i\in\mathbb{R}^{q_i}, \boldsymbol{z}\in\mathbb{R}^q} \{f_1(\boldsymbol{x}_1,\boldsymbol{y}_1)+\cdots+f_m(\boldsymbol{x}_m,\boldsymbol{y}_m) \mid (\boldsymbol{x}_i,\boldsymbol{y}_i)\in C_i, \boldsymbol{y}_i=\boldsymbol{E}_i\ \boldsymbol{z}, i=1,\cdots,m\} \tag{9}$$

where $f_i, i=1,\cdots,m$ are functions, $\boldsymbol{z}$ is the vector of net variables, $\boldsymbol{E}_i, i=1,\cdots,m$ are matrices and $C_i, i=1,\cdots,m$ are the feasible sets of subproblems.

The problem (9) is equivalent to the following master problem

$$\min_{\boldsymbol{z}\in\mathbb{R}^q} \{\phi_1(\boldsymbol{E}_1\ \boldsymbol{z})+\cdots+\phi_m(\boldsymbol{E}_m\ \boldsymbol{z})\}$$

where $\phi_i(\boldsymbol{y}_i) := \inf_{\boldsymbol{x}_i\in\mathbb{R}^{p_i}} \{f_1(\boldsymbol{x}_1,\boldsymbol{y}_i) \mid (\boldsymbol{x}_i,\boldsymbol{y}_i)\in C_i\}, i=1,\cdots,m$ are the subproblems.

Recently, the authors in Ref. [45] addressed the scheduling day-ahead problem of an energy community operating by incorporating a novel Benders dual decomposition proposed in Ref. [46] that can generate stronger feasibility and optimality cuts compared with the classical Benders methods. We refer to an approach called subproblem cuts proposed in Ref. [47], which helps avoid the generation of redundant columns in the column generation method and improve the computational performance. An application to network revenue management problems was presented in Ref. [48], where the authors used the Dantzig-Wolfe decomposition principle[4] for the analytical framework by exploring the structure of approximate linear programs with column generation subproblems to generate the constraints. The obtained results show that solving the reduced programs gives tighter upper bounds on total expected revenues. A decomposition method for solving lasso problems involving zero-sum constraints in high-dimensional spaces can be found in Ref. [49], where a global convergence of the proposed schemes to optimal solutions is guaranteed.

Other recent studies further demonstrate the adaptability and effectiveness of decomposition methods in more structured settings through various application domains. RAHMANIANI *et al.*[50] presented an advanced Benders decomposition algorithm to solve large-scale multicommodity capacitated network design problems under demand uncertainty. To enhance computational efficiency, they incorporated cutting planes, partial decomposition, and warm-start strategies that significantly outperformed existing algorithms on benchmark datasets.

Hybrid decomposition frameworks have also shown promise in solving complex nonlinear and stochastic optimization problems. The authors in Ref. [51] developed a joint decomposition approach that blends the Lagrangian decomposition with the generalized Benders decomposition to solve multi-scenario non-convex MINLPs without traditional branch-and-bound methods. When applied to stochastic process network design problems, their method achieved substantial reductions in solution time and mastered the problem complexity compared to state-of-the-art global solvers.

Decomposition-based heuristics and metaheuristics are effective for tackling real-world production and scheduling scenarios. In Ref. [52], the authors analyzed a manufacturing system composed of two interacting subsystems that are relevant to low-volume aerospace production. They proposed a hybrid random-key genetic algorithm augmented with list scheduling, exact methods, and buffer-aware heuristics to optimize labor, inventory, makespan, and tardiness. Computational results validated the practical effectiveness of their decomposition-driven framework.

Decomposition techniques have also been used effectively in complex scheduling domains. TRAN *et al.*[53] tackled the unrelated parallel machine scheduling problem, accounting for both sequence and machine dependent setup times. They proposed two exact decomposition-based methods: Benders decomposition and branch-and-check methods using a mixed-integer programming master problem. In Ref. [54], the authors introduced a stochastic parallel successive convex approximation method for general non-convex stochastic sum-utility problems. Their approach decomposed the problem into strongly convex subproblems that can be solved in parallel with superior empirical performance relative to conventional stochastic gradient approaches, particularly in multi-agent network environments.

### 2.2 Lagrangian-Based Decomposition Methods

Lagrangian-based decomposition methods involve first dualizing some constraints into the objective function and then using Lagrange multipliers to penalize their violation.

Consider a convex optimization problem

$$\min_{\boldsymbol{x}_i\in\mathbb{R}^{n_i}}\{f_1(\boldsymbol{x}_1)+\cdots+f_m(\boldsymbol{x}_m)\mid \boldsymbol{B}_1\boldsymbol{x}_1+\cdots+ \boldsymbol{B}_m\boldsymbol{x}_m=\boldsymbol{b},\boldsymbol{x}_i\in C_i,i=1,\cdots,m\} \tag{10}$$

where $f_i:\mathbb{R}^{n_i}\to\mathbb{R}, i=1,\cdots,m$ are convex functions, $C_i\subset\mathbb{R}^{n_i}, i=1,\cdots,m$ are non-empty convex sets, $\boldsymbol{B}_i, i=1,\cdots,m$ are $p\times n_i$ matrices and $\boldsymbol{b}\in\mathbb{R}^p$. We define the (augmented) Lagrangian for the problem (10) as

$$L(\boldsymbol{x}_1,\cdots,\boldsymbol{x}_m,\boldsymbol{y})=\sum_{i=1}^m f_i(\boldsymbol{x}_i)+\boldsymbol{y}^{\mathrm{T}}(\boldsymbol{b}-\sum_{i=1}^m\boldsymbol{B}_i\boldsymbol{x}_i)+ \frac{\lambda}{2}\|\boldsymbol{b}-\sum_{i=1}^m\boldsymbol{B}_i\boldsymbol{x}_i\|^2, \boldsymbol{x}_i\in C_i,\ i=1,\cdots,m,\ \boldsymbol{y}\in\mathbb{R}^m$$

with an (augmented) parameter $\lambda\geqslant 0$ and a dual function

$$g(\boldsymbol{y})=\inf_{\boldsymbol{x}_i\in C_i}L(\boldsymbol{x}_1,\cdots,\boldsymbol{x}_m,\boldsymbol{y})$$

Note here that if $\lambda=0$, we obtain the ordinary Lagrangian for problem (10). Now, the dual problem is given by

$$\max_{\boldsymbol{y}\in\mathbb{R}^m} g(\boldsymbol{y}) \tag{11}$$

and a general scheme of augmented Lagrangian decompositions[55] was applied to the dual problem (11) to analyze scenario and nodal decomposition methods for multistage stochastic programs in Ref. [56]. We refer to the reader to Ref. [57] for dual decomposition methods in which the decomposed scheme is ensured by the Lagrangian for a more general framework.

Recent research demonstrates the broad applicability and effectiveness of Lagrangian-based and dual decomposition methods in solving complex optimization problems across various domains. In Ref. [58], the authors developed a semi-proximal augmented Lagrangian decomposition framework for convex composite quadratic conic programming problems with a primal block-angular structure. Their approach, while formulated from the primal side, showed greater computational efficiency when the dual problem was solved using a semi-proximal symmetric Gauss–Seidel-based alternating direction method of multipliers. QUOC *et al*.[59] developed an inexact perturbed path-following algorithm for large-scale separable convex programming problems. Their method accommodated approximate solutions to the primal subproblems and leveraged inexact derivative information within a two-phase algorithmic structure.

Similarly, BAI *et al*.[60] proposed an augmented Lagrangian decomposition method to tackle joint chance-constrained optimization problems under discrete distributions. By reformulating these problems as large-scale mixed-integer programs, their method efficiently managed the randomness in both sides of the chance constraints. The authors in Ref. [61] addressed separable convex minimization problems with linear coupling constraints by proposing an augmented Lagrangian approach with full Jacobian decompositions and logarithmic-quadratic proximal regularizations. The proposed method with both exact and inexact variants allowed one to solve highly parallelizable unconstrained subproblems and ensure global convergence. The convergence behavior of augmented Lagrangian methods was further examined by HE *et al*.[62], who showed that while full Jacobian decomposition can support parallel computation, it can also induce divergence in standard settings.

In the context of proximal and coordinate-based decomposition techniques, TAPPENDEN *et al*.[63] studied separable approximations to the augmented Lagrangian by focusing on the diagonal quadratic approximation method and the parallel coordinate descent method. In parallel, the authors in Ref. [64] proposed proximal-based pre-correction decomposition methods for convex problems with separable structures. Building on earlier work by CHEN and TEBOULLE[65] and HE[66], their algorithms retained the convergence properties of proximal point methods. The proximal method of multipliers also served as the foundation for a general framework developed by SHEFI and TEBOULLE[67]. They proposed two classes of decomposition algorithms within the proximal method of multipliers framework originally formulated by ROCKAFELLAR[68].

Other decomposition-based methods including sparse and branch-and-price techniques have been developed to tackle non-convex and combinatorial optimization challenges. Ref. [69] introduced a penalty decomposition approach for general $\ell_0$ minimization offering convergence guarantees and outperforming traditional $\ell_1$-based methods in various applications, recently developed in Ref. [70] for a nonlinear programming problem with arbitrary abstract constraints. Similarly, Ref. [71] proposed a Benders decomposition extension tailored to semidefinite programs with structured sparsity, effectively leveraging the underlying problem structure. Building on this theme, Ref. [72] developed a modified penalty decomposition method for cardinality-constrained optimization problems by examining both gradient-based and derivative-free implementations. Ref. [73] further advanced this topic by proposing a penalty dual decomposition framework for non-convex problems such as beamforming and matrix factorization with convergence to stationary points.

In particular, branch-and-price methods have been proven to be effective in solving large-scale combinatorial and scheduling problems. For instance, Ref. [74] applied branch-and-price to integrated satellite scheduling by embedding column generation within a branch-and-bound structure. Ref. [75] addressed the two-dimensional vector packing problem with piecewise-linear costs by designing a pricing routine based on dominance rules. Ref. [76] tackled single-machine scheduling problems under periodic maintenance using a set-partitioning model with efficient label-setting algorithms. Furthermore, the branch-and-price algorithm has provided solutions to complex logistics and planning problems: Ref. [77] studied multi-agent production-distribution systems, Ref. [78] focused on emergency unit dispatching under tight deadlines, and Ref. [79] optimized location-route problems in the presence of price-sensitive demands.

Beyond the traditional frameworks, many studies have developed scalable decomposition algorithms for structured optimization. Ref. [80] employed chordal decompositions combined with operator-splitting techniques to solve sparse semidefinite programs and achieve significant scalability improvements. Ref. [81] introduced an alternating direction method of multipliers (ADMM) variant using a semi-linearized Gauss-Seidel scheme for multi-block nonsmooth problems. In energy applications, Ref. [82] used proximal decomposition to coordinate electricity market operations more efficiently than standard ADMM approaches. Ref. [83] developed a distributed framework for optimizing difference-of-convex sum-utility functions with convergence guarantees. Similarly, Ref. [84] proposed a decomposition approach for box-constrained problems based on a violating index.

Numerous specialized decomposition schemes have targeted combinatorial and stochastic problems. Ref. [85] presented pseudo-polynomial formulations and decomposition algorithms for the multiple knapsack problem. Ref. [86] proposed a globally convergent and parallelizable method for solving convex quadratic programs with dense Hessians, motivated by support vector machine training. In service and revenue management, Ref. [87] optimized long-term hotel room allocations through decomposition techniques. Ref. [88] introduced a hybrid of the Benders and Dantzig-Wolfe decompositions for two-stage stochastic supply chain models. Ref. [89] extended the convergence theory to nested decomposition techniques in multistage stochastic convex programming problems.

Recent advances also include projection-based and sampling-driven decomposition techniques. Ref. [90] introduced a dual projection approach for conic feasibility problems with a MATLAB implementation tailored for semidefinite programs. Ref. [91] proposed a distributionally robust stochastic decomposition algorithm for two-stage optimization under uncertainty, while Ref. [92] improved stochastic decomposition techniques for second-stage randomness without relying on arbitrary sampling. To automate the detection of decomposition subproblems, Ref. [93] used community detection methods and Ref. [94] advanced automatic variable decomposition using second-order derivative information for handling high-dimensional spaces.

Decomposition frameworks have enabled significant progress in real-world applications, including energy systems, healthcare, transportation, and bioresource planning. Ref. [95] used approximate decompositions for performance analysis in open queuing networks. Ref. [96] applied decompositions to multistage dynamic networks with stochastic capacity by creating tree-structured subproblems. In infrastructure and energy systems, Refs. [97-99] and Ref. [100] implemented the Dantzig-Wolfe decomposition to optimize district heating, facility layout, efficiency evaluation and power flow, respectively. Decentralized approaches were explored by Refs. [101-102] in grid-level energy systems. In transportation, Ref. [103] developed a space-time decomposition model for urban logistics with time-dependent travel. Other notable applications include parallel decomposition for steel manufacturing[104], electric vehicle charging optimization[105], and biorefinery design using binary approximations[106]. In particular, in the settings of healthcare and scheduling, Refs. [107-108] used the decomposition strategy for vehicle and surgery assignment problems, while Ref. [109] integrated gossip-based distributed computation with column generation to solve large-scale home healthcare routing.

While classical decomposition techniques and frameworks offer strong theoretical guarantees for obtaining or verifying optimal or near-optimal solutions and ensuring convergence for structured optimization models, and have been widely applied in practical contexts, their performance can be limited in highly complex formulations or dynamically changing problem environments. The next section explores how computational intelligence approaches can extend and enhance these traditional techniques.

## 3 Optimization by Computational Intelligence

In contrast to conventional optimization paradigms that classify problems into linear or nonlinear categories and design different algorithms accordingly, computational intelligence approaches adopt a more unified perspective. From this viewpoint, the nature of the objective function or constraints is not the primary concern. Instead, the key idea is to decompose a high-dimensional optimization problem into smaller, more manageable components and solve each each component either iteratively or cooperatively.

This decomposition strategy is particularly effective for large-scale optimization, where handling all decision variables simultaneously is computationally prohibitive. Cooperative coevolution (CC)[110] exemplifies this approach in evolutionary computation. It partitions the $n$ decision variables into $k$ subsets, and evolves a separate population for each subset in a round-robin fashion. During fitness evaluation, each subpopulation's candidate solution is combined with fixed “context vectors” taken from the best solutions of the remaining subpopulations, forming a complete solution for evaluation.

Traditional CC typically assumes a fixed, non-overlapping partition of variables, which can be suboptimal when strong interactions exist among variables. To address this limitation, the overlapping functions decomposition (OFD) approach was recently proposed. This method allows variables to appear in multiple subcomponents based on function overlap, enabling a more accurate representation of interdependencies[111]. A consolidated summary of the principal decomposition methods, along with their key assumptions and representative references, is provided in Table 1.

**Table 1 Summary of Method Classes in Optimization by Computational Intelligence**

| Method Class | Representative Algorithms | Key Assumptions | References |
|---|---|---|---|
| Grouping Variables | Random grouping (RG); dynamic RG (DRG); differential grouping (DG); robust DG; CC with context vectors | Low or sparse inter-block coupling; sufficient reshuffles so interacting variables co-occur; additive separability and bounded noise for DG | Refs. [112-126] |
| Surrogate Modeling | GP/Kriging; radial basis function (RBF); sparse regression; co-/multi-fidelity surrogates to infer interactions and guide search | Surrogates approximate objectives and capture interaction structure; modelability of cross-fidelity correlations | Refs. [127-135] |
| Monotonicity Variable Interaction | Linkage identification by nonlinearity check (LINC)/LI by non-monotonicity detection (LIMD)-style monotonicity tests; pairwise linkage detection; evolutionary algorithms (EA)/CC variants for large-scale and mixed-integer settings | Local monotonicity and sign consistency in small steps; limited noise | Refs. [136-145] |
| Interaction - Driven Decomposition | Fitness-difference and complementary-solution checks; adaptive partitioning to minimize cross-component interactions; overlapping blocks | Sparse/low-order interaction graph; adequate samples to estimate edges; representative context solutions | Refs. [146-155] |

### 3.1 Grouping Variables

CC decomposes a high-dimensional optimization problem into smaller subproblems by grouping interdependent decision variables. Random grouping assigns each decision variable uniformly at random to one of $m$ subcomponents in each of $N$ coevolutionary cycles and has been shown to improve the efficiency of particle swarm optimization (PSO) algorithms on high-dimensional, non-separable problems[112-113]. In this framework, the probability that a given subset of $v$ variables is grouped together in at least $k$ out of $N$ cycles is

$$P(X \geqslant k) = \sum_{r=k}^{N} \binom{N}{r} \left(\frac{1}{m^{v-1}}\right)^{r} \left(1-\frac{1}{m^{v-1}}\right)^{N-r}$$

Building on this idea, dynamic random grouping adapts both the number and size of subcomponents based on intermediate feedback, yielding improved performance on large-scale multi-objective and multi-task problems[114-115]. Random differential grouping further augments random grouping by incorporating finite-difference interaction tests during assignment and has been applied to feature selection in an age-related macular degeneration dataset *via* a bee-colony optimizer[116], while a more recent random differential mechanism has been explored for large-scale feature selection[117]. Delta grouping detects non-separable variables by computing average pairwise function-value differences (delta) and clustering them to determine subcomponent number and size. While it can accurately predict structure in advance, the clustering step introduces high computational cost when many non-separable variables exist[118].

DG detects additive interactions exactly *via* finite-difference tests: two variables $x_p$ and $x_q$ are declared interacting if

$$\Delta_{\delta,x_p}[f](\boldsymbol{x})\big|_{x_p=a,x_q=b_1} \neq \Delta_{\delta,x_p}[f](\boldsymbol{x})\big|_{x_p=a,x_q=b_2}, \quad \forall a,\ b_1 \neq b_2,\ \delta \neq 0$$

where

$$\Delta_{\delta,x_p}[f](\boldsymbol{x}) = f(\cdots,x_p+\delta,\cdots) - f(\cdots,x_p,\cdots)$$

and $f$ is the objective function of the underlying problem. DG achieves the highest precision among grouping mechanisms but its original implementation may require up to 1 001 000 fitness evaluations for a 1 000-dimensional fully separable function[119]. Graph-based DG reduces this cost by constructing an interaction graph and limiting tests to graph edges[120], and divide-and-conquer DG recursively partitions the variable set, further enhancing scalability for large-scale black-box problems[121].

Recursive DG (RDG) refines DG by using a binary-search style procedure on variable subsets: given two disjoint subsets, it computes

$$\begin{aligned}\Delta_1 &= f(\boldsymbol{x}^* + l_1\boldsymbol{u}_1 + l_2(\boldsymbol{u}_3+\boldsymbol{u}_4)) - \\ &\quad f(\boldsymbol{x}^* + l_2(\boldsymbol{u}_3+\boldsymbol{u}_4)), \\ \Delta_2 &= f(\boldsymbol{x}^* + l_1\boldsymbol{u}_1) - f(\boldsymbol{x}^*)\end{aligned}$$

and infers interaction by comparing $\Delta_1$ and $\Delta_2$. On standard benchmarks, RDG reduces the average fitness-evaluation count to about $1.47\times10^4$[122]. An adaptive-threshold variant dynamically tunes the grouping threshold to balance accuracy against cost[123], an overlapping-component extension allows variables to belong to multiple subcomponents by breaking shared links[124], and an efficient RDG (ERDG) introduces heuristic pruning to further lower the computational budget while maintaining or improving decomposition accuracy[125].

DG-based techniques have shown practical value in expensive real-world optimization tasks such as aerodynamic design optimization[156] and flow shop scheduling[126], confirming the efficacy of interaction-based grouping within the CC framework.

### 3.2 Surrogate Modeling

This type of method infers variable interactions *via* building a surrogate or meta-model of the objective function. While meta-modeling is often used for expensive function optimization, there has been little attempt to use it for identifying the problem structure. High-dimensional model representation (HDMR) can be applied to find interacting variables, which are shown on the objective function $f$ as follows:

$$\begin{aligned}f(\boldsymbol{X}) = {}& f_0 + \sum_{i=1}^{N} f_i(x_i) + \sum_{1\leqslant i\leqslant j\leqslant n} f_{ij}(\boldsymbol{x}_{ij}) + \\ & f_{i,\cdots,n}(x_i,\cdots,x_n)\end{aligned}$$

where $f_0$ is the zeroth order term, $f_i(x_i)$ stands for the first-order terms which capture the effect of each variable acting independently, $f_{ij}$ represents the second-order term which depicts the correlated contribution of $x_i$ and $x_j$ and finally, $f_{1,\cdots,n}$ is the nth-order term which captures the joint correlation of all decision variables not covered by all other terms. HDMR contains a finite number of terms and is exact once all terms are included[127].

Early meta-modeling focused on compact representations of high-dimensional functions to reduce simulation costs, using methods like Kriging, RBF and polynomial surface models[128]. Later studies leveraged RBFs for efficient global optimization via tailored improvement criteria[129]. To handle multi-modal objectives in large-scale settings, hybrid models combining Gaussian processes and neural networks were proposed[130], while HDMR-based interaction detection was integrated into cooperative co-evolution to reduce complexity[127]. However, most of these methods relied on static or offline learning. Recent work emphasizes dynamic, interaction-aware surrogate adaptation for high-dimensional black-box optimization[131].

Surrogate-assisted EAs (SAEAs) advanced black-box optimization by improving efficiency and scalability. Clustering-based surrogate allocation enhanced swarm optimization[132], while cooperative frameworks balanced global and local searches[131]. Integrating surrogates with dimensionality reduction further boosted performance[133], leading to a hierarchical optimizer with global, regional, and local models[134]. Applications in engineering, such as antenna design, confirmed their effectiveness using high-order Gaussian processes and Bayesian optimization[135].

Beyond traditional evolutionary and coevolutionary strategies, a promising direction is the integration of machine learning into decomposition for large-scale optimization. Data-driven approaches enhance adaptability and efficiency by using online or reinforcement learning to dynamically guide variable grouping and subproblem prioritization. These adaptive mechanisms allow algorithms to learn problem structures during the search, rather than relying on static decomposition. Surrogate modeling has also been applied in cooperative optimization, where models such as radial basis functions or neural networks approximate the objective function or constraint evaluations[157]. This reduces reliance on expensive function calls by using surrogates for candidate evaluation and selectively confirming results with the true model, significantly decreasing computational cost with minimal accuracy loss. In addition, hybrid frameworks combining machine learning and mathematical optimization are increasingly used, where learned models identify effective decomposition strategies or generate warm-start information to accelerate convergence[158].

Surrogate modeling has evolved from uniform fidelity to multi-fidelity frameworks that balance accuracy and cost. Early hybrid models used coarse approximations for global search and finer models for local search[130]. Adaptive fidelity selection based on optimization stages was later introduced[131], culminating in a three-level architecture combining global RBFs, regional fuzzy clustering, and local refinement[134]. This multi-resolution strategy enables efficient surrogate use throughout the search. Applications under high simulation costs confirmed the robustness and efficiency of hierarchical surrogate models[135].

## 3.3 Monotonicity Variable Interaction

Monotonicity detection[136] was proposed as a method for identifying variable interactions by examining violations of monotonicity conditions through systematic perturbations of the objective function. The monotonicity conditions on the objective function $f$ are defined as follows:

$$\begin{aligned}&\text{if } f(\boldsymbol{s}^{(i)}) > f(\boldsymbol{s}) \text{ and } f(\boldsymbol{s}^{(j)}) > f(\boldsymbol{s})\\&\quad\text{then } f(\boldsymbol{s}^{(ij)}) > f(\boldsymbol{s}^{(i)}) \text{ and } f(\boldsymbol{s}^{(ij)}) > f(\boldsymbol{s}^{(j)})\end{aligned} \tag{12}$$

$$\begin{aligned}&\text{if } f(\boldsymbol{s}^{(i)}) < f(\boldsymbol{s}) \text{ and } f(\boldsymbol{s}^{(j)}) < f(\boldsymbol{s})\\&\quad\text{then } f(\boldsymbol{s}^{(ij)}) < f(\boldsymbol{s}^{(i)}) \text{ and } f(\boldsymbol{s}^{(ij)}) < f(\boldsymbol{s}^{(j)})\end{aligned} \tag{13}$$

where $\boldsymbol{s}^{(\cdot)}$ denotes a candidate solution vector perturbed at the index specified in the bracket. The linkage identification by non-monotonicity detection algorithm evaluates whether conditions (12) and (13) are violated in a randomly initialized population. Variables $i$ and $j$ are considered to be interacting if either of the two conditions fails to hold.

Building on early monotonicity-based detection, later work has enhanced accuracy and efficiency in identifying non-separable variables. A framework called CC with variable interaction learning (CCVIL)[137] addressed fixed-group limitations by starting with independent variables and merging pairs that violate separable behavior, though it relies on heuristic sampling and is sensitive to sampled representative solutions. To improve robustness, the cooperative particle swarm optimizer with statistical variable interdependence learning (CPSO-SL)[138] introduced a probabilistic model estimating interdependencies across samples, enabling overlapping groups but increasing computation cost. A theoretical study[139] formalized monotonicity testing with sublinear query complexity and matching lower bounds, deepening understanding of interaction detection. To balance scalability and accuracy, a recursive block partitioning method[140] was proposed, avoiding exhaustive comparisons, using early pruning and integrating with differential evolution for reduced overhead.

To address overlapping group structures overlooked by traditional methods, Ref. [141] proposed a perturbation-based approach that detects shared variable dependencies across groups, improving decomposition accuracy in complex problems but incurring high computational cost. Extending this, Ref. [142] unified monotonicity-inspired perturbation techniques into a general linkage learning protocol suited for overlapping, hierarchical and adaptive structures, emphasizing structured decomposition over standalone monotonicity testing. Separately, the mixed monotonic programming (MMP) framework[143] accelerated global optimization by exploiting latent monotonicity without reformulation, using tailored branch-and-bound algorithms. Geometrically, Ref. [144] compared local search and linear programming-based methods over simplicial domains, showing that local strategies often offer a better accuracy-efficiency trade-off in low-dimensional subspaces. Finally, a relaxed monotonicity-inspired line search was proposed in Ref. [145] for stochastic and overparameterized problems, enabling stable convergence in deep learning by allowing partial monotonicity.

### 3.4 Interaction-Driven Decomposition

The methods reviewed are based on adaptively rearranging decision variables to minimize a squared error function, aiming to reduce the interactions among subcomponents[146]. The underlying principle is that, for a partially separable function, the discrepancy between the original objective function and the sum of its non-separable subfunctions should ideally vanish. This motivates the minimization of the following expression:

$$\min_{\boldsymbol{x}\in\mathbb{R}^m}\left[f(\boldsymbol{x})-\sum_{i=1}^{m}f_i(\boldsymbol{x}_i)\right]^2$$

which can be easily rewritten as a special case of problem (1). Under the black-box optimization setting, both the number and the size of these subcomponents are unknown. To address this, a uniform decomposition structure is assumed in Ref. [146], where the decision variables are partitioned into $k$ components of equal dimension $d$. The purpose becomes finding a permutation of variables that minimizes the following discrepancy:

$$\min\Big[m(f(\boldsymbol{C}_1)+f(\boldsymbol{C}_2))-\sum_{i=1}^{k}\{\hat{f}_i(\boldsymbol{C}_1;\boldsymbol{C}_2)+ \hat{f}_i(\boldsymbol{C}_2;\boldsymbol{C}_1)\}\Big]^2$$

where $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ denote two constructed solutions with all variables set to constants $c_1$ and $c_2$, respectively. The term $\hat{f}_i(\boldsymbol{C}_1;\boldsymbol{C}_2)$ represents the evaluation of $f$ where the variables in the $i$-th component are assigned values from $\boldsymbol{C}_1$, and the remaining variables are taken from $\boldsymbol{C}_2$. Similarly, $\hat{f}_i(\boldsymbol{C}_2;\boldsymbol{C}_1)$ evaluates $f$ by swapping the roles of $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$. This approach provides an estimate of variable interdependence, guiding the decomposition toward minimizing inter-component interactions.

To implement adaptive decomposition, Ref. [147] introduced frequency and group frequency matrices from evaluations at complementary solutions, then partitioned variables into fixed-size subgroups optimized *via* differential evolution (DE), improving feasibility and runtime. This idea was extended in Ref. [148], where a dependency metric was embedded in a CC framework with self-adaptive DE, enhancing robustness and reducing redundant evaluations. To eliminate manual subgroup settings, a population algorithm based on dynamic variable interaction identification for constrained problems (DVIIC)[149] encoded permutations and boundaries into integer genomes, dynamically adapting group structures with fewer evaluations. A refinement[150] replaced greedy search with simulated annealing and perturbed a single grouping vector, improving decomposition quality and reducing cost.

A grouping genetic algorithm (GGA)[151] was proposed to directly encode variable groups and apply group-level crossover and mutation, optimizing a dependency-based discrepancy function. It outperformed integer-encoded methods on 18 constrained problems. Further work[152] confirmed that group-based encoding yields better decompositions, especially for non-separable functions. Building on this, an evolutionary framework[153] combined interaction-driven partitioning with contribution-based prioritization and constraint consensus to guide resource allocation and repair infeasible solutions, showing strong performance on overlapping constraints. Refinement came *via* recursive differential grouping and feasibility distance-far (FDfar) consensus[154], applying selective repairs and evolving impactful subcomponents. Most recently, a maximum direction-based (DBmax) consensus was integrated into a cooperative DE framework[155], adaptively steering subproblem evolution and achieving faster feasibility and better solution quality.

A key trend is the integration of classical decomposition methods with computational intelligence to form hybrid algorithms that combine global exploration and local refinement. Typically, a metaheuristic explores the global space, while a classical method like linear programming solves subproblems precisely. This memetic structure balances diversity and accuracy. Machine learning also aids by automating strategy selection based on problem features, while embedded decomposition ensures feasibility. Such integration leverages adaptivity and mathematical rigor, often outperforming either approach used alone.

Computational intelligence-based decomposition provides an effective framework for large-scale optimization by dividing problems into subcomponents and iteratively refining solutions. Techniques such as cooperative coevolution scale well through variable partitioning, while recent advances like dynamic grouping and adaptive coordination improve performance on complex non-separable problems. The integration of machine learning adds adaptability to problem partitioning and parameter tuning. These methods provide scalability and flexibility beyond classical approaches, making them increasingly used in hybrid strategies that combine the strengths of both paradigms.

## 4 Summary and Conclusions

In this paper, we have reviewed the decomposition methods from a mathematics point of view and a computational intelligence point of view. While they each have their own strengths and weaknesses, their integration is expected to deliver greater outcomes. The mathematical decomposition methods often exploit the decomposition techniques on the special structures of the models whose optimality principles or convergent criteria of the proposed algorithms and schemes may be theoretically verifiable, while the computational intelligence techniques focus on how to decompose or group variables and functions of very general optimization problems, which may lack geometric and functional properties such as convexity, smoothness, continuity of the problem data. Integration and interplay of both methodologies are expected to produce time-critical, just enough feasible solutions for large-scale optimization problems.

### 4.1 Advantages of Decomposition Techniques

Computational intelligence based decomposition methods have been shown to offer several notable advantages. First, by partitioning high-dimensional decision variables into smaller and more manageable subproblems, these techniques achieve scalability that traditional solvers alone cannot match. Second, adaptive grouping strategies such as differential grouping and cooperative coevolution dynamically identify and exploit variable interactions, improving convergence by focusing computational effort where dependencies are strongest. Third, surrogate-assisted decompositions reduce expensive function evaluations through models like Gaussian processes and radial basis functions, maintaining solution quality while dramatically cutting evaluation cost. Finally, many mathematical decomposition frameworks such as Benders and Dantzig-Wolfe decompositions provide rigorous convergence guarantees under convexity assumptions, offering both theoretical soundness and practical reliability. Because these methods work directly with the analytical form of the problem, they can systematically reduce complexity while maintaining deterministic results. Their reliability and efficiency have led to their widespread integration into mathematical programming solvers, making them a preferred choice when the optimization model is convex, well-structured, and fully specified.

### 4.2 Disadvantages of Decomposition Techniques

Despite these benefits, several limitations persist. Decomposition quality often hinges on user specified parameters or prior knowledge of variable interactions; incorrect grouping can severely hamper overall performance. Moreover, the overhead of interaction tests, surrogate model training and context vector management can offset gains, especially in extremely large or time critical applications. Classical mathematical decompositions typically assume convexity or separability, limiting their applicability to mixed integer or highly non-convex problems without significant heuristic augmentation. Finally, effectively parallelizing subproblem solves demands sophisticated communication, synchronization and fault tolerance mechanisms, increasing implementation complexity and resource requirements.

### 4.3 Future Research Directions

By embracing hybridization, automation and scalability, decomposition techniques will remain central to tackling the next generation of large-scale optimization. Looking ahead, several promising avenues have emerged for advancing decomposition methods in large-scale optimization. Research into more autonomous and robust decomposition frameworks holds great promise. Integrating machine learning techniques for online identification of variable interactions and hyperparameter tuning could reduce manual intervention and adapt strategies to problem characteristics in real time. Extending robust and stochastic optimization concepts into decomposition schemes by combining trust region methods with Benders cuts may improve performance under uncertainty. Developing convergent decomposition algorithms for mixed integer and non-convex domains, perhaps *via* hybrid branch and cut or logic based methods, would broaden applicability. Therefore, leveraging high-performance computing and microservices architectures to orchestrate large-scale parallel decompositions could enable transparent scaling to thousands of cores, unlocking truly massive problem solving capabilities.

One of the most active areas involves the integration of decomposition techniques with machine learning. Recent developments in scientific machine learning suggest that domain decomposition can benefit significantly from data-driven approaches, particularly in enhancing convergence and generalization performance in high-dimensional problems[159]. These hybrid methods combine algorithmic rigor with the adaptability of learning models, enabling more intelligent problem partitioning and parameter selection.

Another important direction lies in the extension of decomposition frameworks to robust and stochastic optimization settings. Stabilized Benders decomposition methods, incorporating trust-region or bundle techniques, have shown strong performance under uncertainty. For instance, in energy planning problems characterized by deep uncertainty, these methods offer both parallelizability and scalability[160]. Such enhancements are critical for real-world applications where data variability cannot be ignored.

The challenges of non-convexity and combinatorial structures also call for attention. In particular, recent research has introduced Benders-based branch-and-cut algorithms tailored for mixed-integer and non-convex two-stage stochastic programs[161]. These methods provide theoretical convergence guarantees even in settings with binary or general integer variables, extending the scope of decomposition far beyond convex domains.

Parallel and distributed implementations of decomposition methods continue to be a key research frontier. Applications in process systems engineering have demonstrated the ability to scale decomposition algorithms across high-performance computing infrastructures. For instance, the use of progressive hedging in distributed settings has enabled the solution of extremely large design spaces involving tens of thousands of design variants[162]. Such progress is instrumental in pushing the boundaries of practical optimization.

Finally, an emerging area with significant potential is differentiable multilevel optimization. Recent work has developed gradient-based decomposition methods that efficiently solve nested optimization problems, such as those encountered in bilevel machine learning[163], continual learning, and hyperparameter tuning. These methods offer the dual benefits of mathematical rigor and end-to-end differentiability, making them attractive for integration with modern learning systems.

### 4.4 Recommendations for Practitioners

The future of decomposition methods lies in their integration with intelligent systems, their extension to uncertain and non-convex settings, and their adaptation to parallel computing and application-specific requirements. To conclude the review, we distill our findings into a few practical recommendations for practitioners applying decomposition methods to large-scale optimization as follows:

1) Parameter tuning: Leverage self-adaptive and automated tuning mechanisms, as decomposition algorithm performance is often sensitive to parameters such as penalty weights, population sizes, and convergence thresholds. Using self-tuning algorithms or systematic hyperparameter optimization reduces trial-and-error and improves the robustness of the underlying method. Recent studies show that adaptive parameter control during execution often outperforms static manual tuning across diverse problem instances.

2) Stopping criteria: Define clear and problem-appropriate stopping conditions to balance solution quality and computational cost. Criteria may include a maximum number of iterations, time limits, or convergence thresholds based on improvement between iterations. In heuristic methods, termination can depend on the lack of progress across multiple runs. Monitoring indicators such as duality gaps or objective value stabilization helps identify best possible solutions when further computation offers limited benefits.

3) Method selection: Choose decomposition methods based on problem structure and data availability. Techniques like Dantzig-Wolfe are suitable for problems with block structures and coupling constraints, while Benders decomposition is effective for complicated variables or stochastic elements. For large-scale or black-box problems, evolutionary or swarm-based approaches may be more suitable. When historical data or instance features are available, machine learning can assist in selecting the appropriate method. Graph-based analysis has been used to predict whether decomposition techniques or monolithic solvers perform better. Matching the method to problem properties such as linearity, separability, uncertainty, and computational resources is essential for effective strategy selection.

## References


[1] R K MARTIN. *Large Scale Linear and Integer Optimization: A Unified Approach* [M]. Boston: Kluwer Academic Publishers, 1999.

[2] V TSURKOV. *Large-Scale Optimization-Problems and Methods* [M]. Dordrecht: Kluwer Academic Publishers, 2001.

[3] F FURINI, I LJUBIĆ, E TRAVERSI. Preface: Decomposition Methods for Hard Optimization Problems [J]. *Annals of Operations Research*, 2020, 284(2): 483-485.

[4] G B DANTZIG, P WOLFE. Decomposition Principle for Linear Programs [J]. *Operations Research*, 1960, 8(1): 101-111.

[5] M E LÜBBECKE, J DESROSIERS. Selected Topics in Column Generation [J]. *Operations Research*, 2005, 53(6): 1007-1023.

[6] J GONDZIO, P GONZÁLEZ-BREVIS. A New Warmstarting Strategy for the Primal-Dual Column Generation Method [J]. *Mathematical Programming*, 2015, 152(1): 113-146.

[7] C A GAMBOA, D M VALLADÃO, A STREET, *et al*. Decomposition Methods for Wasserstein-Based Data-Driven Distributionally Robust Problems [J]. *Operations Research Letters*, 2021, 49(5): 696-702.

[8] F GRENOUILLEAU, N LAHRICHI, L M ROUSSEAU. New Decomposition Methods for Home Care Scheduling with Predefined Visits [J]. *Computers & Operations Research*, 2020, 115: 104855.

[9] A A S LEAO, M M FURLAN, F M B TOLEDO. Decomposition Methods for the Lot-Sizing and Cutting-Stock Problems in Paper Industries [J]. *Applied Mathematical Modelling*, 2017, 48: 250-268.

[10] A FRANGIONI, B GENDRON. A Stabilized Structured Dantzig-Wolfe Decomposition Method [J]. *Mathematical Programming*, 2013, 140(1): 45-76.

[11] C SAGASTIZÁBAL. Divide to Conquer: Decomposition Methods for Energy Optimization [J]. *Mathematical Programming*, 2012, 134(1): 187-222.

[12] S KUNNUMKAL, H TOPALOGLU. Linear Programming Based Decomposition Methods for Inventory Distribution Systems [J]. *European Journal of Operational Research*, 2011, 211(2): 282-297.

[13] Z ZHANG, X GONG, X L SONG, *et al*. A Column Generation-Based Exact Solution Method for Seru Scheduling Problems [J]. *Omega*, 2022, 108: 102581.

[14] M EL TONBARI, S AHMED. Consensus-Based Dantzig-Wolfe Decomposition [J]. *European Journal of Operational Research*, 2023, 307(3): 1441-1456.

[15] D YAZDANI, M N OMIDVAR, J BRANKE, *et al*.

Scaling up Dynamic Optimization Problems: A Divide-and-Conquer Approach [J]. *IEEE Transactions on Evolutionary Computation*, 2020, 24(1): 1-15.
[16] A FLORES-QUIROZ, J M PINTO, Q ZHANG. A Column Generation Approach to Multiscale Capacity Planning for Power-Intensive Process Networks [J]. *Optimization and Engineering*, 2019, 20(4): 1001-1027.
[17] B MAHVASH, A AWASTHI, S CHAUHAN. A Column Generation-Based Heuristic for the Three-Dimensional Bin Packing Problem with Rotation [J]. *Journal of the Operational Research Society*, 2018, 69(1): 78-90.
[18] M S FARHAM, H SÜRAL, C IYIGUN. A Column Generation Approach for the Location-Routing Problem with Time Windows [J]. *Computers & Operations Research*, 2018, 90: 249-263.
[19] J RIERA-LEDESMA, J J SALAZAR-GONZÁLEZ. Solving the Team Orienteering Arc Routing Problem with a Column Generation Approach [J]. *European Journal of Operational Research*, 2017, 262(1): 14-27.
[20] J F BENDERS. Partitioning Procedures for Solving Mixed-Variables Programming Problems [J]. *Numerische Mathematik*, 1962, 4(1): 238-252.
[21] A M GEOFFRION. Generalized Benders Decomposition [J]. J*ournal of Optimization Theory and Applications*, 1972, 10(4): 237-260.
[22] A GLEIXNER, S J MAHER, B MÜLLER, *et al*. Price-and-Verify: A New Algorithm for Recursive Circle Packing Using Dantzig-Wolfe Decomposition [J]. *Annals of Operations Research*, 2020, 284(2): 527-555.
[23] R G VICKSON, E HASSINI, N AZAD. A Benders Decomposition Approach to Product Location in Carousel Storage Systems [J]. *Annals of Operations Research*, 2020, 284(2): 623-643.
[24] J C LARIA, M C AGUILERA-MORILLO, R E LILLO. Group Linear Algorithm with Sparse Principal Decomposition: A Variable Selection and Clustering Method for Generalized Linear Models [J]. *Statistical Papers*, 2023, 64(1): 227-253.
[25] A PERRYKKAD, A T ERNST, M KRISHNAMOORTHY. A Simultaneous Magnanti-Wong Method to Accelerate Benders Decomposition for the Metropolitan Container Transportation Problem [J]. *Operations Research*, 2022, 70(3): 1531-1559.
[26] T L MAGNANTI, R T WONG. Accelerating Benders Decomposition: Algorithmic Enhancement and Model Selection Criteria [J]. *Operations Research*, 1981, 29(3): 464-484.
[27] N OLSEN, N KLIEWER, L WOLBECK. A Study on Flow Decomposition Methods for Scheduling of Electric Buses in Public Transport Based on Aggregated Time-Space Network Models [J]. *Central European Journal of Operations Research*, 2022, 30(3): 883-919.
[28] J GERBAUX, G DESAULNIERS, Q CAPPART. A Machine-Learning-Based Column Generation Heuristic for Electric Bus Scheduling [J]. *Computers & Operations Research*, 2025, 173: 106848.
[29] S A VÁSQUEZ, G ANGULO, M A KLAPP. An Exact Solution Method for the TSP with Drone Based on Decomposition [J]. *Computers & Operations Research*, 2021, 127: 105127.
[30] M E BRUNI, S KHODAPARASTI, M MOSHREF-JAVADI. A Logic-Based Benders Decomposition Method for the Multi-trip Traveling Repairman Problem with Drones [J]. *Computers & Operations Research*, 2022, 145: 105845.
[31] Z ZHANG, X L SONG, H HUANG, *et al*. Logic-Based Benders Decomposition Method for the Seru Scheduling Problem with Sequence-Dependent Setup Time and DeJong's Learning Effect [J]. *European Journal of Operational Research*, 2022, 297(3): 866-877.
[32] A S MICHELS, T C LOPES, L MAGATÃO. An Exact Method with Decomposition Techniques and Combinatorial Benders' Cuts for the Type-2 Multi-manned Assembly Line Balancing Problem [J]. *Operations Research Perspectives*, 2020, 7: 100163.
[33] D SHELAR, S AMIN, I A HISKENS. Evaluating Resilience of Electricity Distribution Networks *via* a Modification of Generalized Benders Decomposition Method [J]. *IEEE Transactions on Control of Network Systems*, 2021, 8(3): 1225-1238.
[34] H MOKHTAR, M KRISHNAMOORTHY, A T ERNST. The 2-Allocation P-Hub Median Problem and a Modified Benders Decomposition Method for Solving Hub Location Problems [J]. *Computers & Operations Research*, 2019, 104: 375-393.
[35] R SINDHWANI, J JAYARAM, D IVANOV. Meeting Economic and Social Viability Goals in Regional Airline Schemes Through Hub-and-Spoke Network Connectivity [J]. *Annals of Operations Research*, 2024: 1-55.
[36] A WITTHAYAPRAPHAKORN, P CHARNSETHIKUL. Benders Decomposition with Special Purpose Method for the Sub Problem in Lot Sizing Problem Under Uncertain Demand [J]. *Operations Research Perspectives*, 2019, 6: 100096.
[37] K KIM, V M ZAVALA. Algorithmic Innovations and Software for the Dual Decomposition Method Applied to Stochastic Mixed-Integer Programs [J]. *Mathematical Programming Computation*, 2018, 10(2): 225-266.
[38] V GUIGUES. Convergence Analysis of Sampling-Based Decomposition Methods for Risk-Averse Multistage Stochastic Convex Programs [J]. *SIAM Journal on Optimization*, 2016, 26(4): 2468-2494.
[39] A FAKHRI, M GHATEE. Application of Benders Decomposition Method in Solution of a Fixed-Charge Multicommodity Network Design Problem Avoiding Congestion [J]. *Applied Mathematical Modelling*, 2016, 40(13-14): 6468-6476.
[40] M SCHMIDT, M SIRVENT, W WOLLNER. A Decomposition Method for MINLPs with Lipschitz Continuous Nonlinearities [J]. *Mathematical Programming*, 2019, 178(1): 449-483.
[41] R VUJANIC, P MOHAJERIN ESFAHANI, P J GOULART, *et al*. A Decomposition Method for Large Scale MILPs, with Performance Guarantees and a Power System Application [J]. *Automatica*, 2016, 67: 144-156.
[42] A BECK, L TETRUASHVILI, Y VAISBOURD, *et al*. Rate of Convergence Analysis of Dual-Based Variables Decomposition Methods for Strongly Convex Problems [J]. *Operations Research Letters*, 2016, 44(1): 61-66.

[43] B YıLDıZ, N BOLAND, M SAVELSBERGH. Decomposition Branching for Mixed Integer Programming [J]. *Operations Research*, 2022, 70(3): 1854-1872.
[44] S BOYD, L XIAO, A MUTAPCIC, *et al*. Notes on Decomposition Methods [EB/OL]. (2008-04-13)[2025-08-01]. https://see.stanford.edu/materials/lsocoee364b/08-decomposition_notes.pdf.
[45] F GARCÍA-MUÑOZ, S DÁVILA, F QUEZADA. A Benders Decomposition Approach for Solving a Two-Stage Local Energy Market Problem Under Uncertainty [J]. *Applied Energy*, 2023, 329: 120226.
[46] R RAHMANIANI, S AHMED, T G CRAINIC, *et al*. The Benders Dual Decomposition Method [J]. *Operations Research*, 2020, 68(3): 878-895.
[47] M E LÜBBECKE, S J MAHER, J T WITT. Avoiding Redundant Columns by Adding Classical Benders Cuts to Column Generation Subproblems [J]. *Discrete Optimization*, 2021, 39: 100626.
[48] T W M VOSSEN, D ZHANG. Reductions of Approximate Linear Programs for Network Revenue Management [J]. *Operations Research*, 2015, 63(6): 1352-1371.
[49] A CRISTOFARI. A Decomposition Method for Lasso Problems with Zero-Sum Constraint [J]. *European Journal of Operational Research*, 2023, 306(1): 358-369.
[50] R RAHMANIANI, T G CRAINIC, M GENDREAU, *et al*. Accelerating the Benders Decomposition Method: Application to Stochastic Network Design Problems [J]. *SIAM Journal on Optimization*, 2018, 28(1): 875-903.
[51] E OGBE, X LI. A Joint Decomposition Method for Global Optimization of Multiscenario Non-convex Mixed-Integer Nonlinear Programs [J]. *Journal of Global Optimization*, 2019, 75(3): 595-629.
[52] A BIELE, L MÖNCH. Decomposition Methods for Cost and Tardiness Reduction in Aircraft Manufacturing Flow Lines [J]. *Computers & Operations Research*, 2019, 103: 134-147.
[53] T T TRAN, A ARAUJO, J C BECK. Decomposition Methods for the Parallel Machine Scheduling Problem with Setups [J]. *INFORMS Journal on Computing*, 2016, 28(1): 83-95.
[54] Y YANG, G SCUTARI, D P PALOMAR, *et al*. A Parallel Decomposition Method for Non-convex Stochastic Multi-agent Optimization Problems [J]. *IEEE Transactions on Signal Processing*, 2016, 64(11): 2949-2964.
[55] A RUSZCZYŃSKI. On Convergence of an Augmented Lagrangian Decomposition Method for Sparse Convex Optimization [J]. *Mathematics of Operations Research*, 1995, 20(3): 634-656.
[56] C H ROSA, A RUSZCZYŃSKI. On Augmented Lagrangian Decomposition Methods for Multistage Stochastic Programs [J]. *Annals of Operations Research*, 1996, 64(1): 289-309.
[57] J KOKO. A Survey on Dual Decomposition Methods [J]. *SeMA Journal*, 2013, 62(1): 27-59.
[58] X Y LAM, D F SUN, K C TOH. Semi-proximal Augmented Lagrangian-Based Decomposition Methods for Primal Block-Angular Convex Composite Quadratic Conic Programming Problems [J]. *INFORMS Journal on Optimization*, 2021, 3(3): 254-277.
[59] Q T DINH, I NECOARA, C SAVORGNAN, *et al*. An Inexact Perturbed Path-Following Method for Lagrangian Decomposition in Large-Scale Separable Convex Optimization [J]. *SIAM Journal on Optimization*, 2013, 23(1): 95-125.
[60] X D BAI, J SUN, X J ZHENG. An Augmented Lagrangian Decomposition Method for Chance-Constrained Optimization Problems [J]. *INFORMS Journal on Computing*, 2020, 33(3): 1056-1069.
[61] M LI, X M YUAN. The Augmented Lagrangian Method with Full Jacobian Decomposition and Logarithmic-Quadratic Proximal Regularization for Multiple-Block Separable Convex Programming [J]. *The SMAI Journal of Computational Mathematics*, 2018, 4: 81-120.
[62] B S HE, L S HOU, X M YUAN. On Full Jacobian Decomposition of the Augmented Lagrangian Method for Separable Convex Programming [J]. *SIAM Journal on Optimization*, 2015, 25(4): 2274-2312.
[63] R TAPPENDEN, P RICHTÁRIK, B BÜKE. Separable Approximations and Decomposition Methods for the Augmented Lagrangian [J]. *Optimization Methods and Software*, 2015, 30(3): 643-668.
[64] Y Y HUANG, S Y LIU. Proximal-Based Pre-correction Decomposition Methods for Structured Convex Minimization Problems [J]. *Journal of the Operations Research Society of China*, 2014, 2(2): 223-235.
[65] G CHENG, M TEBOULLE. A Proximal-Based Decomposition Method for Convex Minimization Problems [J]. *Mathematical Programming*, 1994, 64(1-3): 81-101.
[66] B S HE. Parallel Splitting Augmented Lagrangian Methods for Monotone Structured Variational Inequalities [J]. *Computational Optimization and Applications*, 2009, 42(2): 195-212.
[67] R SHEFI, M TEBOULLE. Rate of Convergence Analysis of Decomposition Methods Based on the Proximal Method of Multipliers for Convex Minimization [J]. *SIAM Journal on Optimization*, 2014, 24(1): 269-297.
[68] R T ROCKAFELLAR. Augmented Lagrangians and Applications of the Proximal Point Algorithm in Convex Programming [J]. *Mathematics of Operations Research*, 1976, 1(2): 97-116.
[69] Z S LU, Y ZHANG. Sparse Approximation *via* Penalty Decomposition Methods [J]. *SIAM Journal on Optimization*, 2013, 23(4): 2448-2478.
[70] C KANZOW, M LAPUCCI. Inexact Penalty Decomposition Methods for Optimization Problems with Geometric Constraints [J]. *Computational Optimization and Applications*, 2023, 85(3): 937-971.
[71] P M KLENIATI, P PARPAS, B RUSTEM. Decomposition-Based Method for Sparse Semidefinite Relaxations of Polynomial Optimization Problems [J]. *Journal of Optimization Theory and Applications*, 2010, 145(2): 289-310.
[72] M LAPUCCI, T LEVATO, M SCIANDRONE. Convergent Inexact Penalty Decomposition Methods for Cardinality-Constrained Problems [J]. *Journal of Optimization Theory and Applications*, 2021, 188(2): 473-496.
[73] Q J SHI, M Y HONG, X FU, *et al*. Penalty Dual Decomposition Method for Nonsmooth Non-convex Optimi-

zation: Part II: Applications [J]. *IEEE Transactions on Signal Processing*, 2020, 68: 4242-4257.
[74] X X HU, W M ZHU, B AN, *et al*. A Branch and Price Algorithm for EOS Constellation Imaging and Downloading Integrated Scheduling Problem [J]. *Computers & Operations Research*, 2019, 104: 74-89.
[75] Q HU, W B ZHU, H QIN, *et al*. A Branch-and-Price Algorithm for the Two-Dimensional Vector Packing Problem with Piecewise Linear Cost Function [J]. *European Journal of Operational Research*, 2017, 260(1): 70-80.
[76] T WANG, R BALDACCI, A LIM, *et al*. A Branch-and-Price Algorithm for Scheduling of Deteriorating Jobs and Flexible Periodic Maintenance on a Single Machine [J]. *European Journal of Operational Research*, 2018, 271(3): 826-838.
[77] A GHARAEI, F JOLAI. A Branch and Price Approach to the Two-Agent Integrated Production and Distribution Scheduling [J]. *Computers & Industrial Engineering*, 2019, 136: 504-515.
[78] G RAUCHECKER, G SCHRYEN. An Exact Branch-and-Price Algorithm for Scheduling Rescue Units During Disaster Response [J]. *European Journal of Operational Research*, 2019, 272(1): 352-363.
[79] A AHMADI-JAVID, E AMIRI, M MESKAR. A Profit-Maximization Location-Routing-Pricing Problem: A Branch-and-Price Algorithm [J]. *European Journal of Operational Research*, 2018, 271(3): 866-881.
[80] Y ZHENG, G FANTUZZI, A PAPACHRISTODOULOU, *et al*. Chordal Decomposition in Operator-Splitting Methods for Sparse Semidefinite Programs [J]. *Mathematical Programming*, 2020, 180(1): 489-532.
[81] J S PANG, M TAO. Decomposition Methods for Computing Directional Stationary Solutions of a Class of Nonsmooth Non-convex Optimization Problems [J]. *SIAM Journal on Optimization*, 2018, 28(2), 1640−1669.
[82] P MAHEY, J KOKO, A LENOIR. Decomposition Methods for a Spatial Model for Long-Term Energy Pricing Problem [J]. *Mathematical Methods of Operations Research*, 2017, 85(1): 137-153.
[83] A ALVARADO, G SCUTARI, J S PANG. A New Decomposition Method for Multiuser DC-Programming and Its Applications [J]. *IEEE Transactions on Signal Processing*, 2014, 62(11): 2984-2998.
[84] J YU, M Q LI, Y L WANG, *et al*. A Decomposition Method for Large-Scale Box Constrained Optimization [J]. *Applied Mathematics and Computation*, 2014, 231: 9-15.
[85] M DELL'AMICO, M DELORME, M IORI, *et al*. Mathematical Models and Decomposition Methods for the Multiple Knapsack Problem [J]. *European Journal of Operational Research*, 2019, 274(3): 886-899.
[86] A MANNO, L PALAGI, S SAGRATELLA. Parallel Decomposition Methods for Linearly Constrained Problems Subject to Simple Bound with Application to the SVMS Training [J]. *Computational Optimization and Applications*, 2018, 71(1): 115-145.
[87] N AYDIN, S I BIRBIL. Decomposition Methods for Dynamic Room Allocation in Hotel Revenue Management [J]. *European Journal of Operational Research*, 2018, 271(1): 179-192.
[88] E OGBE, X LI. A New Cross Decomposition Method for Stochastic Mixed-Integer Linear Programming [J]. *European Journal of Operational Research*, 2017, 256(2): 487-499.
[89] P GIRARDEAU, V LECLERE, A B PHILPOTT. On the Convergence of Decomposition Methods for Multistage Stochastic Convex Programs [J]. *Mathematics of Operations Research*, 2014, 40(1): 130-145.
[90] D HENRION, J MALICK. Projection Methods for Conic Feasibility Problems: Applications to Polynomial Sum-of-Squares Decompositions [J]. *Optimization Methods and Software*, 2011, 26(1): 23-46.
[91] H GANGAMMANAVAR, M BANSAL. Stochastic Decomposition Method for Two-Stage Distributionally Robust Linear Optimization [J]. *SIAM Journal on Optimization*, 2022, 32(3): 1901-1930.
[92] H GANGAMMANAVAR, Y F LIU, S SEN. Stochastic Decomposition for Two-Stage Stochastic Linear Programs with Random Cost Coefficients [J]. *INFORMS Journal on Computing*, 2020, 33(1): 51-71.
[93] A ALLMAN, W T TANG, P DAOUTIDIS. DeCODe: A Community-Based Algorithm for Generating High-Quality Decompositions of Optimization Problems [J]. *Optimization and Engineering*, 2019, 20(4): 1067-1084.
[94] L LI, L C JIAO, R STOLKIN, *et al*. Mixed Second Order Partial Derivatives Decomposition Method for Large Scale Optimization [J]. *Applied Soft Computing*, 2017, 61: 1013-1021.
[95] R MORABITO, M C DE SOUZA, M VAZQUEZ. Approximate Decomposition Methods for the Analysis of Multicommodity Flow Routing in Generalized Queuing Networks [J]. *European Journal of Operational Research*, 2014, 232(3): 618-629.
[96] H Q SONG, R K CHEUNG, H Y WANG. An Arc-Exchange Decomposition Method for Multistage Dynamic Networks with Random Arc Capacities [J]. *European Journal of Operational Research*, 2014, 233(3): 474-487.
[97] M WIRTZ, M HELENO, A MOREIRA, *et al*. 5th Generation District Heating and Cooling Network Planning: A Dantzig-Wolfe Decomposition Approach [J]. *Energy Conversion and Management*, 2023, 276: 116593.
[98] H KARATEKE, R ŞAHIN, S NIROOMAND. A Hybrid Dantzig-Wolfe Decomposition Algorithm for the Multi-floor Facility Layout Problem [J]. *Expert Systems with Applications*, 2022, 206: 117845.
[99] J J DING, S Q CHANG, R F WANG, *et al*. Parallel DEA-Dantzig-Wolfe Algorithm for Massive Data Applications [J]. *Computers & Industrial Engineering*, 2023, 175: 108875.
[100] G BASTIANEL, H ERGUN, D VAN HERTEM. Linearised Optimal Power Flow Problem Solution Using Dantzig-Wolfe Decomposition [C]//*2022 IEEE PES Innovative Smart Grid Technologies Conference Europe* (*ISGT-Europe*). Novi Sad, Serbia: IEEE, 2022: 1-5.
[101] A S ZAMZAM, E DALL'ANESE, N D SIDIROPOULOS. Optimal Distributed Energy Storage Management Using Relaxed Dantzig-Wolfe Decomposition [C]//*2018 IEEE Conference on Decision and Control* (*CDC*). Miami, FL, USA: IEEE, 2018: 2396-2401.

[102] T SCHÜTZ, X L HU, M FUCHS, *et al*. Optimal Design of Decentralized Energy Conversion Systems for Smart Microgrids Using Decomposition Methods [J]. *Energy*, 2018, 156: 250-263.
[103] Y YAO, X N ZHU, H Y DONG, *et al*. ADMM-Based Problem Decomposition Scheme for Vehicle Routing Problem with Time Windows [J]. *Transportation Research Part B: Methodological*, 2019, 129: 156-174.
[104] M T ADHAM, P J BENTLEY, D DIAZ. Evaluating Decomposition Strategies to Enable Scalable Scheduling for a Real-World Multi-line Steel Scheduling Problem [C]//*2017 IEEE Symposium Series on Computational Intelligence* (*SSCI*). Honolulu, HI, USA: IEEE, 2017: 1-8.
[105] M H AMINI, P MCNAMARA, P WENG, *et al*. Hierarchical Electric Vehicle Charging Aggregator Strategy Using Dantzig-Wolfe Decomposition [J]. *IEEE Design & Test*, 2018, 35(6): 25-36.
[106] V PUNNATHANAM, Y SHASTRI. Optimization of a Large-Scale Biorefinery Problem by Decomposition [C]//*29th European Symposium on Computer Aided Process Engineering*. Amsterdam: Elsevier, 2019: 829-834.
[107] J CHOI, C LEE, S PARK. Dantzig-Wolfe Decomposition Approach to the Vehicle Assignment Problem with Demand Uncertainty in a Hybrid Hub-and-Spoke Network [J]. *Annals of Operations Research*, 2018, 264(1): 57-87.
[108] Y WANG, G S ZHANG, L ZHANG, *et al*. A Column-Generation Based Approach for Integrating Surgeon and Surgery Scheduling [J]. *IEEE Access*, 2018, 6: 41578-41589.
[109] S RIAZI, O WIGSTRÖM, K BENGTSSON, *et al*. Decomposition and Distributed Algorithms for Home Healthcare Routing and Scheduling Problem [C]//*2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation* (*ETFA*). Limassol, Cyprus: IEEE, 2017: 1-7.
[110] M A POTTER, K A JONG. A Cooperative Coevolutionary Approach to Function Optimization [C]//*International Conference on Parallel Problem Solving from Nature*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1994: 249-257.
[111] M MESELHI, R SARKER, D ESSAM, *et al*. A Decomposition Approach for Large-Scale Non-separable Optimization Problems [J]. *Applied Soft Computing*, 2022, 115: 108168.
[112] Z Y YANG, K TANG, X YAO. Large Scale Evolutionary Optimization Using Cooperative Coevolution [J]. *Information Sciences*, 2008, 178(15): 2985-2999.
[113] X D LI, X YAO. Tackling High Dimensional Nonseparable Optimization Problems by Cooperatively Coevolving Particle Swarms [C]//*2009 IEEE Congress on Evolutionary Computation*. Trondheim, Norway: IEEE, 2009: 1546-1553.
[114] P N WILLIAMS, K LI, G Y MIN. Large-Scale Evolutionary Optimization *via* Multi-task Random Grouping [C]//*2021 IEEE International Conference on Systems, Man, and Cybernetics* (*SMC*). Melbourne, Australia: IEEE, 2021: 778-783.
[115] R C LIU, J LIU, Y F LI, *et al*. A Random Dynamic Grouping Based Weight Optimization Framework for Large-Scale Multi-objective Optimization Problems [J]. *Swarm and Evolutionary Computation*, 2020, 55: 100684.
[116] B X GUAN, T T XU, Y H ZHAO, *et al*. A Random Grouping-Based Self-Regulating Artificial Bee Colony Algorithm for Interactive Feature Detection [J]. *Knowledge-Based Systems*, 2022, 243: 108434.
[117] R S LATHA, B SARAVANA BALAJI, N BACANIN, *et al*. Feature Selection Using Grey Wolf Optimization with Random Differential Grouping [J]. *Computer Systems Science and Engineering*, 2022, 43(1): 317-332.
[118] M N OMIDVAR, X D LI, X YAO. Cooperative Co-evolution with Delta Grouping for Large Scale Non-separable Function Optimization [C]//*IEEE Congress on Evolutionary Computation*. Barcelona, Spain: IEEE, 2010: 1-8.
[119] M N OMIDVAR, X D LI, Y MEI, *et al*. Cooperative Co-evolution with Differential Grouping for Large Scale Optimization [J]. *IEEE Transactions on Evolutionary Computation*, 2014, 18(3): 378-393.
[120] Y B LING, H J LI, B CAO. Cooperative Co-evolution with Graph-Based Differential Grouping for Large Scale Global Optimization [C]//*2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery* (*ICNC-FSKD*). Changsha, China: IEEE, 2016: 95-102.
[121] Y MEI, M N OMIDVAR, X D LI, *et al*. A Competitive Divide-and-Conquer Algorithm for Unconstrained Large-Scale Black-Box Optimization [J]. *ACM Transactions on Mathematical Software*, 2016, 42(2): 1-24.
[122] Y SUN, M KIRLEY, S K HALGAMUGE. A Recursive Decomposition Method for Large Scale Continuous Optimization [J]. *IEEE Transactions on Evolutionary Computation*, 2018, 22(5): 647-661.
[123] Y SUN, M N OMIDVAR, M KIRLEY, *et al*. Adaptive Threshold Parameter Estimation with Recursive Differential Grouping for Problem Decomposition [C]//*Proceedings of the Genetic and Evolutionary Computation Conference*. Kyoto Japan: ACM, 2018: 889-896.
[124] Y SUN, X D LI, A ERNST, *et al*. Decomposition for Large-Scale Optimization Problems with Overlapping Components [C]//*2019 IEEE Congress on Evolutionary Computation* (*CEC*). Wellington, New Zealand: IEEE, 2019: 326-333.
[125] M YANG, A M ZHOU, C H LI, *et al*. An Efficient Recursive Differential Grouping for Large-Scale Continuous Problems [J]. *IEEE Transactions on Evolutionary Computation*, 2021, 25(1): 159-171.
[126] S RASHIDI, P RANJITKAR. Bus Dwell Time Modeling Using Gene Expression Programming [J]. *Computer-Aided Civil and Infrastructure Engineering*, 2015, 30(6): 478-489.
[127] S MAHDAVI, M E SHIRI, S RAHNAMAYAN. Cooperative Co-evolution with a New Decomposition Method for Large-Scale Optimization [C]//*2014 IEEE Congress on Evolutionary Computation* (*CEC*). Beijing, China: IEEE, 2014: 1285-1292.
[128] S Q SHAN, G G Wang. Metamodeling for High Dimensional Simulation-Based Design Problems [J]. *Journal of Mechanical Design*, 2010, 132(5): 051009.
[129] F A C VIANA, R T HAFTKA, L T WATSON. Efficient Global Optimization Algorithm Assisted by

Multiple Surrogate Techniques [J]. *Journal of Global Optimization*, 2013, 56(2): 669-689.

[130] C WANG, Q Y DUAN, W GONG, *et al*. An Evaluation of Adaptive Surrogate Modeling Based Optimization with Two Benchmark Problems [J]. *Environmental Modelling & Software*, 2014, 60: 167-179.

[131] H BASHAR. Meta-Modelling of Intensive Computational Models [D]. Sheffield: University of Sheffield, 2016.

[132] E Y LI, H WANG, F YE. Two-Level Multi-surrogate Assisted Optimization Method for High Dimensional Nonlinear Problems [J]. *Applied Soft Computing*, 2016, 46: 26-36.

[133] C L SUN, Y C JIN, R CHENG, *et al*. Surrogate-Assisted Cooperative Swarm Optimization of High-Dimensional Expensive Problems [J]. *IEEE Transactions on Evolutionary Computation*, 2017, 21(4): 644-660.

[134] G H LI, Q F ZHANG, Q Z LIN, *et al*. A Three-Level Radial Basis Function Method for Expensive Optimization [J]. *IEEE Transactions on Cybernetics*, 2022, 52(7): 5720-5731.

[135] L H CHRISTIANSEN, A ELTVED, S HARTMANN, *et al*. Fast, Robust and Global Optimisation for Antenna Design Using Meta Modelling [C]//*2025 19th European Conference on Antennas and Propagation* (*EuCAP*). Stockholm, Sweden: IEEE, 2025: 1-5.

[136] M MUNETOMO, D E GOLDBERG. Linkage Identification by Non-monotonicity Detection for Overlapping Functions [J]. *Evolutionary Computation*, 1999, 7(4): 377-398.

[137] W X CHEN, T WEISE, Z Y YANG, *et al*. Large-Scale Global Optimization Using Cooperative Coevolution with Variable Interaction Learning [C]//*International Conference on Parallel Problem Solving from Nature*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010: 300-309.

[138] L SUN, S YOSHIDA, X C CHENG, *et al*. A Cooperative Particle Swarm Optimizer with Statistical Variable Interdependence Learning [J]. *Information Sciences*, 2012, 186(1): 20-39.

[139] X CHEN, R A SERVEDIO, L Y TAN. New Algorithms and Lower Bounds for Monotonicity Testing [C]//*2014 IEEE 55th Annual Symposium on Foundations of Computer Science*. Philadelphia, PA, USA: IEEE, 2014: 286-295.

[140] H W GE, L SUN, X YANG, *et al*. Cooperative Differential Evolution with Fast Variable Interdependence Learning and Cross-Cluster Mutation [J]. *Applied Soft Computing*, 2015, 36: 300-314.

[141] Y SUN, X D LI, A ERNST, *et al*. Decomposition for Large-Scale Optimization Problems with Overlapping Components [C]//*2019 IEEE Congress on Evolutionary Computation* (*CEC*). Wellington, New Zealand: IEEE, 2019: 326-333.

[142] Z LIU. Computational Complexity Meets Statistical Efficiency: From Change Point Estimation to Monotonicity Testing [D]. Göttingen: Georg-August-Universität Göttingen, 2025.

[143] B MATTHIESEN, C HELLINGS, E A JORSWIECK, *et al*. Mixed Monotonic Programming for Fast Global Optimization [J]. *IEEE Transactions on Signal Processing*, 2020, 68: 2529-2544.

[144] L G CASADO, B G TÓTH, E M T HENDRIX, *et al*. Local Search Versus Linear Programming to Detect Monotonicity in Simplicial Branch and Bound [J]. *Journal of Global Optimization*, 2025, 91(2): 311-330.

[145] L GALLI, H RAUHUT, M SCHMIDT. Don't Be So Monotone: Relaxing Stochastic Line Search in Over-Parameterized Models [J]. *Advances in Neural Information Processing Systems*, 2023, 36, 34752−34764.

[146] E SAYED, D ESSAM, R SARKER. Dependency Identification Technique for Large Scale Optimization Problems [C]//*2012 IEEE Congress on Evolutionary Computation*. Brisbane, QLD, Australia: IEEE, 2012: 1-8.

[147] E SAYED, D ESSAM, R SARKER, *et al*. Decomposition-Based Evolutionary Algorithm for Large Scale Constrained Problems [J]. *Information Sciences*, 2015, 316: 457-486.

[148] G M DAI, X Y CHEN, L CHEN, *et al*. Cooperative Coevolution with Dependency Identification Grouping for Large Scale Global Optimization [C]//*2016 IEEE Congress on Evolutionary Computation* (*CEC*). Vancouver, BC, Canada: IEEE, 2016: 5201-5208.

[149] A E AGUILAR-JUSTO, E MEZURA-MONTES, S M ELSAYED, *et al*. Decomposition of Large-Scale Constrained Problems Using a Genetic-Based Search [C]//*2016 IEEE International Autumn Meeting on Power, Electronics and Computing* (*ROPEC*). Ixtapa, Mexico: IEEE, 2016: 1-6.

[150] A E AGUILAR-JUSTO, E MEZURA-MONTES. Towards an Improvement of Variable Interaction Identification for Large-Scale Constrained Problems [C]//*2016 IEEE Congress on Evolutionary Computation* (*CEC*). Vancouver, BC, Canada: IEEE, 2016: 4167-4174.

[151] G CARMONA-ARROYO, M QUIROZ-CASTELLANOS, E MEZURA-MONTES. Variable Decomposition for Large-Scale Constrained Optimization Problems Using a Grouping Genetic Algorithm [J]. *Mathematical and Computational Applications*, 2022, 27(2): 23.

[152] G CARMONA ARROYO. A Grouping Genetic Algorithm for Variable Decomposition in Large-Scale Constrained Optimization Problems [D]. Xalapa: Universidad Veracruzana, 2024.

[153] M MESELHI, N HAMZA, S ELSAYED, *et al*. An Evolutionary Framework for Large-Scale Constrained Optimization [C]//*2024 IEEE International Conference on Systems, Man, and Cybernetics* (*SMC*). Kuching, Malaysia: IEEE, 2024: 3751-3756.

[154] N HAMZA, R SARKER, D ESSAM, *et al*. Constraint Consensus for Solving Large-Scale Constrained Optimization Problems [C]//*2024 IEEE Congress on Evolutionary Computation* (*CEC*). Yokohama, Japan: IEEE, 2024: 1-8.

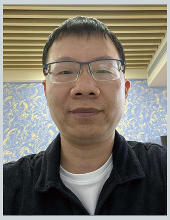
[155] N HAMZA, S ELSAYED, R SARKER, *et al*. Constraint Consensus Assisted Evolutionary Algorithm for Large-Scale Constrained Optimization [J]. *Applied Soft Computing*, 2025, 181: 113383.

[156] Y C JIN, B SENDHOFF. A Systems Approach to Evolutionary Multiobjective Structural Optimization and Beyond [J]. *IEEE Computational Intelligence Magazine*, 2009, 4(3): 62-76.

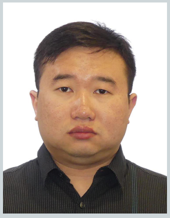
[157] Z G REN, B PANG, M Y WANG, *et al*. Surrogate

Model Assisted Cooperative Coevolution for Large Scale Optimization [J]. *Applied Intelligence*, 2019, 49(2): 513-531.
[158] I MITRAI, P DAOUTIDIS. Taking the Human out of Decomposition-Based Optimization *via* Artificial Intelligence, Part II: Learning to Initialize [J]. *Computers & Chemical Engineering*, 2024, 186: 108686.
[159] A KLAWONN, M LANSER, J WEBER. Machine Learning and Domain Decomposition Methods - A Survey [J]. *Computational Science and Engineering*, 2024, 1(1): 2.
[160] L GÖKE, F SCHMIDT, M KENDZIORSKI. Stabilized Benders Decomposition for Energy Planning Under Climate Uncertainty [J]. *European Journal of Operational Research*, 2024, 316(1): 183-199.
[161] A MAHÉO, S BELIERES, Y ADULYASAK, *et al*. Unified Branch-and-Benders-Cut for Two-Stage Stochastic Mixed-Integer Programs [J]. *Computers & Operations Research*, 2024, 164: 106526.
[162] G STINCHFIELD, J P WATSON, C D LAIRD. Progressive Hedging Decomposition for Solutions of Large-Scale Process Family Design Problems [C]// *34th European Symposium on Computer Aided Process Engineering / 15th International Symposium on Process Systems Engineering*. Amsterdam: Elsevier, 2024: 1285-1290.
[163] Y H ZHANG, P KHANDURI, I TSAKNAKIS, *et al*. An Introduction to Bilevel Optimization: Foundations and Applications in Signal Processing and Machine Learning [J]. *IEEE Signal Processing Magazine*, 2024, 41(1): 38-59.

## Author Introduction



**Thai Doan CHUONG** Department of Mathematics, Brunel University of London, London, United Kingdom. Thai Doan CHUONG received a Ph.D. degree in Mathematics from the Institute of Mathematics, Vietnam Academy of Science and Technology, Vietnam. He is currently a Lecturer in Mathematics and Statistics at Brunel University of London, United Kingdom. He has published more than 80 research articles in the areas of Optimization and Operations Research. He now serves as an Editor of *Journal of Nonlinear and Variational Analysis* and a Member of Editorial Board of *Applied Analysis and Optimization*. Email: chuong.thaidoan@brunel.ac.uk; chuongthaidoan@yahoo.com.



**Chen LIU** (Member, IEEE) School of Engineering, RMIT University, Melbourne, Australia. Chen LIU received the bachelor's degree (Hons.) in computer science, the master's degree in information technology and the Ph.D. degree in Electrical & Electronic Engineering from RMIT University, Melbourne, VIC, Australia, in 2015, 2016 and 2021 respectively. He is currently a research fellow at RMIT's School of Engineering. His current research interests include bilevel optimization, machine learning, and large language models. Email: chen.liu3@rmit.edu.au.

**Xinghuo YU** (Fellow, IEEE) received B.E. and M.E. degrees in electrical and electronic engineering from the University of Science and Technology of China, Hefei, China, in 1982 and 1984, and Ph.D. degree in control science and engineering from Southeast University, Nanjing, China in 1988, respectively. He is a Distinguished Professor and an Associate Deputy Vice-Chancellor at RMIT University, Melbourne, Australia. He was the President of IEEE Industrial Electronics Society in 2018 and 2019. His research interests include control systems, complex and intelligent systems, and power and energy systems. He received a number of awards and honors for his contributions, including 2013 Dr.-Ing. Eugene Mittelmann Achievement Award of IEEE Industrial Electronics Society and 2018 M. A. Sargent Medal from Engineers Australia. He is a Fellow of the Australian Academy of Science and an Honorary Fellow of the Institution of Engineers Australia. Email: xinghuo.yu@rmit.edu.au.