

Wei Wei, Tsinghua University

Tutorials on Advanced Optimization Methods

July 19, 2020

Springer

Tutorials on Advanced Optimization Methods

Wei Wei, Tsinghua University

This material is the appendix part of my book collaborated with Professor Jianhui Wang at Southern Methodist University:

Wei Wei, Jianhui Wang. *Modeling and Optimization of Interdependent Energy Infrastructures*. Springer Nature Switzerland, 2020.

<https://link.springer.com/book/10.1007%2F978-3-030-25958-7>

This material provides thorough tutorials on some optimization techniques frequently used in various engineering disciplines, including

- ♣ Convex optimization
- ♣ Linearization technique and mixed-integer linear programming
- ♣ Robust optimization
- ♣ Equilibrium/game problems

It discusses how to reformulate a difficult (non-convex, multi-agent, min-max) problem to a solver-compatible form (semidefinite program, mixed-integer linear program) via convexification, linearization, and decomposition, so the original problem can be reliably solved by commercial/open-source software. Fundamental algorithms (simplex algorithm, interior-point algorithm) are not the main focus.

This material is a good reference for self-learners who have basic knowledge in linear algebra and linear programming. It is one of the main references for an optimization course taught at Tsinghua University. If you need teaching slides, please contact wei-wei04@mails.tsinghua.edu.cn or find the up-to-date contact information at <https://sites.google.com/view/weiweipes/>

Contents

A	Basics of Linear and Conic Programs	7
A.1	Basic Notations	8
A.1.1	Convex Sets	8
A.1.2	Generalized Inequalities	13
A.1.3	Dual Cones and Dual Generalized Inequalities	14
A.1.4	Convex Function and Epigraph	16
A.2	From Linear to Conic Program	18
A.2.1	Linear Program and its Duality Theory	18
A.2.2	General Conic Linear Program	23
A.2.3	Second-order Cone Program	27
A.2.4	Semidefinite Program	34
A.3	Convex Relaxation Methods for Non-convex QCQPs	39
A.3.1	SDP Relaxation and Valid Inequalities	40
A.3.2	Successively Tightening the Relaxation	43
A.3.3	Completely Positive Program Relaxation	47
A.3.4	MILP Approximation	49
A.4	MILP Formulation of Nonconvex QPs	50
A.4.1	Nonconvex QPs over polyhedra	50
A.4.2	Standard Nonconvex QPs	53
A.5	Further Reading	55
	References	57
B	Formulation Recipes in Integer Programming	61
B.1	Piecewise Linear Approximation of Nonlinear Functions	62
B.1.1	Univariate Continuous Function	62
B.1.2	Bivariate Continuous Nonlinear Function	67
B.1.3	Approximation Error	70
B.2	Linear Formulation of Product Terms	72
B.2.1	Product of Two Binary Variables	72
B.2.2	Product of Integer and Continuous Variables	73
B.2.3	Product of Two Continuous Variables	74

B.2.4	Monomial of Binary Variables	76
B.2.5	Product of Functions in Integer Variables	77
B.2.6	Log-sum Functions	78
B.3	Other Frequently used Formulations	79
B.3.1	Minimum Values	79
B.3.2	Maximum Values	79
B.3.3	Absolute Values	80
B.3.4	Linear Fractional of Binary Variables	80
B.3.5	Disjunctive Inequalities	81
B.3.6	Logical Conditions	85
B.4	Further Reading	86
References	87
C	Basics of Robust Optimization	89
C.1	Static Robust Optimization	90
C.1.1	Basic Assumptions and Formulations	91
C.1.2	Tractable Reformulations	93
C.1.3	Formulation Issues	97
C.2	Adjustable Robust Optimization	104
C.2.1	Basic Assumptions and Formulations	104
C.2.2	Affine Policy Based Approximation Model	108
C.2.3	Algorithms for Fully Adjustable Models	110
C.3	Distributionally Robust Optimization	119
C.3.1	Static Distributionally Robust Optimization	120
C.3.2	Adjustable Distributionally Robust Optimization	129
C.4	Data-driven Robust Stochastic Program	133
C.4.1	Robust Chance Constrained Stochastic Program	134
C.4.2	Stochastic Program with Discrete Distributions	148
C.4.3	Formulations based on Wasserstein Metric	153
C.5	Further Reading	168
References	170
D	Equilibrium Problems	175
D.1	Standard Nash Equilibrium Problem	176
D.1.1	Formulation and Optimality Condition	176
D.1.2	Variational Inequality Formulation	177
D.1.3	Best Response Algorithms	180
D.1.4	Nash Equilibrium of Matrix Games	182
D.1.5	Potential Games	186
D.2	Generalized Nash Equilibrium Problem	189
D.2.1	Formulation and Optimality Condition	190
D.2.2	Best-Response Algorithm	195
D.3	Bilevel Programs	198
D.3.1	Bilevel Programs with a Convex Lower Level	198
D.3.2	Special Bilevel Programs	204

D.3.3	Bilevel Mixed-integer Program	212
D.4	Mathematical Programs with Equilibrium Constraints	217
D.4.1	Mathematical Formulation.....	217
D.4.2	Linear Complementarity Problem	218
D.4.3	Linear Programs with Complementarity Constraints ..	222
D.5	Equilibrium Programs with Equilibrium Constraints	222
D.5.1	Mathematical model	223
D.5.2	Methods for Solving an EPEC	223
D.6	Conclusions and Further Reading.....	229
References	230

Appendix A

Basics of Linear and Conic Programs

*The great watershed in optimization isn't between
linearity and nonlinearity, but convexity and
non-convexity.*

—Ralph Tyrrell Rockafellar

The mathematical programming theory has been thoroughly developed in width and depth since its birth in 1940s, when George Dantzig invented simplex algorithm for linear programming. The most influential findings in the field of optimization theory can be summarized as [1]:

1) Recognition of the fact that under mild conditions, a convex optimization program is computationally tractable: the computational effort under a given accuracy grows moderately with the problem size even in the worst case. In contrast, a non-convex program is generally computationally intractable: the computational effort of the best known methods grows prohibitively fast with respect to the problem size, and it is reasonable to believe that this is an intrinsic feature of such problems rather than a limitation of existing optimization techniques.

2) The discovery of interior-point methods, which was originally developed in 1980s to solve LPs and could be generalized to solve convex optimization problems as well. Moreover, between these two extremes (LPs and general convex programs), there are many important and useful convex programs. Although nonlinear, they still possess nice structured properties, which can be utilized to develop more dedicated algorithms. These polynomial-time interior-point algorithms turn out to be considerably more efficient than those exploiting only the convex property.

The superiority of formulating a problem as a convex optimization problem is apparent. The most appealing advantage is that the problem can be solved reliably and efficiently. It is also convenient to build the associated dual problem, which gives insights on sensitivity information and may help develop distributed algorithm for solving the problem. Convex optimization has been applied in a number of energy system operational issues, and well acknowledged for its computational superiority. We believe that it is imperative for researchers and engineers to develop certain understanding on this important topic.

As we have already learnt in previous chapters, many optimization problems in energy system engineering can be formulated as or converted to con-

vex programs. The goal of this chapter is to help readers develop necessary background knowledge and skills to apply several well-structured convex optimization models, including LPs, SOCPs, and SDPs, i.e., to formulate or transform their problems as these specific convex programs. Certainly, convex transformation (or convexification) may be rather tricky and require special knowledge and skills. Nevertheless, the attempt often turns out to be worthwhile. We also pay special attention to nonconvex QCQPs, which can model various decision-making problems in engineering, such as optimal power flow in Chap. ?? and optimal gas flow in Chap. ?. We discuss convex relaxation technique based on SDP, which is shown to be very useful to get a high-quality objective lower bound. We also present MILP formulations for some special QPs; because of the special problem structure, these MILP models can tackle practically sized problems in reasonable time.

Most materials regarding convex sets and functions come from [2] and its solution manual [3]; extensions of duality theory from linear programming to conic programming follows from [1]. We consolidate necessary contents in a convenient way to make this book self-contained and easy to follow.

A.1 Basic Notations

A.1.1 Convex Sets

A set $C \in \mathbb{R}^n$ is convex if the line segment connecting any two points in C is contained in C , i.e., for any $x_1, x_2 \in C$, we have $\theta x_1 + (1 - \theta)x_2 \in C$, $\forall \theta \in [0, 1]$. Roughly speaking, standing at anywhere in a convex set, you can see every other point in the set. Fig. A.1 illustrates a simple convex set and a non-convex set in \mathbb{R}^2 .

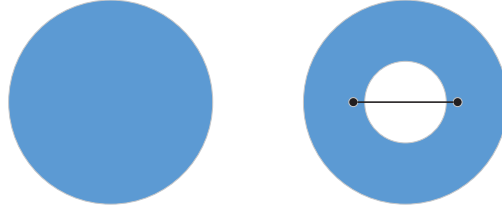


Fig. A.1 Left: the circle is convex; right: the ring is non-convex.

The convex combination of k points x_1, \dots, x_k is defined as $\theta_1 x_1 + \dots + \theta_k x_k$, where $\theta_1, \dots, \theta_k \geq 0$, and $\theta_1 + \dots + \theta_k = 1$. A convex combination of points can be regarded as a weighted average of the points, with θ_i the weight of x_i in the mixture.

The convex hull of set C , denoted $\text{conv}(C)$, is the smallest convex set that contains C . Particularly, if C has finite elements, then

$$\text{conv}(C) = \{\theta_1 x_1 + \cdots + \theta_k x_k \mid x_i \in C, \theta_i \geq 0, i = 1, \dots, k, \theta_1 + \cdots + \theta_k = 1\}$$

Fig. A.2 illustrates the convex hulls of two sets in \mathbb{R}^2 .

Some useful convex sets are briefly introduced.

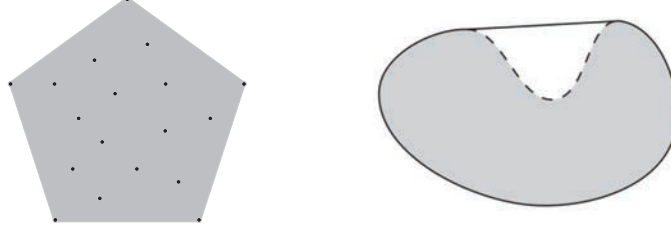


Fig. A.2 Left: The convex hull of eighteen points. Right: The convex hull of a kidney shaped set.

1. Cones

A set C is called a cone, or nonnegative homogeneous, if for any $x \in C$, we have $\theta x \in C, \forall \theta \geq 0$. A set C is a convex cone if it is convex and a cone: for any $x_1, x_2 \in C$ and $\theta_1, \theta_2 \geq 0$, we have $\theta_1 x_1 + \theta_2 x_2 \in C$.

The conic combination (or nonnegative linear combination) of k points x_1, \dots, x_k is defined as $\theta_1 x_1 + \cdots + \theta_k x_k$, where $\theta_1, \dots, \theta_k \geq 0$. If a set of finite points $\{x_i\}$, $i = 1, 2, \dots$ resides in a convex cone C , then every conic combination of $\{x_i\}$ remains in C . Conversely, a set C is a convex cone if and only if it contains all conic combinations of its elements. The conic hull of set C is the smallest convex cone that contains C . Fig. A.3 illustrates the conic hulls of two sets in \mathbb{R}^2 .

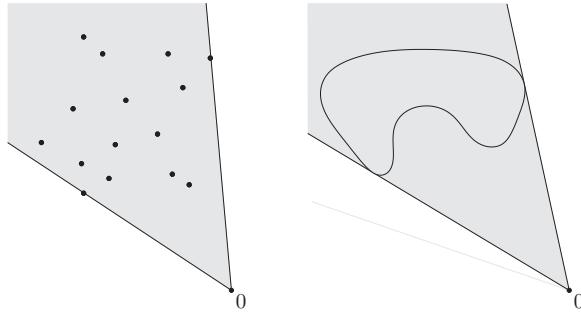


Fig. A.3 The conic hulls of the two sets [2].

Some widely used cones are introduced.

a. The nonnegative orthant

The nonnegative orthant is defined as

$$\mathbb{R}_+^n = \{x \in \mathbb{R}^n \mid x \geq 0\} \quad (\text{A.1})$$

It is the set of vectors composed of non-negative entries. It is clearly a convex cone.

b. Second-order cone

The unit second-order cone is defined as

$$\mathbb{L}_C^{n+1} = \{(x, t) \in \mathbb{R}^{n+1} \mid \|x\|_2 \leq t\} \quad (\text{A.2})$$

It is also called the Lorentz cone or ice-cream cone. Fig. A.4 exhibits \mathbb{L}_C^3 .

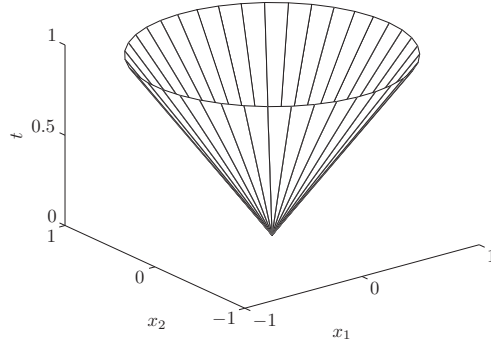


Fig. A.4 $\mathbb{L}_C^3 = \{(x_1, x_2, t) \mid \sqrt{x_1^2 + x_2^2} \leq t\}$ in \mathbb{R}^3 [2].

For any $(x, t) \in \mathbb{L}_C^{n+1}$ and $(y, z) \in \mathbb{L}_C^{n+1}$, we have

$$\|\theta_1 x + \theta_2 y\|_2 \leq \theta_1 \|x\|_2 + \theta_2 \|y\|_2 \leq \theta_1 t + \theta_2 z \Rightarrow \theta_1 \begin{bmatrix} x \\ t \end{bmatrix} + \theta_2 \begin{bmatrix} y \\ z \end{bmatrix} \in \mathbb{L}_C^{n+1}$$

which means that the unit second-order cone is a convex cone.

Sometimes, it is convenient to use the following inequality to represent a second-order cone in optimization problems

$$\|Ax + b\|_2 \leq c^T x + d \quad (\text{A.3})$$

where $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$, $d \in \mathbb{R}$. It is the inverse image of the unit second-order cone under the affine mapping $f(x) = (Ax + b, c^T x + d)$, and hence is convex. Second-order cones in forms of (A.2) and (A.3) are

interchangeable.

$$\|Ax + b\|_2 \leq c^T x + d \Leftrightarrow \begin{bmatrix} A \\ c^T \end{bmatrix} x + \begin{bmatrix} b \\ d \end{bmatrix} \in \mathbb{L}_C^{m+1}$$

and hence is convex.

c. Positive semidefinite cone

The set of symmetric $m \times m$ matrices is denoted by

$$\mathbb{S}^m = \{X \in \mathbb{R}^{m \times m} \mid X = X^T\}$$

which is a vector space with dimension $m(m+1)/2$.

The set of symmetric positive semidefinite matrices is denoted by

$$\mathbb{S}_+^m = \{X \in \mathbb{S}^m \mid X \succeq 0\}$$

The set of symmetric positive definite matrices is denoted by

$$\mathbb{S}_{++}^m = \{X \in \mathbb{S}^m \mid X \succ 0\}$$

Clearly, \mathbb{S}_+^m is a convex cone: if $A, B \in \mathbb{S}_+^m$, then for any $x \in \mathbb{R}^m$ and positive scalars $\theta_1, \theta_2 \geq 0$, we have

$$x^T(\theta_1 A + \theta_2 B)x = \theta_1 x^T A x + \theta_2 x^T B x \geq 0$$

implying $\theta_1 A + \theta_2 B \in \mathbb{S}_+^m$.

A positive semidefinite cone in \mathbb{R}^2 can be expressed via three variables x, y, z as

$$\begin{bmatrix} x & y \\ y & z \end{bmatrix} \succeq 0 \Leftrightarrow x \geq 0, \quad xz \geq y^2$$

which is plotted in Fig. A.5. In fact, \mathbb{L}_C^3 and \mathbb{S}_+^2 are equivalent to each other. To see this, the hyperbolic inequality $xz \geq y^2$ with $x \geq 0, z \geq 0$ defines the same feasible region in \mathbb{R}^3 as the following second-order cone

$$\left\| \begin{bmatrix} 2y \\ x - z \end{bmatrix} \right\|_2 \leq x + z, \quad x \geq 0, \quad z \geq 0$$

In higher-order dimensions, every second-order cone can be written as an LMI via Schur complement as

$$\|Ax + b\|_2 \leq c^T x + d \Rightarrow \begin{bmatrix} (c^T x + d)I & Ax + b \\ (Ax + b)^T & c^T x + d \end{bmatrix} \succeq 0 \quad (\text{A.4})$$

In this sense of representability, positive semidefinite cones are more general than second-order cones. However, the transformation in (A.4) may not

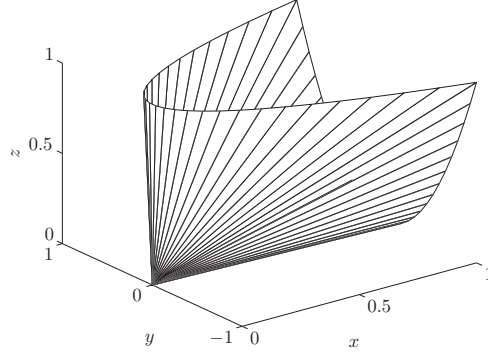


Fig. A.5 Positive semidefinite cone in \mathbb{S}^2 (or in \mathbb{R}^3) [2].

be superior from the computational perspective, because SOCPs are more tractable than SDPs.

d. Copositive cone

A copositive cone \mathbb{C}_+^n consists of symmetric matrices whose quadratic form is nonnegative over the nonnegative orthant \mathbb{R}_+^n :

$$\mathbb{C}_+^n = \{A \mid A \in \mathbb{S}^n, x^T A x \geq 0, \forall x \in \mathbb{R}_+^n\} \quad (\text{A.5})$$

The copositive cone \mathbb{C}_+^n is closed, pointed, and convex [4]. Clearly, $\mathbb{S}_+^n \subseteq \mathbb{C}_+^n$, and every entry-wise nonnegative symmetric matrix A belongs to \mathbb{C}_+^n . Actually, \mathbb{C}_+^n is significantly larger than the positive semidefinite cone and the nonnegative symmetric matrix cone.

2. Polyhedra

A polyhedron is defined as the solution set of a finite number of linear inequalities:

$$P = \{x \mid Ax \leq b\} \quad (\text{A.6})$$

(A.6) is also called a hyperplane representation for a polyhedron. It is easy to show that polyhedra are convex sets. Sometimes, a polyhedron is also called a polytope. The two concepts are often used interchangeably in this book. Because of physical bounds of decision variables, the polyhedral feasible regions in practical energy system optimization problems are usually bounded, which means that there is no extreme ray.

Polyhedra can be expressed via the convex combination as well. The convex hull of a finite number of points

$$\text{conv}\{v_1, \dots, v_k\} = \{\theta_1 v_1 + \dots + \theta_k v_k \mid \theta \geq 0, 1^T \theta = 1\} \quad (\text{A.7})$$

defines a polyhedron. (A.7) is called a convex hull representation. If the polyhedron is unbounded, a generalization of this convex hull representation is

$$\{\theta_1 v_1 + \cdots + \theta_k v_k \mid \theta \geq 0, \theta_1 + \cdots + \theta_m = 1, m \leq k\} \quad (\text{A.8})$$

which considers nonnegative linear combinations of v_i , but only the first m coefficients whose summation is 1 are bounded, and the remaining ones can take arbitrarily large values. In view of this, the convex hull of points v_1, \dots, v_m plus the conic hull of points v_{m+1}, \dots, v_k is a polyhedron. The reverse is also correct: any polyhedron can be represented by convex hull and conic hull.

How to represent a polyhedron depends on what information is available: if its boundaries are expressed via linear inequalities, the hyperplane representation is straightforward; if its extreme points and extreme rays are known in advance, the convex-conic hull representation is more convenient. With the growth in dimension, it is becoming more difficult to switch (derive one from the other) between the hyperplane representation and the hull representation.

A.1.2 Generalized Inequalities

A cone $K \subseteq \mathbb{R}^n$ is called a proper cone if it satisfies:

- 1) K is convex and closed.
- 2) K is solid, i.e., it has non-empty interior.
- 3) K is pointed, i.e., $x \in K, -x \in K \Rightarrow x = 0$.

A proper cone K can be used to define a generalized inequality, a partial ordering on \mathbb{R}^n , as follows

$$x \preceq_K y \iff y - x \in K \quad (\text{A.9})$$

We denote $x \succeq_K y$ for $y \preceq_K x$. Similarly, a strict partial ordering can be defined by

$$x \prec_K y \iff y - x \in \text{int}(K) \quad (\text{A.10})$$

where $\text{int}(K)$ stands for the interior of K , and write $x \succ_K y$ for $y \prec_K x$.

The nonnegative orthant \mathbb{R}_+^n is a proper cone. When $K = \mathbb{R}_+^n$, the partial ordering \preceq_K comes down to the element-wise comparison between vectors: for $x, y \in \mathbb{R}^n$, $x \preceq_{\mathbb{R}_+^n} y$ means $x_i \leq y_i$, $i = 1, \dots, n$, or the traditional notation $x \leq y$.

The positive semidefinite cone \mathbb{S}_+^n is a proper cone in \mathbb{S}^n . When $K = \mathbb{S}_+^n$, the partial ordering \preceq_K comes down to a linear matrix inequality between symmetric matrices: for $X, Y \in \mathbb{S}^n$, $X \preceq_{\mathbb{S}_+^n} Y$ means $Y - X$ is positive semidefinite. Because it arises so frequently, we can drop the subscript \mathbb{S}_+^n when we write a linear matrix inequality $Y \succeq X$ or $X \preceq Y$. It is understood

that such a generalized inequality corresponds to the positive semidefinite cone without particular mention.

A generalized inequality is equivalent to linear constraints with $K = \mathbb{R}_+^n$; for other cones, such as the second-order cone \mathbb{L}_C^{n+1} or the positive semidefinite cone \mathbb{S}_+^n , the feasible region is nonlinear but remains convex.

A.1.3 Dual Cones and Dual Generalized Inequalities

Let K be a cone in \mathbb{R}^n . Its dual is defined as the following set

$$K^* = \{y \mid x^T y \geq 0, \forall x \in K\} \quad (\text{A.11})$$

Because K^* is the intersection of homogeneous half spaces (half spaces passing through the origin). It is a closed convex cone.

The interior of K^* is given by

$$\text{int}(K^*) = \{y \mid x^T y > 0, \forall x \in K, x \neq 0\} \quad (\text{A.12})$$

To see this, if $y^T x > 0, \forall x \in K$, then $(y + u)^T x > 0, \forall x \in K$ holds for all u that is sufficiently small; hence $y \in \text{int}(K^*)$. Conversely, if $y \in K^*$ and $\exists x \in K : y^T x = 0, x \neq 0$, then $(y - tx)^T x < 0, \forall t > 0$, indicating $y \notin \text{int}(K^*)$.

If $\text{int}(K) \neq \emptyset$, then K^* is pointed. If this is not true, suppose $\exists y \neq 0 : y \in K^*, -y \in K^*$, i.e., $y^T x \geq 0, \forall x \in K$ and $-y^T x \geq 0, \forall x \in K$, so we have $x^T y = 0, \forall x \in K$, which is in contradiction with $\text{int}(K) \neq \emptyset$.

In conclusion, K^* is a proper cone, if the original cone K is so; K^* is closed and convex, regardless of the original cone K . Fig. A.6 shows a cone K (the region between L_2 and L_3) and its dual cone K^* (the region between L_1 and L_4) in \mathbb{R}^2 .

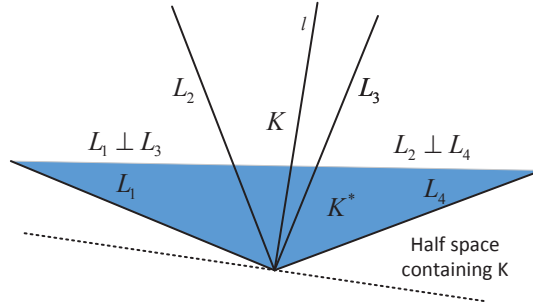


Fig. A.6 Illustration of a cone and its dual cone in \mathbb{R}^2 .

In light of the definition of K^* , a non-zero vector y is the normal of a homogeneous half space which contains K if and only if $y \in K^*$. The intersection of all such half spaces containing K constitutes the cone K (if K is closed), in view of this

$$K = \bigcap_{y \in K^*} \{x \mid y^T x \geq 0\} = \{x \mid y^T x \geq 0, \forall y \in K^*\} = K^{**} \quad (\text{A.13})$$

This fact can be also understood in \mathbb{R}^2 from Fig. A.6. The extreme cases for the normal vector y such that the corresponding half space contains K are L_1 and L_4 , and the intersection of these half spaces for all $y \in K^*$ turns out to be the original cone K .

Next, we investigate the dual cones of three special proper cones, i.e., \mathbb{R}_+^n , \mathbb{L}_C^{n+1} , and \mathbb{S}_+^n , respectively.

1. The nonnegative orthant

By observing the fact

$$x^T y \geq 0, \forall x \geq 0 \iff y \geq 0$$

we naturally have $(\mathbb{R}_+^n)^* = \mathbb{R}_+^n$; in other words, the nonnegative orthant is self-dual.

2. The second-order cone

Now, we show that the second-order cone is also self-dual: $(\mathbb{L}_C^{n+1})^* = \mathbb{L}_C^{n+1}$. To this end, we need to demonstrate

$$x^T u + tv \geq 0, \forall (x, t) \in \mathbb{L}_C^{n+1} \iff \|u\|_2 \leq v$$

\Rightarrow : Suppose the right-hand condition is false, and $\exists (u, v) : \|u\|_2 > v$, by recalling Cauchy-Schwarz inequality $|a^T b| \leq \|a\|_2 \|b\|_2$, we have

$$\min_x \{x^T u \mid \text{s.t. } \|x\|_2 \leq t\} = -t\|u\|_2$$

In such circumstance, $x^T u + tv = t(v - \|u\|_2) < 0, \forall t > 0$, which is in contradiction with the left-hand condition.

\Leftarrow : Again, according to Cauchy-Schwarz inequality, we have

$$x^T u + tv \geq -\|x\|_2 \|u\|_2 + tv \geq -\|x\|_2 \|u\|_2 + \|x\|_2 v = \|x\|_2 (v - \|u\|_2) \geq 0$$

3. The positive semidefinite cone

We investigate the dual cone of \mathbb{S}_+^n . The inner product of $X, Y \in \mathbb{S}^n$ is defined by the element-wise summation

$$\langle X, Y \rangle = \sum_{i=1}^n \sum_{j=1}^n X_{ij} Y_{ij} = \text{tr}(XY^T)$$

We establish this fact: $(\mathbb{S}_+^n)^* = \mathbb{S}_+^n$, which boils down to

$$\text{tr}(XY^T) \geq 0, \forall X \succeq 0 \iff Y \succeq 0$$

\Rightarrow : Suppose $Y \notin \mathbb{S}_+^n$, then $\exists q \in \mathbb{R}^n$ such that

$$q^T Y q = \text{tr}(qq^T Y^T) < 0$$

which is in contradiction with the left-hand condition because $X = qq^T \in \mathbb{S}_+^n$.

\Leftarrow : Now suppose $X, Y \in \mathbb{S}_+^n$. X can be expressed via its eigenvalues $\lambda_i \geq 0$ and eigenvectors q_i as $X = \sum_{i=1}^n \lambda_i q_i q_i^T$, then we arrive at

$$\text{tr}(XY^T) = \text{tr}\left(Y \sum_{i=1}^n \lambda_i q_i q_i^T\right) = \sum_{i=1}^n \lambda_i q_i^T Y q_i \geq 0$$

In summary, it follows that the positive semidefinite cone is self-dual.

4. The completely positive cone

Following the same concept of matrix inner product, it is shown that $(\mathbb{C}_+^n)^*$ is the cone of so-called completely positive matrices and can be expressed as [5]

$$(\mathbb{C}_+^n)^* = \text{conv}\{xx^T \mid x \in \mathbb{R}_+^n\} \quad (\text{A.14})$$

In contrast to previous three cones, the copositive cone \mathbb{C}_+^n is not self-dual.

When the dual cone K^* is proper, it induces a generalized inequality \preceq_{K^*} , which is called the dual generalized inequality of the one induced by cone K (if K is proper). According to the definition of dual cone, an important fact relating a generalized inequality and its dual is

- 1) $x \preceq_K y$ if and only if $\lambda^T x \leq \lambda^T y, \forall \lambda \in K^*$.
- 2) $x \prec_K y$ if and only if $\lambda^T x < \lambda^T y, \forall \lambda \in K^*, \lambda \neq 0$.

When $K = K^{**}$, the dual generalized inequality of \preceq_{K^*} is \preceq_K , and the above property holds if the positions of K and K^* are swapped.

A.1.4 Convex Function and Epigraph

A function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex if its feasible region X is a convex set, and for all $x_1, x_2 \in X$, the following condition holds

$$f(\theta x_1 + (1 - \theta)x_2) \leq \theta f(x_1) + (1 - \theta)f(x_2), \forall \theta \in [0, 1] \quad (\text{A.15})$$

The geometrical interpretation of inequality (A.15) is that the chord connecting points $(x_1, f(x_1))$ and $(x_2, f(x_2))$ always lies above the curve of f between x_1 and x_2 (see Fig. A.7). Function f is strictly convex if strict inequality holds in (A.15) when $x_1 \neq x_2$ and $0 < \theta < 1$. Function f is called (strictly) concave if $-f$ is (strictly) convex. An affine function is both convex and concave.

The graph of a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is defined as

$$\text{graph } f = \{(x, f(x)) \mid x \in X\} \quad (\text{A.16})$$

which is a subset of \mathbb{R}^{n+1} .

The epigraph of a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is defined as

$$\text{epi } f = \{(x, t) \mid x \in X, f(x) \leq t\} \quad (\text{A.17})$$

which is a subset of \mathbb{R}^{n+1} . These definitions are illustrated through Fig. A.7.

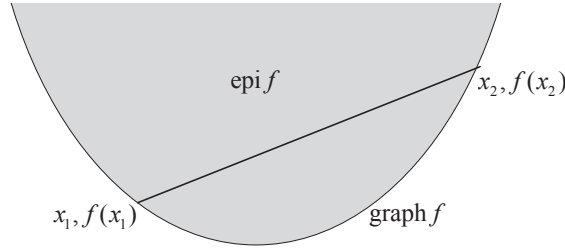


Fig. A.7 Illustration of the graph of a convex function $f(x)$ (the solid line) and its epigraph (the shaded area) in \mathbb{R}^2 .

Epigraph bridges the concepts of convex sets and convex functions: A function is convex if and only if its epigraph is a convex set. Epigraph is frequently used in formulating optimization problems. A nonlinear objective function can be replaced by a linear objective and an additional constraint in epigraph form. In this sense, we can assume that any optimization problem has a linear objective function. Nonetheless, this does not facilitate solving the problem, as non-convexity moves to the constraints, if the objective function is not convex. Nonetheless, the solution to an optimization problem with a linear objective can always be found at the boundary of the convex hull of its feasible region, implying that if we can characterize the convex hull, a problem in epigraph form admits an exact convex hull relaxation. However, in general, it is difficult to express convex hull in an analytical form.

Analyzing convex functions is a well developed field. Broadening the knowledge in convex analysis could be mathematically demanding, especially for readers who are primarily interested in applications. We will not pursue in so-

phisticated theories in depth any more. Readers are referred to the literature suggested at the end of this chapter for further information.

A.2 From Linear to Conic Program

Linear programming is one of the most mature and tractable mathematical programming problems. In this section, we first investigate and explain the motivation of linear programming duality theory, then provide a unified model for conic programming problems. LPs, SOCPs, and SDPs are special cases of conic programs associated with generalized inequalities \preceq_K where $K = \mathbb{R}^n$, \mathbb{L}_C^{n+1} , and \mathbb{S}_+^n , respectively. Our aim is to help readers who are not familiar with conic programs build their decision-making problems in these formats with structured convexity, and write out their dual problems more conveniently. The presentation logic is consistent with [1], and most of the presented materials in this section also come from [1].

A.2.1 Linear Program and its Duality Theory

A linear program is an optimization program with the form

$$\min\{c^T x \mid Ax \geq b\} \tag{A.18}$$

where x is the vector of decision variables, A , b , c are constant coefficient matrices with compatible dimensions. We assume LP (A.18) is feasible, i.e., its feasible set $X = \{x \mid Ax \geq b\}$ is a non-empty polyhedron; moreover, because of the limited ranges of decision variables representing physical quantities, we assume X is bounded. In such circumstance, LP (A.18) always has a finite optimum. LPs can be solved by mature algorithms, such as the simplex algorithm and the interior-point algorithm, which are not the main focus of this book.

A question which is important both in theory and practice is: how to find a systematic way to bound the optimal value of (A.18)? Clearly, if x is a feasible solution, an instant upper bound is given by $c^T x$. Lower bounding is to find a value a , such that $c^T x \geq a$ holds for all $x \in X$.

A trivial answer is to solve the problem and retrieve its optimal value, which is the tightest lower bound. However, there may be a smarter way to retrieve a valid lower bound with much cheaper computational expense. To outline the basic motivation, let us consider the following example

$$\min \left\{ \sum_{i=1}^6 x_i \mid \begin{array}{l} 2x_1 + 1x_2 + 3x_3 + 8x_4 + 5x_5 + 3x_6 \geq 5 \\ 6x_1 + 2x_2 + 6x_3 + 1x_4 + 1x_5 + 4x_6 \geq 2 \\ 2x_1 + 7x_2 + 1x_3 + 1x_4 + 4x_5 + 3x_6 \geq 1 \end{array} \right\} \quad (\text{A.19})$$

Although LP (A.19) is merely a toy case for modern solvers and computers, one may guess it is still a little bit complicated for mental arithmetic. In fact, we can claim the optimal value is 0.8 at a glance without any sophisticated calculation: summing up the three constraints yields an inequality

$$10(x_1 + x_2 + x_3 + x_4 + x_5 + x_6) \geq 8 \quad (\text{A.20})$$

which immediately gives the optimal value is 0.8. To understand why such a value is indeed the optimum, by adding the constraints together and dividing both sides by 10, inequality (A.20) implies that the objective function must get a value which is greater than or equal to 0.8 at any feasible point; moreover, to demonstrate that 0.8 is attainable, we can find a point x^* which activates the three constraints simultaneously, so (A.20) becomes an equality. LP duality is merely a formal generalization of this simple trick.

Multiplying each constraint in $Ax \geq b$ with a non-negative weight λ_i , and adding all constraints together, we will see

$$\lambda^T Ax \geq \lambda^T b$$

If we choose λ elaborately such that $\lambda^T A = c^T$, then $\lambda^T b$ will be a valid lower bound of the optimal value of (A.18). To improve the lower bound estimation, one may optimize the weighting vector λ , giving rise to the following problem

$$\max_{\lambda} \{ \lambda^T b \mid A^T \lambda = c, \lambda \geq 0 \} \quad (\text{A.21})$$

where λ is the vector of decision variables or dual variables, and the feasible region $D = \{ \lambda \mid A^T \lambda = c, \lambda \geq 0 \}$ is a polyhedron. Clearly, (A.21) is also an LP, and is called the dual problem of LP (A.18). Correspondingly, (A.18) is called the primal problem. From above construction, we immediately conclude $c^T x \geq \lambda^T b$.

Proposition A.1. (*Weak duality*): *The optimal value of (A.21) is less than or equal to the optimal value of (A.18).*

In fact, the optimal bound offered by (A.21) is tight.

Proposition A.2. (*Strong duality*): *Optimal values of (A.21) and (A.18) are equal.*

To see this, an explanation is given in [1]. If a real number a is the optimal value of the primal LP (A.18), the system of linear inequalities

$$S^P : \begin{cases} -c^T x > -a : \lambda_0 \\ Ax \geq b : \lambda \end{cases}$$

must have an empty solution set, indicating that at least one of the following two systems does have a solution (called separation property later)

$$S_1^D : \begin{cases} -\lambda_0 c + A^T \lambda = 0 \\ -\lambda_0 a + b^T \lambda \geq 0 \\ \lambda_0 > 0, \quad \lambda \geq 0 \end{cases}$$

$$S_2^D : \begin{cases} -\lambda_0 c + A^T \lambda = 0 \\ -\lambda_0 a + b^T \lambda > 0 \\ \lambda_0 \geq 0, \quad \lambda \geq 0 \end{cases}$$

We can show that S^P has no solutions if and only if S_1^D has a solution.

S_1^D has a solution $\Rightarrow S^P$ has no solution is clear. Otherwise, suppose that S^P has a solution x , because λ_0 is strictly positive, the weighted summation of inequalities in S^P leads to

$$0 = 0^T x = (-\lambda_0 c + A^T \lambda)^T x = -\lambda_0 c^T x + \lambda^T A x > -\lambda_0 a + \lambda^T b$$

which is in contradiction with the second inequality in S_1^D .

S^P has no solution $\Rightarrow S_1^D$ has a solution. Suppose S_1^D has no solution, S_2^D must have a solution owing to the separation property (Theorem 1.2.1 in [1]). Moreover, if $\lambda_0 > 0$, the solution of system S_2^D also solves system S_1^D , so there must be $\lambda_0 = 0$. As a result, the solution of S_2^D is independent of the values of a and c . Let $c = 0$ and $a = 0$, the solution λ of S_2^D satisfies $A^T \lambda = 0$, $b^T \lambda > 0$. Therefore, for any x with a compatible dimension, $\lambda^T (Ax - b) = \lambda^T Ax - \lambda^T b < 0$ holds. In addition, because $\lambda \geq 0$, we can conclude that $Ax \geq b$ has no solution, a contradiction to the assumption that (A.18) is feasible.

Now, consider the solution of S_1^D . Without loss of generality, we can assume $\lambda_0 = 1$; otherwise, if $\lambda_0 \neq 1$, $(1, \lambda/\lambda_0)$ also solves S_1^D . In view of this, in normalized condition ($\lambda_0 = 1$), S_1^D comes down to

$$S_3^D : \begin{cases} A^T \lambda = c \\ b^T \lambda \geq a \\ \lambda \geq 0 \end{cases}$$

Now we can see the strong duality: Let a^* be the optimal solution of (A.18). For any $a < a^*$, S^P has no solution, so S_1^D has a solution $(1, \lambda^*)$. According to S_3^D , the optimal value of (A.21) is no smaller than a , i.e., $a \leq b^T \lambda^* \leq a^*$. When a tends to a^* , we can conclude that the primal and dual optimal values are equal. Since the primal problem always has a finite optimum (as we assumed before), so does the dual problem, as they share the same optimal value. Nevertheless, even if the primal feasible region is bounded, the dual feasible set D may be unbounded, and the dual problem is always bounded

above. Please refer to [29, 30, 31] for more information on duality theory in linear programming.

Proposition A.3. (*Primal-dual optimality condition*) *If LP (A.18) is feasible and X is bounded, then any feasible solution to the following system*

$$\begin{aligned} Ax &\geq b \\ A^T \lambda &= c, \lambda \geq 0 \\ c^T x &= b^T \lambda \end{aligned} \tag{A.22}$$

solves the original primal-dual pair of LPs: x^ is the optimal solution of (A.18), and λ^* is the optimal solution of (A.21).*

(A.22) is also called the primal-dual optimality condition of LPs. It consists of linear inequalities and equalities, and there is no objective function to be optimized.

Substituting $c = A^T \lambda$ into the last equation of (A.22) gives $\lambda^T Ax = \lambda^T b$, i.e.

$$\lambda^T (b - Ax) = 0$$

Since $\lambda \geq 0$ and $Ax \geq b$, above equation is equivalent to

$$\lambda_i (b - Ax)_i = 0$$

where notation $(b - Ax)_i$ and λ_i stand for the i -th components of vectors $b - Ax$ and λ , respectively. This condition means that at most one of λ_i and $(b - Ax)_i$ can take a strictly positive value. In other words, if the i -th inequality constraint is inactive, then its dual multiplier λ_i must be 0; otherwise, if $\lambda_i > 0$, then the corresponding inequality constraint must be binding. This phenomenon is called the complementarity and slackness condition.

Applying KKT optimality condition for general nonlinear programs to LP (A.18) we have:

Proposition A.4. (*KKT optimality condition*) *If LP (A.18) is feasible and X is bounded, the following system*

$$\begin{aligned} 0 &\leq \lambda \perp Ax - b \leq 0 \\ A^T \lambda &= c \end{aligned} \tag{A.23}$$

has a solution (x^, λ^*) (may not be unique), where $a \perp b$ means $a^T b = 0$, x^* solves (A.18) and λ^* solves (A.21).*

The question that which one of (A.22) and (A.23) is better can be subtle and has very different practical consequences. At the first look, the former one seems more tractable because (A.22) is a linear system while (A.23) contains complementarity and slackness conditions. However, the actual situation in practice is more complicated. For example, to solve a bilevel program with an

LP lower level, the LP is often replaced by its optimality condition. In a bilevel optimization structure, some of the coefficients A , b , and c are optimized by the upper-level agent, say, the coefficient vector c representing the price is controlled by the upper level decision maker, while A and b are constants. If we use (A.22), the term $c^T x$ in the single-level equivalence becomes non-convex, although c is a constant in the lower level, preventing a global optimal solution from being found easily. In contrast to this, if we use (A.23) and linearize the complementarity and slackness condition via auxiliary integer variables, the single-level equivalent problem can be formulated as an MILP, whose global optimal solution can be procured with reasonable computation effort.

The dual problem of LPs which maximize its objective can be derived in the same way. Consider the LP

$$\max\{c^T x \mid Ax \leq b\} \quad (\text{A.24})$$

For this problem, we need an upper bound on the objective function. To this end, associating a non-negative dual vector λ with the constraint, and adding the weighted inequalities together, we have

$$\lambda^T Ax \leq \lambda^T b$$

If we intentionally choose λ such that $\lambda^T A = c^T$, then $\lambda^T b$ will be a valid upper bound of the optimal value of (A.24). The dual problem

$$\min_{\lambda} \{\lambda^T b \mid A^T \lambda = c, \lambda \geq 0\} \quad (\text{A.25})$$

optimizes the weighting vector λ to offer the tightest upper bound.

Constraints in the form of equality and \geq inequality can be considered using the same paradigm. Bearing in mind that we are seeking an upper bound, so we need a certification for $c^T x \leq a$, so the dual variables for equalities have no signs and those for \geq inequalities should be negative.

Sometimes it is useful to define the dual cone of a polyhedron, despite that a bounded polyhedron is not a cone. Recall its definition, the dual cone of a polyhedron P can be defined as

$$P^* = \{y \mid x^T y \geq 0, \forall x \in P\} \quad (\text{A.26})$$

where $P = \{x \mid Ax \geq b\}$. As we have demonstrated in Sect. A.1.3, the dual cone is always closed and convex; however, for a general set, its dual cone does not have an analytical expression.

For polyhedral sets, the condition in (A.26) holds if and only if the minimal value of $x^T y$ over P is non-negative. For a given vector y , let us investigate the minimum of $x^T y$ through an LP

$$\min_x \{y^T x \mid Ax \geq b\}$$

It is known from Proposition A.1 that $y^T x \geq b^T \lambda$, $\forall \lambda \in D_P$, where $D_P = \{\lambda \mid A^T \lambda = y, \lambda \geq 0\}$. Moreover, if $\exists \lambda \in D_P$ such that $b^T \lambda < 0$, Proposition A.2 certifies the existence of $x \in P$ such that $y^T x = b^T \lambda < 0$. In conclusion, the dual cone of polyhedron P can be cast as

$$P^* = \{y \mid \exists \lambda : b^T \lambda \geq 0, A^T \lambda = y, \lambda \geq 0\} \quad (\text{A.27})$$

which is also a polyhedron. It can be observed from (A.27) that all constraints in P^* are homogeneous, so P^* is indeed a polyhedral cone.

A.2.2 General Conic Linear Program

Linear programs cover vast topics in engineering optimization problems. Its duality program provides informative quantifications and valuable insights of the problem at hand, which help develop efficient algorithms for itself and facilitate building tractable reformulations for more complicated mathematical programming models, such as robust optimization, multi-level optimization and equilibrium problems. The algorithms of LPs, which are perfectly developed by now, can solve quite large instances (with up to hundreds of thousands of variables and constraints). Nevertheless, there are practical problems which cannot be modeled by LPs. To cope with these essentially nonlinear cases, one needs to explore new models and computational methods beyond the reach of LPs.

The broadest class of optimization problems which the LP can be compared with is the class of convex optimization problems. Convexity marks whether a problem can be solved efficiently, and any local optimizer of a convex program must be a global optimizer. Efficiency is quantified by the number of arithmetic operations required to solve the problem. Suppose that all we know about the problem is its convexity: its objective and constraints are convex functions in decision variables $x \in \mathbb{R}^n$, and their values along with their derivatives at any given point can be evaluated within M arithmetic operations. The best known complexity for finding an ϵ -solution turns out to be [1]

$$O(1)n(n^3 + M) \ln \left(\frac{1}{\epsilon} \right)$$

Although this bound grows polynomially with n , the computation time may be still unacceptable for a large n like $n = 1,000$, which is in contrast to LPs which are solvable with $n = 100,000$. The reason is: linearity are much stronger than convexity; the structure of an affine function $a^T x + b$ solely depends on its constant coefficients a and b ; function values and derivatives are never evaluated in a state-of-the-art LP solver. There are many classes of convex programs which are essentially nonlinear, but still possess nice analytical structure, which can be used to develop more dedicated algorithms.

These algorithms may perform much more efficiently than those exploiting only convexity. In what follows, we consider such a class of convex program, i.e., the conic program, which is a simple extension of LP. Its general form and mathematical model are briefly introduced, while the details about interior-point algorithms is beyond the scope of this book, which can be found in [1, 2].

1. Mathematical model

When we consider to add some nonlinear factors in LP (A.18), the most common way is to replace a linear function $a^T x$ with a nonlinear but convex function $f(x)$. As what has been explained, this may not be advantageous from a computational perspective. In contrast to this, we sustain all functions to be linear, but inject nonlinearity in the comparative operators \geq or \leq . Recall the definition of generalized inequalities \succeq_K with cone K , we consider the following problem in this section

$$\min_x \{c^T x \mid Ax \succeq_K b\} \quad (\text{A.28})$$

which is called a conic programming problem. An LP is a special case of the conic program with $K = \mathbb{R}_+^n$. With this generalization, we are able to formulate a much wider spectrum of optimization problems which cannot be modeled as LPs, while enjoy nice properties of structured convexity.

2. Conic duality

Aside from developing high-performance algorithms, the most important and elegant theoretical result in the area of LP is its duality theorem. In view of their similarities in mathematical appearances, how can the LP duality theorem be extended to conic programs? Similarly, the motivation of duality is the desire of a systematic way to certify a lower bound on the optimal value of conic program (A.28). Let us try the same trick: multiplying the dual vector λ on both sides of $Ax \succeq_K b$, and adding them together, we obtain $\lambda^T Ax$ and $b^T \lambda$; moreover, if we are lucky to get $A^T \lambda = c$, we guess $b^T \lambda$ can serve as a lower bound of the optimum of (A.28) under some condition. The condition can be translated into: what is the admissible region of λ , such that the inequality $\lambda^T Ax \geq b^T \lambda$ is a consequence of $Ax \succeq_K b$? A nice answer has been given at the end of Sect. A.1.3. Let us explain the problem from some simple cases.

Particularly, when $K = \mathbb{R}_+^n$, the admissible region of λ is also \mathbb{R}_+^n , because we have already known the fact that the dual variable of \geq inequalities in an LP which minimizes its objective should be non-negative. However, \mathbb{R}_+^n is no longer a feasible region of λ for conic programs with generalized inequality \succeq_K if $K \neq \mathbb{R}_+^n$. To see this, consider \mathbb{L}_C^3 and the corresponding generalized inequality

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} \succeq_{\mathbb{L}_C^3} \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \iff z \geq \sqrt{x^2 + y^2}$$

$(x, y, z) = (-1, -1, 1.5)$ is a feasible solution. However, the weighted summation of both sides with $\lambda = [1, 1, 1]^T$ gives a false inequality $-0.5 \geq 0$.

To find the feasible region of λ , consider the condition

$$\forall a \succeq_K 0 \Rightarrow \lambda^T a \geq 0 \quad (\text{A.29})$$

If (A.29) is true, we have the following logical inferences

$$\begin{aligned} & Ax \succeq_K b \\ \Leftrightarrow & Ax - b \succeq_K 0 \\ \Rightarrow & \lambda^T (Ax - b) \geq 0 \\ \Leftrightarrow & \lambda^T Ax \geq \lambda^T b \end{aligned}$$

Conversely, if λ is an admissible vector for certifying

$$\forall (a, b : a \succeq_K b) \Rightarrow \lambda^T a \geq \lambda^T b$$

then, (A.29) is clearly true by letting $b = 0$. Therefore, the admissible set of λ for generalized inequality \succeq_K with cone K can be written as

$$K^* = \{\lambda \mid \lambda^T a \geq 0, \forall a \in K\} \quad (\text{A.30})$$

which contains vectors whose inner products with all vectors belonging to K are nonnegative. Recall the definition in (A.11), we can observe that the set K^* is actually the dual cone of cone K .

Now we are ready to setup the dual problem of conic program (A.28). As in the case of LP duality, we try to recover the objective function from the linear combination of constraints by choosing a proper dual variable λ , i.e., $\lambda^T Ax = c^T x$, in addition, $\lambda \in K^*$ ensures $\lambda^T Ax \geq \lambda^T b$, implying that $\lambda^T b$ is a valid lower bound of the objective function. The best bound one can expect is the optimum of the problem

$$\max_{\lambda} \{b^T \lambda \mid A^T \lambda = c, \lambda \succeq_{K^*} 0\} \quad (\text{A.31})$$

which is also a conic program, and called the dual problem of conic program (A.28). From above construction, we have already known that $c^T x \geq b^T \lambda$ is satisfied for all feasible x and λ , which is the weak duality of conic programs.

In fact, the primal-dual pair of conic programs has following properties:

Proposition A.5. (Conic Duality Theorem) [1] : The following conclusions hold true for conic program (A.28) and its dual (A.31).

1) Conic duality is symmetric: the dual problem is still a conic one, and the primal and dual problems are dual to each other.

2) *Weak duality holds: the duality gap $c^T x - b^T \lambda$ is nonnegative over the primal and dual feasible sets.*

2) *If either of the primal problem or the dual problem is strictly feasible and has a finite optimum, then the other is solvable, and the duality gap is zero: $c^T x^* = b^T \lambda^*$ for some x^* and λ^* .*

3) *If either of the primal problem or the dual problem is strictly feasible and has a finite optimum, then a pair of primal-dual feasible solutions (x, λ) solves the respective problems if and only if*

$$\begin{aligned} Ax &\succeq_K b \\ A^T \lambda &= c \\ \lambda &\succeq_{K^*} 0 \\ c^T x &= b^T \lambda \end{aligned} \tag{A.32}$$

or

$$\begin{aligned} 0 &\preceq_{K^*} \lambda \perp Ax - b \succeq_K 0 \\ A^T \lambda &= c \end{aligned} \tag{A.33}$$

where (A.32) is called the primal-dual optimality condition, and (A.33) is called the KKT optimality condition.

The proof can be found in [1] and is omitted here. To highlight the role of strict feasibility in Proposition A.5, consider the following example

$$\min_x \left\{ x_2 \mid \begin{bmatrix} x_1 \\ x_2 \\ x_1 \end{bmatrix} \succeq_{\mathbb{L}_C^3} 0 \right\}$$

The feasible region is

$$\sqrt{x_1^2 + x_2^2} \leq x_1 \Leftrightarrow x_2 = 0, x_1 \geq 0$$

So its optimal value is 0. As explained before, second-order cones are self-dual: $(\mathbb{L}_C^3)^* = \mathbb{L}_C^3$, it is easy to see the dual problem is

$$\max_{\lambda} \left\{ 0 \mid \lambda_1 + \lambda_3 = 0, \lambda_2 = 1, \lambda \succeq_{\mathbb{L}_C^3} 0 \right\}$$

The feasible region is

$$\left\{ \lambda \mid \sqrt{\lambda_1^2 + \lambda_2^2} \leq \lambda_3, \lambda_3 \geq 0, \lambda_2 = 1, \lambda_1 = -\lambda_3 \right\}$$

which is empty, because $\sqrt{(-\lambda_3)^2 + 1} > \lambda_3$.

This example demonstrates that the existence of a strictly feasible point is indispensable for conic duality. But this condition is not necessary in LP

duality, which means strong duality holds in conic programming with stronger assumptions.

Several classes of conic programs with particular cones are of special interests. The cones in these problems are self-dual, so we can set up the dual program directly, which allows to explore deeply into the original problem, or convert it into equivalent formulations which are more computationally friendly. The structure of these relatively simple cones also helps develop efficient algorithms for corresponding conic programs. In what follows, we will investigate two extremely important classes of conic programs.

A.2.3 Second-order Cone Program

1. Mathematical models of the primal and dual problems

Second-order cone program is a special class of conic problem with $K = \mathbb{L}_C^{n+1}$. It minimizes a linear function over the intersection of a polytope and the Cartesian product of second-order cones, and can be formulated as

$$\min_x \{c^T x \mid Ax - b \succeq_K 0\} \quad (\text{A.34})$$

where $x \in \mathbb{R}^n$, and $K = \mathbb{L}_C^{m_1} \times \cdots \times \mathbb{L}_C^{m_k} \times \mathbb{R}_+^{m_p}$, in other words, the conic constraints in (A.34) can be expressed as k second-order cones $A_i x - b_i \succeq_{\mathbb{L}^{m_i}} 0$, $i = 1, \dots, k$ plus one polyhedron $A_p x - b_p \geq 0$ with the following matrix partition

$$[A; b] = \begin{bmatrix} [A_1; b_1] \\ \vdots \\ [A_k; b_k] \\ [A_p; b_p] \end{bmatrix}$$

Recall the definition of second-order cone, we further partition the submatrices A_i, b_i into

$$[A_i; b_i] = \begin{bmatrix} D_i & d_i \\ p_i^T & q_i \end{bmatrix}, \quad i = 1, \dots, k$$

where $D_i \in \mathbb{R}^{(m_i-1) \times n}$, $p_i \in \mathbb{R}^n$, $d_i \in \mathbb{R}^{m_i-1}$, $q_i \in \mathbb{R}$. Then we can write (A.34) as

$$\begin{aligned} \min_x \quad & c^T x \\ \text{s.t.} \quad & A_p x \geq b_p \\ & \|D_i x - d_i\|_2 \leq p_i^T x - q_i, \quad i = 1, \dots, k \end{aligned} \quad (\text{A.35})$$

(A.35) is often more convenient for model builders.

It is easy to see that the cone K in (A.34) is self-dual, as both second-order cone and non-negative orthant are self-dual. In this regard, the dual problem of SOCP (A.34) can be expressed as

$$\max_{\lambda} \{b^T \lambda \mid A^T \lambda = c, \lambda \succeq_K 0\} \quad (\text{A.36})$$

Partitioning the dual vector as

$$\lambda = \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_k \\ \lambda_p \end{bmatrix}, \quad \lambda_i \in \mathbb{L}_C^{m_i}, \quad i = 1, \dots, k, \quad \lambda_p \geq 0$$

We can write the dual problem as

$$\begin{aligned} \max_{\lambda} \quad & \sum_{i=1}^k b_i^T \lambda_i + b_p^T \lambda_p \\ \text{s.t.} \quad & \sum_{i=1}^k A_i^T \lambda_i + A_p^T \lambda_p = c \\ & \lambda_i \in \mathbb{L}_C^{m_i}, \quad i = 1, \dots, k \\ & \lambda_p \geq 0 \end{aligned} \quad (\text{A.37})$$

We further partition λ_i according to the norm representation in (A.35)

$$\lambda_i = \begin{bmatrix} \mu_i \\ \nu_i \end{bmatrix}, \quad \mu_i \in \mathbb{R}^{m_i-1}, \quad \nu_i \in \mathbb{R}$$

all second-order cone constraints are associated with dual variables as

$$\begin{bmatrix} D_i x \\ p_i^T x \end{bmatrix} - \begin{bmatrix} d_i \\ q_i \end{bmatrix} \in \mathbb{L}_C^{m_i} : \begin{bmatrix} \mu_i \\ \nu_i \end{bmatrix}, \quad i = 1, \dots, k$$

so the admissible region of dual variables (μ_i, ν_i) is

$$\begin{bmatrix} \mu_i \\ \nu_i \end{bmatrix} \in (\mathbb{L}_C^{m_i})^* \Rightarrow \|\mu_i\|_2 \leq \nu_i$$

Finally, we arrive at the dual form of (A.35)

$$\begin{aligned}
& \max_{\lambda} \quad \sum_{i=1}^k (\mu_i^T d_i + \nu_i q_i) + b_p^T \lambda_p \\
& \text{s.t.} \quad \sum_{i=1}^k (D_i^T \mu_i + \nu_i p_i) + A_p^T \lambda_p = c \\
& \quad \|\mu_i\|_2 \leq \nu_i, \quad i = 1, \dots, k \\
& \quad \lambda_p \geq 0
\end{aligned} \tag{A.38}$$

(A.35) and (A.38) are more convenient than (A.34) and (A.37) respectively because norm constraints can be recognized by most commercial solvers, whereas generalized inequalities \succeq_K and constraints with the form $\in \mathbb{L}_C^{m_i}$ are supported only in some dedicated packages. Strict feasibility can be expressed in a more straightforward manner via norm constraints: the primal problem is strictly feasible if $\exists x : \|D_i x - d_i\|_2 < p_i^T x - q_i, \quad i = 1, \dots, k, \quad A_p x > b_p$; the dual problem is strictly feasible if $\|\mu_i\|_2 < \nu_i, \quad i = 1, \dots, k, \quad \lambda_p > 0$. In view of this, (A.35) and (A.38) are treated as the standard forms of an SOCP and its dual by practitioners whose primary interests are applications.

2. What can be expressed via SOCPs?

Mathematical programs raised in engineering applications may not always appear in standard convex forms, and convexity may be hidden in seemingly non-convex expressions. Therefore, an important step is to recognize the potential existence of a convex form that is equivalent to the original formulation. This task can be rather tricky. We introduce some frequently used functions and constraints that can be represented by second-order cone constraints.

a. Convex quadratic constraints

A convex quadratic constraint has the form

$$x^T P x + q^T x + r \leq 0 \tag{A.39}$$

where $P \in \mathbb{S}_+^n$, $q \in \mathbb{R}^n$, $r \in \mathbb{R}$ are constant coefficients. Let $t = q^T x + r$, we have

$$t = \frac{(t+1)^2}{4} - \frac{(t-1)^2}{4}$$

Performing the Cholesky factorization $P = D^T D$, (A.39) can be represented by

$$\|Dx\|_2^2 + \frac{(t+1)^2}{4} \leq \frac{(t-1)^2}{4}$$

So (A.39) is equivalent to the following second-order cone constraint

$$\left\| \begin{bmatrix} 2Dx \\ q^T x + r + 1 \end{bmatrix} \right\|_2 \leq q^T x + r - 1 \tag{A.40}$$

However, not every second-order cone constraint can be expressed via a convex quadratic constraint. By squaring $\|Dx - d\|_2 \leq p^T x - q$ we get an equivalent quadratic inequality

$$x^T(D^T D - pp^T)x + 2(qp^T - d^T D)x + d^T d - q^2 \leq 0 \quad (\text{A.41})$$

with $p^T x - q \geq 0$. The matrix $M = D^T D - pp^T$ is not always positive semidefinite. Indeed, $M \succeq 0$ if and only if $\exists u, \|u\|_2 \leq 1 : p = D^T u$. On this account, SOCPs are more general than convex QCQPs.

b. Hyperbolic constraints

Hyperbolic constraints are frequently encountered in engineering optimization problems. They are non-convex in their original forms but can be represented by a second-order cone constraint. A hyperbolic constraint has the form

$$x^T x \leq yz, \quad y > 0, \quad z > 0 \quad (\text{A.42})$$

where $x \in \mathbb{R}^n$, $y, z \in \mathbb{R}_{++}$. Noticing the fact that $4yz = (y + z)^2 - (y - z)^2$, (A.42) is equivalent to the following second-order cone constraint

$$\left\| \begin{bmatrix} 2x \\ y - z \end{bmatrix} \right\|_2 \leq y + z, \quad y > 0, \quad z > 0 \quad (\text{A.43})$$

However, a hyperbolic constraint can not be expressed via a convex quadratic constraint, because the compact quadratic form of (A.42) is

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix}^T P \begin{bmatrix} x \\ y \\ z \end{bmatrix} \leq 0, \quad P = \begin{bmatrix} 2I & 0 & 0 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

where the matrix P is indefinite.

Many instances can be regarded as special cases of hyperbolic constraints, such as the upper branch of hyperbola

$$\{(x, y) \mid xy \geq 1, \quad x > 0\}$$

and the epigraph of a fractional-quadratic function $g(x, s) = x^T x / s$, $s > 0$

$$\left\{ (x, s, t) \mid t \geq \frac{x^T x}{s}, \quad s > 0 \right\}$$

c. Composition of second-order cone representable functions

A function is called second-order cone representable if its epigraph can be represented by second-order cone constraints. Second-order cone representable functions are closed under composition [10]. Suppose two univariate convex functions $f_1(x)$ and $f_2(x)$ are second-order cone representable, and

$f_1(x)$ is monotonically increasing, the composition $g(x) = f_1(f_2(x))$ is also second-order cone representable, because its epigraph $\{(x, t) \mid g(x) \leq t\}$ can be expressed by

$$\{(x, t) \mid \exists s : f_1(s) \leq t, f_2(x) \leq s\}$$

where $f_1(s) \leq t$ and $f_2(x) \leq s$ essentially come down to second-order cone constraints.

d. Maximizing the production of concave functions

Suppose two functions $f_1(x)$ and $f_2(x)$ are concave with $f_1(x) \geq 0$, $f_2(x) \geq 0$, and $-f_1(x)$ and $-f_2(x)$ are second-order cone representable [which means $f_1(x) \geq t_1$ and $f_2(x) \geq t_2$ are (equivalent to) second-order cone constraints]. Consider the maximum of their production

$$\max_x \{f_1(x)f_2(x) \mid x \in X, f_1(x) \geq 0, f_2(x) \geq 0\} \quad (\text{A.44})$$

where the feasible region X is the intersection of a polyhedron and second-order cones. It is not instantly clear whether problem (A.44) is a convex optimization problem or not. This formulation frequently arises in engineering applications, such as the Nash Bargaining problem and multi-objective optimization problems. An example can be found in Sect. ??.

By introducing auxiliary variables t, t_1, t_2 , it is immediately seen that problem (A.44) is equivalent to the following SOCP

$$\begin{aligned} \max_{x, t, t_1, t_2} \quad & t \\ \text{s.t.} \quad & x \in X \\ & t_1 \geq 0, t_2 \geq 0, t_1 t_2 \geq t^2 \\ & f_1(x) \geq t_1, f_2(x) \geq t_2 \end{aligned} \quad (\text{A.45})$$

At the optimal solution, $f_1(x)f_2(x) = t^2$.

3. Polyhedral approximation of second-order cones

Although SOCPs can be solved very efficiently, the state-of-the-art in numerical computing of SOCPs is still incomparable to that in LPs. The salient computational superiority of LPs inspires a question: can we approximate an SOCP by an LP without dramatically increasing the problem size? There have been other reasons to explore LP approximations for SOCPs. For example, to solve a bilevel program with an SOCP lower level, the SOCP should be replaced by its optimality conditions. However, the primal-dual optimality condition (A.32) may introduce bilinear terms, while the second-order cone complementarity constraints in KKT optimality condition (A.33) cannot be linearized easily. If the SOCP can be approximated by an LP, then the KKT optimality condition can be linearized and the original bilevel program can be reformulated as an MILP. Clearly, if we only work in original variables,

the number of additional constraints would quickly grow unacceptable with the increasing problem dimension and required accuracy. In this section, we introduce the technique developed in [11], which lifts the problem into higher dimensions with moderate numbers of auxiliary variables and constraints.

We start with the basic question: find a polyhedral ϵ -approximation Π for \mathbb{L}_C^3 such that:

- 1) If $x \in \mathbb{L}_C^3$, then $\exists u : (x, u) \in \Pi$.
- 2) If $(x, u) \in \Pi$ for some u , then $\sqrt{x_1^2 + x_2^2} \leq (1 + \epsilon)x_3$.

Geometrically, the polyhedral cone Π includes a system of homogeneous linear equalities and inequalities in variables x, u ; its projection on x -space is an ϵ -outer approximation of \mathbb{L}_C^3 , and the error bound is quantified by ϵx_3 . The answer to this question is given in [11]. It is shown that Π can be expressed by

$$\begin{aligned}
 (a) \quad & \begin{cases} \xi^0 \geq |x_1| \\ \eta^0 \geq |x_2| \end{cases} \\
 (b) \quad & \begin{cases} \xi^j = \cos\left(\frac{\pi}{2^{j+1}}\right) \xi^{j-1} + \sin\left(\frac{\pi}{2^{j+1}}\right) \eta^{j-1} \\ \eta^j \geq \left| -\sin\left(\frac{\pi}{2^{j+1}}\right) \xi^{j-1} + \cos\left(\frac{\pi}{2^{j+1}}\right) \eta^{j-1} \right| \end{cases}, \quad j = 1, \dots, v \\
 (c) \quad & \begin{cases} \xi^v \leq x_3 \\ \eta^v \leq \tan\left(\frac{\pi}{2^{v+1}}\right) \xi^v \end{cases}
 \end{aligned} \tag{A.46}$$

Formulation (A.46) can be understood from an geometric point of view:

- 1) Given $x \in \mathbb{L}_C^3$, set $\xi^0 = |x_1|$, $\eta^0 = |x_2|$, which satisfies (a) in (A.46), and point $P^0 = (\xi^0, \eta^0)$ belongs to the first quadrant. Let

$$\begin{cases} \xi^j = \cos\left(\frac{\pi}{2^{j+1}}\right) \xi^{j-1} + \sin\left(\frac{\pi}{2^{j+1}}\right) \eta^{j-1} \\ \eta^j = \left| -\sin\left(\frac{\pi}{2^{j+1}}\right) \xi^{j-1} + \cos\left(\frac{\pi}{2^{j+1}}\right) \eta^{j-1} \right| \end{cases}$$

which ensures (b). Point $P^j = (\xi^j, \eta^j)$ is obtained from P^{j-1} according to following operation: rotate P^{j-1} by angle $\phi_j = \pi/2^{j+1}$ clockwise and get a mediate point Q^{j-1} ; if Q^{j-1} resides in the upper half-plane, $P^j = Q^{j-1}$; otherwise P^j is the reflection of Q^{j-1} with respect to the x -axis. By this construction, it is clear that all vectors from the origin to P^j have the same Euclidean norm, i.e., $\|[x_1, x_2]\|_2$. Moreover, as P^0 belongs to the first quadrant, the angle of Q^0 must satisfy $-\pi/4 \leq \arg(Q^0) \leq \pi/4$, and $0 \leq \arg(P^1) \leq \pi/4$. With the procedure going on, we have $|\arg(Q^j)| \leq \pi/2^{j+1}$, and $0 \leq \arg(P^{j+1}) \leq \pi/2^{j+1}$, for $j = 1, \dots, v$. In the last step, $\xi^v \leq \|P^v\|_2 = \|[x_1, x_2]\|_2 \leq x_3$ and $0 \leq \arg(P^v) \leq \pi/2^{v+1}$ hold, ensuring condition (c). In this manner, a point in \mathbb{L}_C^3 has been extended to a solution of (A.46).

2) Given $(x, u) \in \Pi$, where $u = \{\xi^j, \eta^j\}, j = 1, \dots, v$. Define $P^j = [\xi^j, \eta^j]$, and it directly follows from (a) and (b) that all P^j belongs to the first quadrant, and $\|P^0\|_2 \geq \sqrt{x_1^2 + x_2^2}$. Moreover, recall the construction of Q^j in previous analysis, it is seen $\|P^j\|_2 = \|Q^j\|_2$; the absolute value of the vertical coordinate of P^{j+1} is no less than that of Q^j ; therefore, $\|P^{j+1}\|_2 \geq \|Q^j\|_2 = \|P^j\|_2$. At last

$$\|P^v\|_2 \leq \frac{x_3}{\cos\left(\frac{\pi}{2^{v+1}}\right)}$$

so we arrive at $\sqrt{x_1^2 + x_2^2} \leq (1 + \epsilon)x_3$, where

$$\epsilon = \frac{1}{\cos\left(\frac{\pi}{2^{v+1}}\right)} - 1 \quad (\text{A.47})$$

In this way, a solution of (A.46) has been approximately extended to \mathbb{L}_C^3 .

Now, let us consider the general case: approximating

$$\mathbb{L}_C^{n+1} = \left\{ (y, t) \mid \sqrt{y_1^2 + \dots + y_n^2} \leq t \right\}$$

via a polyhedral cone. Without loss of generality, we assume $n = 2^K$. To make use of the outcome in (A.46), y is split into 2^{K-1} pairs $(y_1, y_2), \dots, (y_{n-1}, y_n)$, which are called variables of generation 0. A successor variable is associated with each pair, which is called variable of generation 1, and is further divided into 2^{K-2} pairs and associated with variable of generation 2, and so on. After $K - 1$ steps of dichotomy, we complete variable splitting with two variables of generation $K - 1$. The only variable of generation K is t . For notation convenience, let y_i^l be i -th variable of generation l , the original vector $y = [y_1^0, \dots, y_n^0]$, and $t = y_1^K$. The “parents” of y_i^l are variables $y_{2i-1}^{l-1}, y_{2i}^{l-1}$. The total number of variables in the “tower” is $2n - 1$.

Using the tower of variables $y^l, \forall l$, the system of constraints

$$\sqrt{(y_{2i-1}^{l-1})^2 + (y_{2i}^{l-1})^2} \leq y_i^l, \quad i = 1, \dots, 2^{K-l}, \quad l = 1, \dots, K \quad (\text{A.48})$$

gives the same feasible region on y as L_C^{n+1} , and each second-order cone in \mathbb{L}_C^3 in (A.48) can be approximated by a polyhedral cone given in (A.46). To see the first claim, a simple example is given in Sect. ??.

The size of this polyhedral approximation is unveiled in [11]:

- 1) The dimension of the lifted variable is $p \leq n + O(1) \sum_{l=1}^K 2^{K-l} v_l$.
- 2) The number of constraints is $q \leq O(1) \sum_{l=1}^K 2^{K-l} v_l$.

The quality of the approximation is [11]

$$\beta = \prod_{l=1}^K \frac{1}{\cos\left(\frac{\pi}{2^{v_l+1}}\right)} - 1$$

Given a desired tolerance ϵ , choose

$$v_l = \lfloor O(1)l \ln \frac{2}{\epsilon} \rfloor$$

with a proper constant $O(1)$, we can guarantee the following bounds:

$$\begin{aligned} \beta &\leq \epsilon \\ p &\leq O(1)n \ln \frac{2}{\epsilon} \\ q &\leq O(1)n \ln \frac{2}{\epsilon} \end{aligned}$$

which implies that the required numbers of variables and constraints grow linearly in the dimension of the target second-order cone.

A.2.4 Semidefinite Program

1. Notation clarification

In this section, variables appear in the form of symmetric matrices, some notations should be clarified first.

The Frobenius inner product of two matrices $A, B \in \mathbb{M}^n$ is defined by

$$\langle A, B \rangle = \text{tr}(AB^T) = \sum_{i=1}^n \sum_{j=1}^n A_{ij}B_{ij} \quad (\text{A.49})$$

The Euclidean norm of a matrix $X \in \mathbb{M}^n$ can be defined through the Frobenius inner product as follows

$$\|X\|_2 = \sqrt{\langle X, X \rangle} = \sqrt{\text{tr}(X^T X)} \quad (\text{A.50})$$

Equipped with the Frobenius inner product, the dual cone of a given cone $K \subset \mathbb{S}^n$ is defined by

$$K^* = \{Y \in \mathbb{S}^n \mid \langle Y, X \rangle \geq 0, \forall X \in K\} \quad (\text{A.51})$$

Among the cones in \mathbb{S}^n , this section talks about the positive semidefinite cone \mathbb{S}_+^n . As what has been demonstrated in Sect. A.1.3, \mathbb{S}_+^n is self-dual, i.e., $(\mathbb{S}_+^n)^* = \mathbb{S}_+^n$. The interior of cone \mathbb{S}_+^n consists of all $n \times n$ matrices that are positive definite, and is denoted by \mathbb{S}_{++}^n .

2. Primal and dual formulations of SDPs

When $K = \mathbb{S}_+^n$, conic program (A.28) boils down to an SDP

$$\min_x \{c^T x \mid Ax - b \in \mathbb{S}_+^n\}$$

which minimizes a linear objective over the intersection of affine plane $y = Ax - b$ and the positive semidefinite cone \mathbb{S}_+^n . However, the notation in such a form is a little confusing: $Ax - b$ is a vector, which is not dimensionally compatible with the cone \mathbb{S}_+^n . In fact, we have met a similar difficulty at the very beginning: the vector inner product does not apply to matrices, which is consequently replaced with the Frobenius inner product. There are two prevalent ways to resolve the confliction in dimension, leading to different formulations which will be discussed.

a. Formulation based on vector decision variables

In this formulation, b is replaced with a matrix $B \in \mathbb{S}^n$, and Ax is replaced with a linear mapping $\mathcal{A}x : \mathbb{R}^n \rightarrow \mathbb{S}^n$. In this way, $\mathcal{A}x - B$ becomes an element of \mathbb{S}^n . A simple way to specify the linear mapping $\mathcal{A}x$ is

$$\mathcal{A}x = \sum_{j=1}^n x_j A_j, \quad x = [x_1, \dots, x_n]^T, \quad A_1, \dots, A_n \in \mathbb{S}^n$$

With all these input matrices, an SDP can be written as

$$\min_x \{c^T x \mid x_1 A_1 + \dots + x_n A_n - B \succeq 0\} \quad (\text{A.52})$$

where the cone \mathbb{S}_+^n is omitted in the operator \succeq without causing confusion. The constraint in (A.52) is an LMI. This formulation is general enough to capture the situation in which multiple LMIs exist, because

$$\mathcal{A}_i x - B_i \succeq 0, \quad i = 1, \dots, k \Leftrightarrow \mathcal{A}x - B \succeq 0$$

with $\mathcal{A}x = \text{Diag}(\mathcal{A}_1 x, \dots, \mathcal{A}_k x)$ and $B = \text{Diag}(B_1, \dots, B_k)$.

The general form of conic duality can be specified in the case when the cone $K = \mathbb{S}_+^n$. Associating a matrix dual variable Λ with the LMI constraint, and recalling the fact that $(\mathbb{S}_+^n)^* = \mathbb{S}_+^n$, the dual program of SDP (A.52) reads:

$$\max_{\Lambda} \{ \langle B, \Lambda \rangle \mid \langle A_i, \Lambda \rangle = c_i, \quad i = 1, \dots, n, \quad \Lambda \succeq 0 \} \quad (\text{A.53})$$

which remains an SDP.

Apply conic duality theorem given in Proposition A.5 to SDPs (A.52) and (A.53).

1) Suppose A_1, \dots, A_n are linearly independent i.e., no nontrivial linear combination of A_1, \dots, A_n gives an all zero matrix.

2) The primal SDP (A.52) is strict feasible, i.e., $\exists x : x_1 A_1 + \cdots + x_n A_n \succ B$, and is solvable (the minimum is attainable)

3) The dual SDP (A.53) is strict feasible, i.e., $\exists \Lambda \succ 0 : \langle A_i, \Lambda \rangle = c_i, i = 1, \dots, n$, and is solvable (the maximum is attainable).

The optimal values of (A.52) and (A.53) are equal, and the complementarity and slackness condition

$$\langle \Lambda, x_1 A_1 + \cdots + x_n A_n - B \rangle = 0 \quad (\text{A.54})$$

is necessary and sufficient for a pair of primal and dual feasible solutions (x, Λ) to be optimal for their respective problems. For a pair of positive semidefinite matrices, it can be shown that

$$\langle XY \rangle = 0 \Leftrightarrow XY = YX = 0$$

indicating that the eigenvalues of these two matrices in some certain basis are complementary: for every common eigenvector, at most one of the two eigenvalues of X and Y can be strictly positive.

b. Formulation based on matrix decision variables

This formulation directly incorporates a matrix decision variable $X \in \mathbb{S}_+^n$, and imposes other restrictions on X through linear equations. In the objective function, the vector inner product $c^T x$ is replaced by a Frobenius inner product $\langle C, X \rangle$. In this way, an SDP can be written as

$$\begin{aligned} \min_X \quad & \langle C, X \rangle \\ \text{s.t.} \quad & \langle A_i, X \rangle = b_i : \lambda_i, \quad i = 1, \dots, m \\ & X \succeq 0 : \Lambda \end{aligned} \quad (\text{A.55})$$

By introducing dual variables (following the colon) for individual constraints, the dual program of (A.55) can be constructed as

$$\begin{aligned} \max_{\lambda, \Lambda} \quad & b^T \lambda + \langle 0, \Lambda \rangle \\ \text{s.t.} \quad & \Lambda + \lambda_1 A_1 + \cdots + \lambda_n A_n = C \\ & \Lambda \succeq 0 \end{aligned}$$

Eliminating Λ , we obtain

$$\begin{aligned} \max_{\lambda} \quad & b^T \lambda \\ \text{s.t.} \quad & C - \lambda_1 A_1 - \cdots - \lambda_n A_n \succeq 0 \end{aligned} \quad (\text{A.56})$$

It is observed that (A.55) and (A.56) are in the same form compared with (A.53) and (A.52), respectively, except for the signs of some coefficients.

SDP handles positive semidefinite matrices, so it is especially powerful in eigenvalue related problems, such as Lyapunov stability analysis and controller design, which are the main field of control theorists. Moreover, every SOCP can be formulated as an SDP because

$$\|y\|_2 \leq t \Leftrightarrow \begin{bmatrix} tI & y \\ y^T & t \end{bmatrix} \succeq 0$$

Nevertheless, solving SOCPs via SDP may not be a good idea. Interior-point algorithms for SOCPs have much better worst-case complexity than those for SDPs. In fact, SDPs are extremely popular in the convex relaxation technique for non-convex quadratic optimization problems, owing to its ability to offer a nearly global optimal solution in many practical applications, such as the OPF problem in power systems. The SDP based convex relaxation method for non-convex QCQPs will be discussed in the next section. Here we talk about some special cases involving homogeneous quadratic functions or at most two non-homogeneous quadratic functions.

3. Homogeneous quadratic programs

Consider the following quadratic program

$$\begin{aligned} \min \quad & x^T B x \\ \text{s.t.} \quad & x^T A_i x \geq 0, \quad i = 1, \dots, m \end{aligned} \tag{A.57}$$

where $A_1, \dots, A_m, B \in \mathbb{S}^n$ are constant coefficients. Suppose that problem (A.57) is feasible. Due to its homogeneity, the optimal value is clear: $-\infty$ or 0, depending on whether there is a feasible solution x such that $x^T B x < 0$ or not. But it is unclear which situation takes place, i.e., to judge $x^T B x \geq 0$ over the intersection of homogeneous inequalities $x^T A_i x \geq 0, i = 1, \dots, m$, or whether the implication

$$x^T A_i x \geq 0, \quad i = 1, \dots, m \Rightarrow x^T B x \geq 0 \tag{A.58}$$

holds.

Proposition A.6. *If there exist $\lambda_i \geq 0, i = 1, 2, \dots$ such that $B \succeq \sum_i \lambda_i A_i$, then the indication in (A.58) is true.*

To see this, $B \succeq \sum_i \lambda_i A_i \Leftrightarrow x^T (B - \sum_i \lambda_i A_i) x \geq 0 \Leftrightarrow x^T B x \geq \sum_i \lambda_i x^T A_i x$; therefore, $x^T B x$ is a direct consequence of $x^T A_i x \geq 0, i = 1, \dots, m$, as the right-hand side of the last inequality is non-negative. Proposition A.6 provides a sufficient condition for (A.58), and necessity is generally not guaranteed. Nevertheless, if $m = 1$, the condition is both necessary and sufficient.

Proposition A.7. (*S-Lemma*) *Let $A, B \in \mathbb{S}^n$ and a homogeneous quadratic inequality*

$$(a) \quad x^T A x \geq 0$$

is strictly feasible. Then the homogeneous quadratic inequality

$$(b) \quad x^T B x \geq 0$$

is a consequence of (a) if and only if $\exists \lambda \geq 0 : B \succeq \lambda A$.

Proposition A.7 is called the S-Lemma or S-Procedure. It can be proved by many means. The most instructive one, in our tastes, is based on the semidefinite relaxation, which can be found in [1].

4. Non-homogeneous quadratic programs with a single constraint

Consider the following quadratic program

$$\begin{aligned} \min \quad & f_0(x) = x^T A_0 x + 2b_0^T x + c_0 \\ \text{s.t.} \quad & f_1(x) = x^T A_1 x + 2b_1^T x + c_1 \leq 0 \end{aligned} \quad (\text{A.59})$$

Let f^* denote the optimal solution, so $f_0(x) - f^* \geq 0$ is a consequence of $-f_1(x) \geq 0$. A sufficient condition for this implication is $\exists \lambda \geq 0 : f_0(x) - f^* + \lambda f_1(x) \geq 0$. The left-hand side is a quadratic function with matrix form

$$\begin{bmatrix} x \\ 1 \end{bmatrix}^T \begin{bmatrix} A_0 + \lambda A_1 & b_0 + \lambda b_1 \\ (b_0 + \lambda b_1)^T & c_0 + \lambda c_1 - f^* \end{bmatrix} \begin{bmatrix} x \\ 1 \end{bmatrix}$$

Its non-negativeness is equivalent to

$$\begin{bmatrix} A_0 + \lambda A_1 & b_0 + \lambda b_1 \\ (b_0 + \lambda b_1)^T & c_0 + \lambda c_1 - f^* \end{bmatrix} \succeq 0 \quad (\text{A.60})$$

Similar to the homogeneous case, this condition is also sufficient. In view of this, the optimal value f^* of (A.59) solves the following SDP

$$\begin{aligned} \min_{\lambda, f} \quad & f \\ \text{s.t.} \quad & \begin{bmatrix} A_0 + \lambda A_1 & b_0 + \lambda b_1 \\ (b_0 + \lambda b_1)^T & c_0 + \lambda c_1 - f \end{bmatrix} \succeq 0 \end{aligned} \quad (\text{A.61})$$

This conclusion is known as the non-homogeneous S-Lemma:

Proposition A.8. (Non-homogeneous S-Lemma) Let $A_i \in \mathbb{S}^n$, $b_i \in \mathbb{R}^n$, and $c_i \in \mathbb{R}$, $i = 0, 1$, if $\exists x : x^T A_1 x + 2b_1^T x + c_1 < 0$, the implication

$$x^T A_1 x + 2b_1^T x + c_1 \leq 0 \Rightarrow x^T A_0 x + 2b_0^T x + c_0 \leq 0$$

holds if and only if

$$\exists \lambda \geq 0 : \begin{bmatrix} A_0 & b_0 \\ b_0^T & c_0 \end{bmatrix} \preceq \lambda \begin{bmatrix} A_1 & b_1 \\ b_1^T & c_1 \end{bmatrix} \quad (\text{A.62})$$

Because the implication can boil down to the maximum of quadratic function $x^T A_0 x + 2b_0^T x + c_0$ being non-positive over set $\{x | x^T A_1 x + 2b_1^T x + c_1 \leq 0\}$, which is a special case of (A.59) by letting $f_0(x) = -x^T A_0 x - 2b_0^T x - c_0$, Proposition A.8 is a particular case of (A.61) with the optimum $f^* = 0$.

A formal proof based on semidefinite relaxation is given in [1]. Since a quadratic inequality describes an ellipsoid, Proposition A.8 can be used to test whether an ellipsoid is contained in another one.

As a short conclusion, we summarize the relation of discussed convex programs in Fig. A.8.

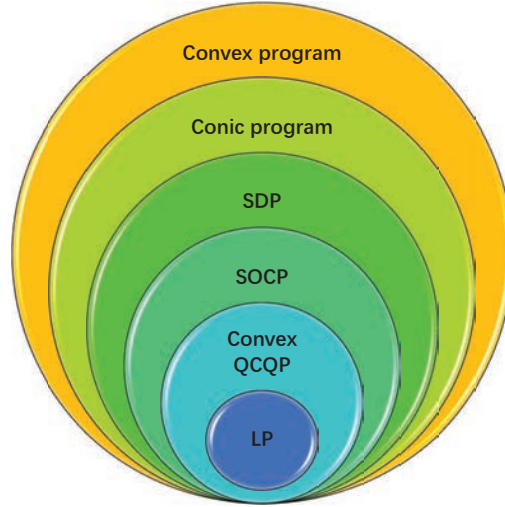


Fig. A.8 Relations of the discussed convex programs.

A.3 Convex Relaxation Methods for Non-convex QCQPs

One of the most prevalent and promising applications of SDP is to build tractable approximations of computationally intractable optimization problems. One of the most quintessential appliances is the convex relaxation of quadratically constrained quadratic programs (QCQPs), which cover vast engineering optimization problems. QCQPs are generally non-convex and could

have more than one locally optimal solution, and each of them may yield significant different objective values. However, gradient based algorithms can only find a local solution which largely depends on the initial point. One primary interest is to identify the global optimal solution or determine a high-quality bound for the optimum, which can be used to quantify the optimality gap of a given local optimal solution. The SDP relaxation technique for solving non-convex QCQPs are briefly reviewed in this section.

A.3.1 SDP Relaxation and Valid Inequalities

A standard fact of quadratic expression is

$$x^T Q x = \langle Q, x x^T \rangle \quad (\text{A.63})$$

where $\langle \cdot \rangle$ stands for the Frobenius inner product.

Following the logic in [13], we focus our attention on QCQPs in the following form

$$\min \{x^T C x + c^T x \mid x \in F\} \quad (\text{A.64})$$

where

$$F = \{x \in \mathbb{R}^n \mid x^T A_k x + a_k^T x \leq b_k, \ k = 1, \dots, m, \ l \leq x \leq u\} \quad (\text{A.65})$$

All coefficient matrices and vectors have compatible dimensions. If $A_k = 0$ in all constraints, then the feasible set F is a polyhedron, and (A.64) reduces to a quadratic program (QP); If $A_k \succeq 0$, $k = 1, \dots, m$ and $C \succeq 0$, (A.64) is a convex QCQP, which is easy to solve. Without loss of generality, we assume A_k , $k = 1, \dots, m$ and C are indefinite, F is a non-convex set, and the objective is a non-convex function. In fact, a number of hard optimization problems can be cast as non-convex QCQP (A.64). For example, a polynomial optimization problem can be reduced to a QCQP by introducing a tower of condensing variables, e.g., $x_1 x_2 x_3 x_4$ could be replaced by quadratic term $x_{12} x_{34}$ with $x_{12} = x_1 x_2$ and $x_{34} = x_3 x_4$. Moreover, a binary constraint $x \in \{0, 1\}$ is equivalent to quadratic equality $x(x - 1) = 0$ where x is continuous.

A common idea to linearize non-convex terms $x^T A_k x$ is to define new variables $X_{ij} = x_i x_j$, $i = 1, \dots, n$, $j = 1, \dots, n$. In this way, $x^T A_k x = \sum_i \sum_j A_{ij} x_i x_j = \sum_{ij} A_{ij} X_{ij}$, and the last term is linear. Recall (A.63), this fact can be written in a compact form

$$x^T A_k x = \langle A_k, X \rangle, \ X = x x^T$$

With this transformation, QCQP (A.64) becomes

$$\min \{ \langle C, X \rangle + c^T x \mid (x, X) \in \hat{F} \} \quad (\text{A.66})$$

where

$$\hat{F} = \left\{ (x, X) \in \mathbb{R}^n \times \mathbb{S}^n \mid \begin{array}{l} \langle A_k, X \rangle + a_k^T x \leq b_k, \quad k = 1, \dots, m \\ l \leq x \leq u, \quad X = xx^T \end{array} \right\} \quad (\text{A.67})$$

In problem (A.66), non-convexity are concentrated in the relation between the lifting variable X and the original variable x , whereas all other constraints are linear. Moreover, if we replace \hat{F} with its convex hull $\text{conv}(\hat{F})$, the optimal solution of (A.66) will not change, because its objective function is linear. However, $\text{conv}(\hat{F})$ does not have a closed form expression. Convex relaxation approaches can be interpreted as attempting to approximate $\text{conv}(\hat{F})$ through structured convex constraints which can be recognized by existing solvers.

We define the following linear relaxation

$$\hat{L} = \left\{ (x, X) \in \mathbb{R}^n \times \mathbb{S}^n \mid \begin{array}{l} \langle A_k, X \rangle + a_k^T x \leq b_k \\ k = 1, \dots, m \\ l \leq x \leq u \end{array} \right\} \quad (\text{A.68})$$

which contains only linear constraints. Now, let us consider the lifting constraint

$$X = xx^T \quad (\text{A.69})$$

which is called a rank-1 constraint. However, a rank constraint is non-convex and cannot be accepted by most solvers. Notice the fact that if (A.69) holds, then

$$\begin{bmatrix} 1 & x^T \\ x & X \end{bmatrix} = \begin{bmatrix} 1 & x^T \\ x & xx^T \end{bmatrix} = \begin{bmatrix} 1 \\ x \end{bmatrix} \begin{bmatrix} 1 \\ x \end{bmatrix}^T \succeq 0$$

Define an LMI constraint

$$\text{LMI} = \left\{ (x, X) \mid Y = \begin{bmatrix} 1 & x^T \\ x & X \end{bmatrix} \succeq 0 \right\} \quad (\text{A.70})$$

The positive semi-definiteness condition is true over $\text{conv}(\hat{F})$.

The basic SDP relaxation of (A.66) replaces the rank-1 constraint in \hat{F} with a weaker but convex constraint (A.70), giving rise to the following SDP

$$\begin{aligned} \min \quad & \langle C, X \rangle + c^T x \\ \text{s.t.} \quad & (x, X) \in \hat{L} \cap \text{LMI} \end{aligned} \quad (\text{A.71})$$

Clearly, the LMI constraint enlarges the feasible region defined by (A.69), so the optimal solution to (A.71) may not be feasible in the original QCQP, and the optimal value is a strict lower bound. In this situation, the SDP relaxation is inexact. Conversely, if matrix Y is indeed rank-1 at the optimal solution, then the SDP relaxation is exact and x solves the original QCQP (A.64).

The basic SDP relaxation model (A.71) can be further improved by enforcing additional linkages between x and X , which are called valid inequalities. Suppose linear inequalities $\alpha^T x \leq \alpha_0$ and $\beta^T x \leq \beta_0$ are chosen from \hat{L} , then the quadratic inequality

$$(\alpha_0 - \alpha^T x)(\beta_0 - \beta^T x) = \alpha_0\beta_0 - \alpha_0\beta^T x - \beta_0\alpha^T x + x^T\alpha\beta^T x \geq 0$$

holds for all $x \in \hat{L}$. The last quadratic term can be linearized via the lifting variable X , resulting in the following linear inequality

$$\alpha_0\beta_0 - \alpha_0\beta^T x - \beta_0\alpha^T x + \langle \beta\alpha^T, X \rangle \geq 0 \quad (\text{A.72})$$

Any linear inequality in \hat{L} (possibly the same) can be used to construct valid inequalities. Because additional constraints are imposed on X , the relaxation could be tightened, and the feasible region shrinks but may still be larger than $\text{conv}(\hat{F})$.

If we construct valid inequality (A.72) from side constraint $l \leq x \leq u$, we get

$$\left. \begin{aligned} (x_i - l_i)(x_j - l_j) &\geq 0 \\ (x_i - l_i)(u_j - x_j) &\geq 0 \\ (u_i - x_i)(x_j - l_j) &\geq 0 \\ (u_i - x_i)(u_j - x_j) &\geq 0 \end{aligned} \right\}, \quad \forall i, j = 1, \dots, n, \quad i \leq j \quad (\text{A.73})$$

Expanding these quadratic inequalities, the coefficients of quadratic terms $x_i x_j$ are equal to 1, and we obtain simple bounds on X_{ij}

$$\begin{aligned} x_i l_j + x_j l_i - l_i l_j &\leq X_{ij} \\ x_i u_j + x_j l_i - l_i u_j &\geq X_{ij} \\ u_i x_j - u_i l_j + x_i l_j &\geq X_{ij} \\ u_i x_j + x_i u_j - u_i u_j &\leq X_{ij} \end{aligned}$$

or in a compact matrix form [13]

$$\text{RLT} = \left\{ (x, X) \left| \begin{array}{l} lx^T + xl^T - ll^T \leq X \\ ux^T + xu^T - uu^T \leq X \\ xu^T + lx^T - lu^T \geq X \end{array} \right. \right\} \quad (\text{A.74})$$

(A.74) is known as the reformulation-linearization technique after the term appeared in [14]. These constraints have been extensively studied since it was proposed in [15], due to the simple structure and satisfactory performance in various applications. The improved SDP relaxation with valid inequalities can be written as

$$\begin{aligned} \min \quad & \langle C, X \rangle + c^T x \\ \text{s.t.} \quad & (x, X) \in \hat{L} \cap \text{LMI} \cap \text{RLT} \end{aligned} \quad (\text{A.75})$$

From the construction of \hat{L} , LMI, and RLT, it is directly concluded that

$$\text{conv}(\hat{F}) \subseteq \hat{L} \cap \text{LMI} \cap \text{RLT} \quad (\text{A.76})$$

The inclusion becomes tight only in some very special situations, such as those encountered in the homogeneous and non-homogeneous S-Lemma. Nevertheless, what we really need is the equivalence between the optimal solution of the relaxed problem (A.75) and that of the original problem (A.64): if the optimal matrix variable of (A.75) allows a rank-1 decomposition

$$\begin{bmatrix} 1 & x^T \\ x & X \end{bmatrix} = \begin{bmatrix} 1 \\ x \end{bmatrix} \begin{bmatrix} 1 \\ x \end{bmatrix}^T$$

which indicates X has a rank-1 decomposition $X = xx^T$, then x is optimal in (A.64), and the SDP relaxation is said to be exact, although $\text{conv}(\hat{F})$ may be a strict subset of $\hat{L} \cap \text{LMI} \cap \text{RLT}$.

A.3.2 Successively Tightening the Relaxation

If the matrix X has a rank higher than 1, the corresponding optimal solution x in (A.75) may be infeasible in (A.64). The rank-1 constraint on X can be exactly described by a pair of LMIs $X \succeq xx^T$ and $X \preceq xx^T$. The former one is redundant to (A.70) indicated by the Schur complement theorem; the latter one is non-convex, which is simply neglected in the SDP relaxation.

1. A dynamical valid inequality generation approach

An approach is proposed in [13] to generate valid inequalities dynamically by harnessing the constraint violations in $X \preceq xx^T$. The motivation comes from the fact that

$$X - xx^T \preceq 0 \Leftrightarrow \langle X, v_i v_i^T \rangle \leq (v_i^T x)^2, \quad i = 1, \dots, n$$

where $\{v_1, \dots, v_n\}$ is a set of orthogonal basis of \mathbb{R}^n . To see this, any vector $h \in \mathbb{R}^n$ can be expressed as the linear combination of the orthogonal basis as $h = \sum_{i=1}^n \lambda_i v_i$, therefore, $h^T (X - xx^T) h = \langle X, hh^T \rangle - (h^T x)^2 = \sum_{i=1}^n \lambda_i^2 [\langle X, v_i v_i^T \rangle - (v_i^T x)^2] \leq 0$. In view of this,

$$\hat{F} = \hat{L} \cap \text{LMI} \cap \text{NSD}$$

where

$$\begin{aligned} \text{NSD} &= \{(x, X) \mid X - xx^T \preceq 0\} \\ &= \{(x, X) \mid \langle X, v_i v_i^T \rangle \leq (v_i^T x)^2, i = 1, \dots, n\} \end{aligned}$$

If $\{v_1, \dots, v_n\}$ is the standard orthogonal basis,

$$\text{NSD} = \{(x, X) \mid X_{ii} \leq x_i^2, i = 1, \dots, n\} \quad (\text{A.77})$$

It is proposed in [13] to construct NSD as

$$\text{NSD} = \{(x, X) \mid \langle X, \eta_i \eta_i^T \rangle \leq (\eta_i^T x)^2, i = 1, \dots, n\} \quad (\text{A.78})$$

where $\{\eta_1, \dots, \eta_n\}$ are the eigenvectors of matrix $X - xx^T$, because they exclude infeasible points with respect to $X - xx^T \preceq 0$ most effectively.

Non-convex constraints in (A.77) and (A.78) can be handled by a special disjunctive programming derived in [16] and the convex-concave procedure investigated in [17]. The former one is an exact approach which requires binary variables to formulate disjunctive constraints; the latter is a heuristic approach which only solves convex optimization problems. We do not further detail these techniques here.

2. A rank penalty method [18]

In view of the rank-1 exactness condition, another way to tighten SDP relaxation is to work on the rank of the optimal solution. A successive rank penalty approach is proposed in [18]. We consider problem (A.66) as a rank-constrained SDP

$$\min \{ \langle \Omega, Y \rangle \mid Y \in \hat{L} \cap \text{LMI} \cap \text{RLT}, \text{rank}(Y) = 1 \} \quad (\text{A.79})$$

where

$$\Omega = \begin{bmatrix} 0 & 0.5c^T \\ 0.5c & C \end{bmatrix}, \quad Y = \begin{bmatrix} 1 & x^T \\ x & X \end{bmatrix}$$

constraints \hat{L} , LMI, and RLT (rearranged for variable Y) are defined in (A.68), (A.70), and (A.74), respectively. The last constraint in (A.79) ensures that Y has a rank-1 decomposition such that $X = xx^T$. Actually, LMI and RLT are redundant to the rank-1 constraint, but will give a high quality convex relaxation when the rank constraint is relaxed.

To treat the rank-1 constraint in a soft manner, we introduce a dummy variable Z , and penalize the matrix rank in the objective function, giving rising to the following problem

$$\begin{aligned} \min_Y \quad & \left\{ \langle \Omega, Y \rangle + \min_Z \frac{\rho}{2} \|Y - Z\|_2^2 \right\} \\ \text{s.t.} \quad & Y \in \hat{L} \cap \text{LMI} \cap \text{RLT}, \text{rank}(Z) = 1 \end{aligned} \quad (\text{A.80})$$

If the penalty parameter ρ is sufficiently large, the penalty term will be zero at the optimal solution, so $Y = Z$ and $\text{rank}(Z) = 1$. One advantage of this treatment is that the constraints on Y and Z are decoupled, and the inner rank minimization problem has a closed-form solution.

To see this, if $\text{rank}(Y) = k > 1$, the singular value decomposition of Y has the form $Y = U\Sigma V^T$, where

$$\Sigma = \text{diag}(S, 0), \quad S = \text{diag}(\sigma_1, \dots, \sigma_k), \quad \sigma_1 \geq \dots \geq \sigma_k > 0$$

U and V are orthogonal matrices. Let matrix D have the same dimension as Y , $D_{11} = \sigma_1$, and $D_{ij} = 0$, $\forall (i, j) \neq (1, 1)$, we have

$$\begin{aligned} & \min_Z \left\{ \frac{\rho}{2} \|Y - Z\|_2^2 \mid \text{rank}(Z) = 1 \right\} \\ &= \min_Z \left\{ \frac{\rho}{2} \|U(Y - Z)V^T\|_2^2 \mid \text{rank}(Z) = 1 \right\} \\ &= \min_Z \left\{ \frac{\rho}{2} \|\Sigma - UZV^T\|_2^2 \mid \text{rank}(Z) = 1 \right\} \\ &= \frac{\rho}{2} \|\Sigma - D\|_2^2 = \frac{\rho}{2} \sum_{i=2}^k \sigma_i^2 \\ &= \frac{\rho}{2} \|Y\|_2^2 - \frac{\rho}{2} \sigma_1^2(Y) \end{aligned}$$

To represent the latter term via a convex function, let matrix Θ have the same dimension as Y , $\Theta_{11} = 1$, and $\Theta_{ij} = 0$, $\forall (i, j) \neq (1, 1)$, we have

$$\text{tr}(Y^T U \Theta U^T Y) = \text{tr}(V \Sigma U^T U \Theta U^T U \Sigma V^T) = \text{tr}(V \Sigma \Theta \Sigma V^T) = \text{tr}(\Sigma \Theta \Sigma) = \sigma_1^2(Y)$$

Define two functions $f(Y) = \langle \Omega, Y \rangle + \frac{\rho}{2} \|Y\|_2^2$ and $g(Y) = \text{tr}(Y^T U \Theta U^T Y)$. Because $\|Y\|_2$ is convex in Y (Example 3.11, [2]), so is $\|Y\|_2^2$ (composition rule, page 84, [2]); clearly, $f(Y)$ is a convex function in Y , as it is the sum of a linear function and a convex function. For the latter one, the Hessian matrix of $g(Y)$ is

$$\nabla_Y^2 g(Y) = U \Theta U^T = U \Theta^T \Theta U^T = (\Theta U^T)^T \Theta U^T \succeq 0$$

so $g(Y)$ is also convex in Y . Substituting above results into problem (A.80), the rank constrained SDP (A.79) boils down to

$$\min_Y \left\{ \langle \Omega, Y \rangle + \frac{\rho}{2} \|Y\|_2^2 - \frac{\rho}{2} \text{tr}(Y^T U \Theta U^T Y) \mid Y \in \hat{L} \cap \text{LMI} \cap \text{RLT} \right\} \quad (\text{A.81})$$

The objective function is a DC function, and the feasible region is convex, so (A.81) is a DC program. One can employ the convex-concave procedure discussed in [17] to solve this problem. The flowchart is summarized in Algorithm A.1.

For the convergence of Algorithm A.1, we have the following properties.

Proposition A.9. [18] *The optimal value sequence $F(Y^i)$ generated by Algorithm A.1 is monotonically decreasing.*

Denote by $F(Y) = f(Y) - \frac{\rho}{2} g(Y)$ the objective function of (A.81) in the DC form, and $H(Y, Y^i) = f(Y) - \frac{\rho}{2} g_L(Y, Y^i)$ the convexified objective function in (A.83) by linearizing the concave term in $F(Y)$. Two basic facts help explain this proposition:

Algorithm A.1 : Sequential SDP

-
- 1: Choose an initial penalty parameter ρ^0 , a penalty growth rate $\tau > 0$, and solve the following SDP relaxation model

$$\min \{ \langle \Omega, Y \rangle \mid Y \in \hat{L} \cap \text{LMI} \cap \text{RLT} \}$$

The optimal solution is Y^* .

- 2: Construct the linear approximation of $g(Y)$ as

$$g_L(Y, Y^*) = g(Y^*) + \langle \nabla g(Y^*), Y - Y^* \rangle \quad (\text{A.82})$$

Solve the the following SDP

$$\min_Y \left\{ f(Y) - \frac{\rho}{2} g_L(Y, Y^*) \mid Y \in \hat{L} \cap \text{LMI} \cap \text{RLT} \right\} \quad (\text{A.83})$$

The optimal solution is Y^* .

- 3: If $\text{rank}(Y^*) = 1$, terminate and report the optimal solution Y^* ; otherwise, update $\rho \leftarrow (1 + \tau)\rho$, and go to step 2.
-

1) $g_L(Y^*, Y^*) = g(Y^*)$, $\forall Y^*$ which directly follows from the definition in (A.82).

2) For any given Y^* , $g(Y) \geq g_L(Y, Y^*)$, $\forall Y$, because the graph of a convex function must lie over its tangent plane at any fixed point.

First we can asset inequality $H(Y^{i+1}, Y^i) \leq H(Y^i, Y^i)$, because $H(Y, Y^i)$ is optimized in problem (A.83). The optimum $H(Y^{i+1}, Y^i)$ deserves a value no greater than that at any feasible point. Furthermore, with the definition of $H(Y^i, Y^i)$, we have

$$H(Y^{i+1}, Y^i) \leq H(Y^i, Y^i) = f(Y^i) - \frac{\rho}{2} g_L(Y^i, Y^i) = f(Y^i) - \frac{\rho}{2} g(Y^i) = F(Y^i)$$

On the other hand,

$$H(Y^{i+1}, Y^i) = f(Y^{i+1}) - \frac{\rho}{2} g_L(Y^{i+1}, Y^i) \geq f(Y^{i+1}) - \frac{\rho}{2} g(Y^{i+1}) = F(Y^{i+1})$$

Consequently, we arrive at the monotonic property

$$F(Y^{i+1}) \leq F(Y^i)$$

Proposition A.10. [18] *The solution sequence Y^i generated by Algorithm A.1 approaches to the optimal solution of problem (A.79) when $\rho \rightarrow \infty$.*

It is easy to understand that whenever ρ is sufficiently large, the penalty term will tend to 0, and the rank-1 constraint in (A.79) is met. A formal proof can be found in [18]. A few more remarks are given below.

1) The convex-concave procedure in [12] is a local algorithm under mild conditions and needs a manually supplied initial point. Algorithm A.1, however, is elaborately initiated at the solution offered by the SDP relaxation model, which usually appears to be close to the global optimal one for many engineering optimization problems. Therefore, Algorithm A.1 generally performs well and will identify the global optimal solution, although a provable guarantee is non-trivial.

2) In practical applications, Algorithm A.1 could converge without the penalty parameter approaching infinity, because when some constraint quantification holds, there exists an exact penalty parameter ρ^* , such that the optimal solution leads to a zero penalty term for any $\rho \geq \rho^*$ [19, 20], and Algorithm A.1 converges in a finite number of steps. If the exact penalty parameter does not exist, Algorithm A.1 may fail to converge. In such circumstance, one can impose an upper bound on ρ , and use an alternative convergence criterion: the change of the objective value $F(Y)$ in two consecutive steps is less than a given threshold value. As a result, Algorithm A.1 will be able to find an approximate solution of problem (A.79), and the rank-1 constraint may not be enforced.

3) From the numeric computation perspective, a very large ρ may cause ill-conditioned problem and lead to numerical instability, so it is useful to gradually increase ρ from a small value. Another reason for the moderate growth of ρ is that it does not cause dramatic change of optimal solutions in two successive iterations. As a result, $g_L(Y, Y^*)$ can provide relatively accurate approximation for $g(Y)$ in every iteration.

4) The penalty term $\rho_i p(Y^i)/2 = \rho_i [\|Y^i\|_2^2 - \text{tr}(Y^i U \Theta U^T Y^i)]/2$ gives an upper bound on the optimality gap induced by rank relaxation. To see this, let ρ^* and Y^* be the exact penalty parameter and corresponding optimal solution of (A.83), i.e., $p(Y^*) = 0$; ρ_i and Y^i be the penalty parameter and optimal solution in i -th iteration. According to Proposition A.9, we have $\langle \Omega, Y^* \rangle \leq \langle \Omega, Y^i \rangle + \rho_i p(Y^i)/2$; moreover, since the rank-1 constraint is relaxed before Algorithm A.1 could converge, $\langle \Omega, Y^i \rangle \leq \langle \Omega, Y^* \rangle$ holds. Therefore, $\langle \Omega, Y^i \rangle$ and $\langle \Omega, Y^i \rangle + \rho_i p(Y^i)/2$ are lower and upper bounds for the optimal value of problem (A.79). In this regard, $\rho_i p(Y^i)/2$ is an estimation on the optimality gap.

A.3.3 Completely Positive Program Relaxation

Inspired by the convex hull expression in (A.14), researchers have shown that most non-convex QCQPs can be modeled as linear programs over the intersection of a completely positive cone and a polyhedron [6, 7, 8]. For example, consider minimizing a quadratic function over a standard simplex

$$\begin{aligned}
\min \quad & x^T Q x \\
\text{s.t.} \quad & e^T x = 1 \\
& x \geq 0
\end{aligned} \tag{A.84}$$

where $Q \in \mathbb{S}^n$, and e denotes the all-one vector with n entries. Following the paradigm similar to (A.66), let $X = xx^T$, and then we can construct a valid inequality

$$1 = x^T e e^T x = x^T E x = \langle E, X \rangle$$

where $E = ee^T$ is the all-one matrix. According to (A.14), $\text{conv}\{xx^T | x \in \mathbb{R}_+^n\}$ is given by $(\mathbb{C}_+^n)^*$. Therefore, problem (A.84) transforms to

$$\begin{aligned}
\min \quad & \langle Q, X \rangle \\
\text{s.t.} \quad & \langle E, X \rangle = 1 \\
& X \in (\mathbb{C}_+^n)^*
\end{aligned} \tag{A.85}$$

Problem (A.85) is a convex relaxation of (A.84). Because the objective is linear, the optimal solution must be located at one extremal point of the convex hull of the feasible region. In view of the representation in (A.14), the extremal points are exactly rank-1, so the convex relaxation (A.85) is always exact.

Much more general results are demonstrated in [7] that every quadratic program with linear and binary constraints can be rewritten as a completely positive program. More precisely, a mixed-integer quadratic program

$$\begin{aligned}
\min \quad & x^T Q x + 2c^T x \\
\text{s.t.} \quad & a_i^T x = b_i, \quad i = 1, \dots, m \\
& x \geq 0, \quad x_j \in \{0, 1\}, j \in B
\end{aligned} \tag{A.86}$$

and the following completely positive program

$$\begin{aligned}
\min \quad & \langle Q, X \rangle + 2c^T x \\
\text{s.t.} \quad & a_i^T x = b_i, \quad i = 1, \dots, m \\
& \langle a_i a_i^T, X \rangle = b_i^2, \quad i = 1, \dots, m \\
& x_j = X_{jj}, \quad j \in B \\
& X \in (C_+^n)^*
\end{aligned} \tag{A.87}$$

have the same optimal solution, as long as problem (A.86) satisfies: $a_i^T x = b_i$, $\forall i$ and $x \geq 0$ implies $x_j \leq 1$, $\forall j \in B$. Actually, this is a relatively mild condition [7]. Complementarity constraints can be handled in the similar way. Whether problems with general quadratic constraints can be restated as completely positive programs in the similar way remains an open question.

The NP-hardness of problem (A.86) makes (A.87) NP-hard itself. The complexity has been encapsulated into the last cone constraint. The relax-

ation model is still interesting due to its convexity. Furthermore, it can be approximated via a sequence of SDPs with growing sizes [9] given an arbitrarily small error bound.

A.3.4 MILP Approximation

SDP relaxation technique introduces a squared matrix variable that contains $n(n+1)/2$ independent variables. Although exploiting the sparse pattern of X via graphic theory is helpful to expedite problem solution, the computational burden is still high especially when the initial relaxation is inexact and a sequence of SDPs should be solved. Inspired by difference-of-convex programming an alternative choice is to express the non-convexity of QCQP by univariate concave functions, and approximate these concave functions via a PWL functions compatible with mixed-integer programming solvers. This approach has been expounded in [21].

Consider nonconvex QCQP

$$\begin{aligned} \min \quad & x^T A_0 x + a_0^T x \\ \text{s.t.} \quad & x^T A_k x + a_k^T x \leq b_k, \quad k = 1, \dots, m \end{aligned} \quad (\text{A.88})$$

We can always find $\delta_0, \delta_1, \dots, \delta_m \in \mathbb{R}^+$, such that $A_k + \delta_k I \succeq 0, k = 0, \dots, m$. For example, δ_k can take the absolute value of the most negative eigenvalue of A_k , and $\delta_k = 0$ if $A_k \succeq 0$. Then, problem (A.88) can be cast as

$$\begin{aligned} \min \quad & x^T (A_0 + \delta_0 I) x + a_0^T x - \delta_0 1^T y \\ \text{s.t.} \quad & x^T (A_k + \delta_k I) x + a_k^T x - \delta_k 1^T y \leq b_k, \quad k = 1, \dots, m \\ & y_i = x_i^2, i = 1, \dots, n \end{aligned} \quad (\text{A.89})$$

Problem (A.89) is actually a difference-of-convex program; however, the non-convex terms are consolidated in much simpler parabolic equalities, which can be linearized via the SOS2 based PWL approximation technique discussed in Appendix B.1. Except for the last n quadratic equalities, remaining constraints and objective function of problem (A.89) are all convex, so the linearized problem gives rise to a mixed-integer convex quadratic program.

Alternatively, we can first perform convex relaxation by replacing $y_i = x_i^2$ with $y_i \geq x_i^2, i = 1, \dots, n$; if strict inequality holds at the optimal solution, a disjunctive cut is generated to remove this point from the feasible region. However, the initial convex relaxation can be very weak ($y = +\infty$ is usually an optimal solution). Predefined disjunctive cuts can be added [21].

Finally, nonconvex QCQP is a hard optimization problem. Developing an efficient algorithm should leverage specific problem structure. For example, SDP relaxation is suitable for OPF problems; MILP approximation can be

used for small and dense problems. Unlike SDP relaxation works on a squared matrix variable, the number of auxiliary variables in (A.89) and its mixed-integer convex quadratic program approximation is moderate. Therefore, this approach is promising to tackle practical problems whose coefficient matrices are usually sparse. Furthermore, no particular assumption is needed to guarantee the exactness of relaxation, so this method is general enough to tackle a wide spectrum of engineering optimization problems.

A.4 MILP Formulation of Nonconvex QPs

In a non-convex QCQP, if the constraints are all linear, it is called a nonconvex QP. There is no doubt that convex relaxation methods presented in the previous section can be applied to nonconvex QPs. However, the relaxation is generally inexact. In this section, we introduce exact MILP formulations to globally solve such a nonconvex optimization problem; unlike the mixed-integer programming approximation method in Sect. A.3.4, in which approximation error is inevitable, by using duality theory, the MILP models will be completely equivalent to the original QP. Thanks to the advent of powerful MILP solvers, this method is becoming increasingly competitive compared to existing global solution methods and is attracting more attentions from the research community.

A.4.1 Nonconvex QPs over polyhedra

The presented approach is devised in [22]. A nonconvex QP with linear constraints has the form of

$$\begin{aligned} \min \quad & \frac{1}{2}x^T Qx + c^T x \\ \text{s.t.} \quad & Ax \leq b \end{aligned} \tag{A.90}$$

where Q is a symmetric, but indefinite matrix; A , b , c are constant coefficients with compatible dimensions. We assume that finite lower and upper limits of the decision variable x have been included, and thus the feasible region is a bounded polyhedron. The KKT conditions of (A.90) can be written as:

$$\begin{aligned} c + Qx + A^T \xi &= 0 \\ 0 &\leq \xi \perp b - Ax \geq 0 \end{aligned} \tag{A.91}$$

If there is a multiplier ξ so that the pair (x, ξ) of primal and dual variables satisfies KKT condition (A.91), then x is said to be a KKT point or a stationary point. The complementarity and slackness condition in (A.91)

gives $b^T \xi = x^T A^T \xi$. For any primal-dual pair (x, ξ) that satisfies (A.91), the following relations hold

$$\begin{aligned} \frac{1}{2} x^T Q x + c^T x &= \frac{1}{2} c^T x + \frac{1}{2} x^T (c + Qx) \\ &= \frac{1}{2} c^T x - \frac{1}{2} x^T A^T \xi = \frac{1}{2} (c^T x - b^T \xi) \end{aligned} \quad (\text{A.92})$$

As such, the non-convex quadratic objective function is equivalently stated as a linear function in the primal and dual variables without loss of accuracy. Thus, if problem (A.90) has an optimal solution, then the solution can be retrieved by solving an LPCC

$$\begin{aligned} \min \quad & \frac{1}{2} (c^T x - b^T \xi) \\ \text{s.t.} \quad & c + Qx + A^T \xi = 0 \\ & 0 \leq \xi \perp b - Ax \geq 0 \end{aligned} \quad (\text{A.93})$$

which is equivalent to the following MILP

$$\begin{aligned} \min \quad & \frac{1}{2} c^T x - b^T \xi \\ \text{s.t.} \quad & c + Qx + A^T \xi = 0 \\ & 0 \leq \xi \leq M(1 - z) \\ & 0 \leq b - Ax \leq Mz \\ & z \text{ binary} \end{aligned} \quad (\text{A.94})$$

where M is a sufficiently large constant; z is a vector of binary variables. Regardless of the value of z_i , at most one of ξ_i and $(b - Ax)_i$ can take a strictly positive value. For more rigorous discussions on this method, please see [22], in which an unbounded feasible region is considered. More tricks in MILP reformulation technique can be found in the next chapter.

It should be pointed out that the set of optimal solutions of (A.90) is a subset of stationary points described by (A.91), because (A.91) is only a necessary condition for optimality but not sufficient. Nevertheless, as we assumed that the feasible region is a bounded polytope (thus compact), QP (A.90) must have a finite optimum, then according to [23], the optimal value is equal to the minimum of objective function values perceived at stationary points. Therefore, MILP (A.94) provides an exact solution to (A.90).

Finally, we shed some light on the selection of M , since it has notable impact on the computational efficiency of (A.94). An LP based bound preprocessing method is thoroughly discussed in [24], which is used in a finite branch-and-bound method for solving LPCC (A.93). Here we briefly introduce the bounding method.

For the primal variable x which represents physical quantities or measures, its bounds depends on practical situations and security considerations, and we assume that the bound is $0 \leq x \leq U$. The bound can be tightened by solving

$$\min(\max) \{x_j \mid Ax \leq b, 0 \leq x \leq U\} \quad (\text{A.95})$$

In (A.95), we can incorporate individual bounds for the components of vector x , which never wrecks the optimal solution and can be supplemented in (A.94).

For the dual variables, we consider (A.90) again with explicit bounds on primal variable x

$$\begin{aligned} \min \quad & \frac{1}{2}x^T Qx + c^T x \\ \text{s.t.} \quad & Ax \leq b : \xi \\ & 0 \leq x \leq U : \lambda, \rho \end{aligned}$$

where ξ, λ, ρ following the colon are dual variables. Its KKT condition reads

$$c + Qx + A^T \xi - \lambda + \rho = 0 \quad (\text{A.96a})$$

$$0 \leq \xi \perp b - Ax \geq 0 \quad (\text{A.96b})$$

$$0 \leq x \perp \lambda \geq 0 \quad (\text{A.96c})$$

$$0 \leq U - x \perp \rho \geq 0 \quad (\text{A.96d})$$

Multiplying both sides of (A.96a) by a feasible solution x^T

$$c^T x + x^T Qx + x^T A^T \xi - x^T \lambda + x^T \rho = 0 \quad (\text{A.96e})$$

Substituting $\xi^T Ax = \xi^T b$, $x^T \lambda = 0$, and $x^T \rho = \rho^T U$ concluded from (A.96b)-(A.96d) into (A.96e) outcomes

$$c^T x + x^T Qx + b^T \xi + U^T \rho = 0 \quad (\text{A.96f})$$

The upper bounds (lower bounds are 0) on the dual variables required for MILP (A.94) can be computed from the following LP:

$$\max \quad \lambda_j \quad (\text{A.97a})$$

$$\text{s.t.} \quad c + Qx + A^T \xi - \lambda + \rho = 0 \quad (\text{A.97b})$$

$$\text{tr}(Q^T X) + c^T x + b^T \xi + U^T \rho = 0 \quad (\text{A.97c})$$

$$\text{Cons-RLT} = \{(x, X) \mid (\text{A.74})\} \quad (\text{A.97d})$$

$$0 \leq x \leq U, Ax \leq b, \lambda, \xi, \rho \geq 0 \quad (\text{A.97e})$$

In (A.97c), quadratic equality (A.96f) is linearized by letting $X = xx^T$, and (A.97d) is a linear relaxation for above rank-1 condition, as explained in Sect. A.3.1.

By exploiting the relaxation revealed in (A.97c), it has been proved that problem (A.97) always has a finite optimum, because the recession cone of the set comprised of the primal and dual variables as well as their associated valid inequalities is empty, see the proof of Proposition 3.1 in [24]. This is a pivotal theoretical guarantee. Other bounding techniques which only utilize KKT conditions hardly ensure a finite optimum.

A.4.2 Standard Nonconvex QPs

The presented approach is devised in [28]. A standard nonconvex QP entails minimizing a nonconvex quadratic function over a unit probability simplex

$$\begin{aligned} v(Q) = \min \quad & x^T Q x \\ \text{s.t.} \quad & x \in \Delta_n \end{aligned} \quad (\text{A.98})$$

where Q is a symmetric matrices, and unit simplex

$$\Delta_n = \{x \in \mathbb{R}_+^n \mid e^T x = 1\}$$

where e is all-one vector. A nonhomogeneous objective can always be transformed to a quadratic form given the simplex constraint Δ_n :

$$x^T Q x + 2c^T x = x^T (Q + ec^T + ce^T)x, \quad \forall x \in \Delta_n$$

Standard nonconvex QPs have wide applications in portfolio optimization, quadratic resource allocation, graphic theory and so on. In addition, for a given symmetric matrix Q , a necessary and sufficient condition for Q being copositive is $v(Q) \geq 0$. Copositive programming is a young and active research field, and can help the research in convex relaxation. A fundamental problem is copositivity test, which entails solving (A.98) globally.

Problem (A.98) is a special case of nonconvex QP (A.90), so the methods in previous subsection also work for (A.98). The core trick is to select a big-M parameter in linearizing complementarity and slackness conditions. Due to its specific structure, the valid big-M parameter for problem (A.98) can be chosen in a much more convenient way. To see this, the KKT condition of (A.98) reads as

$$Qx - \lambda e - \mu = 0 \quad (\text{A.99a})$$

$$e^T x = 1 \quad (\text{A.99b})$$

$$x \geq 0 \quad (\text{A.99c})$$

$$\mu \geq 0 \quad (\text{A.99d})$$

$$x_j \mu_j = 0, \quad j = 1, \dots, n \quad (\text{A.99e})$$

where λ and μ are dual variables associated with equality constraint $e^T x = 1$ and inequality constraint $x \geq 0$. Because the feasible region is polyhedral, constraint quantification always holds, and any optimal solution of (A.98) must solve KKT system (A.99).

Multiplying both sides of (A.99a) by x results in $x^T Qx = \lambda x^T e - x^T \mu$; substituting (A.99b) and (A.99e) into the right-hand side concludes $x^T Qx = \lambda$. Provided with eligible big-M parameter, problem (A.98) is (exactly) equivalent to the following MILP

$$\begin{aligned} \min \quad & \lambda \\ \text{s.t.} \quad & Qx - \lambda e - \mu = 0 \\ & e^T x = 1, \quad 0 \leq x \leq y \\ & 0 \leq \mu_j \leq M_j(1 - y_j), \quad j = 1, \dots, n \end{aligned} \tag{A.100}$$

where $y \in \{0, 1\}^n$, and M_j is the big-M parameter. It is the upper bound of dual variable μ_j . To estimate such a bound, according to (A.99a)

$$\mu_j = e_j^T Qx - \lambda, \quad j = 1, \dots, n$$

where e_j is the j -th column of $n \times n$ identity matrix. For the first term,

$$x^T Qe_j \leq \max_{i \in \{1, \dots, n\}} Q_{ij}, \quad j = 1, \dots, n$$

As for the second term, we know $\lambda \geq v(Q)$, so any known lower bound of $v(Q)$ can be used to obtain an upper bound of M_j . One possible lower bound of $v(Q)$ is suggested in [28] as

$$l(Q) = \min_{1 \leq i, j \leq n} Q_{ij} + \frac{1}{\sum_{k=1}^n \left(Q_{kk} - \min_{1 \leq i, j \leq n} Q_{ij} \right)^{-1}}$$

If the minimal element of Q locates on the main diagonal, the second term vanishes and $l(Q) = \min_{1 \leq i, j \leq n} Q_{ij}$.

In summary, a valid choice of M_j would be

$$M_j = \max_{i \in \{1, \dots, n\}} Q_{ij} - l(Q), \quad j = 1, \dots, n \tag{A.101}$$

It is found in [28] that if we relax (A.99a) as an inequality and solve the following MILP

$$\begin{aligned} \min \quad & \lambda \\ \text{s.t.} \quad & Qx - \lambda e - \mu \leq 0 \\ & e^T x = 1, \quad 0 \leq x \leq y \\ & 0 \leq \mu_j \leq M_j(1 - y_j), \quad j = 1, \dots, n \end{aligned} \tag{A.102}$$

which is an relaxed version of (A.100), the optimal solution will not change. However, in some instances, solving (A.102) is significantly faster than solving (A.100). More thorough theoretical analysis can be found in [28].

A.5 Further Reading

Decades of wonderful research has resulted in elegant theoretical developments and sophisticated computational softwares, which have brought convex optimization to an unprecedented dominating stage where it serves as the baseline and reference model for optimization problems in almost every discipline. Only problems which can be formulated as convex programs are regarded as theoretically solvable. We suggest following materials for readers who want to build a solid mathematical background or know more about applications in the field of convex optimization.

1. Convex analysis and convex optimization. Convex analysis is a classic topic in mathematics, and focuses on basic concepts and topological properties of convex sets and convex functions. We recommend monographs [25, 26, 27]. The last one sheds more light on optimization related topics, including DC programming, polynomial programming, and equilibrium constrained programming, which are originally non-convex. The most popular textbooks on convex optimization include [1, 2]. They contain important materials that everyone who wants to apply this technique should know.

2. Special convex optimization problems. The most mature convex optimization problems are LPs, SOCPs, and SDPs. We recommend [29, 30, 31] for the basic knowledge of duality theory, simplex algorithm, interior-point algorithm, and applications of LPs. The modeling abilities of SOCPs and SDPs have been well discussed in [1, 2]. A geometric program is a type of optimization problem whose objective and constraints are characterized by special monomials and posynomial functions. Through a logarithmic variable transformation, a geometric program can be mechanically converted to a convex optimization problem. Geometric programming is relatively restrictive in structure, and it may not be apparent to see whether a given problem can be expressed by a geometric program. We recommend a tutorial paper [32] and references therein on this topic. Copositive program is a relatively young field in operational research. It is a special class of conic programming which is more general than SDP. Basic information on copositive/completely positive programs is introduced in [33, 34, 35]. They are particularly useful in combinatorial and quadratic optimization. Though very similar to SDPs in appearances, copositive programs are NP-hard. Algorithms and applications of copositive and completely positive programs have continued to be highly active research fields [36, 37, 38].

3. General convex optimization problems. Beside above mature convex optimization models that can be specified without high level of expertise,

recognizing the convexity of a general mathematical programming problem may be rather tricky. A deep understanding on convex analysis is unavoidable. Furthermore, to solve the problem using off-the-shelf solvers, a user must find a way to transform the problem into one of the standard forms (if a general purpose NLP solver fails to solve it). The so-called disciplined convex programming method is proposed in [39] to lower this expertise barrier. The method consists of a set of rules and conventions that one must follow when setting up the problem such that the convexity is naturally sustained. This methodology has been implemented in `cvx` toolbox under Matlab environment.

4. Convex relaxation methods. One major application of convex optimization is to derive tractable approximations for non-convex programs, so as to facilitate problem resolution in terms of computational efficiency and robustness. A general QCQP is a quintessential non-convex optimization problem. Among various convex relaxation approaches, the SDP relaxation is shown to be able to offer high quality solutions for many QCQPs raised in signal process [40, 41] and power system energy management [42, 43]. Decades of excellent studies on SDP relaxation methods for QCQPs are comprehensively reviewed in [13, 44, 45]. Some recent advances are reported in [46, 47, 48, 49, 50, 51, 52]. The rank of the matrix variable has a decisive impact on the exactness (or tightness) of the SDP relaxation. Low rank SDP method are attracting increasing attentions from researchers, and many approaches are proposed to recover a low-rank solution. More information can be found in [53, 54, 55, 56, 57] and references therein.

5. Sum-of-squares (SOS) programming is originally devised in [58] to decompose a polynomial $f(x)$ as the square of another polynomial $g(x)$ (if there exists), such that $f(x) = [g(x)]^2$ must be non-negative. Non-negativity of a polynomial over a semi-algebraic set can be certified in a similar way via Positivstellensatz refutations. This can be done by solving a structured SDP [58], and implemented in a Matlab based toolbox [59]. Based on these outcomes, a promising methodology is quickly developed for polynomial programs, which cover a broader class of optimization problems than QCQPs. It is proved that the global solution of a polynomial program can be found by solving a hierarchy of SDPs under mild conditions. This is very inspiring since polynomial programs are generally non-convex while SDPs are convex. We recommend [60] for a very detailed discussion on this approach, and [61, 62, 63, 64] for some recent advances. However, users should be aware that this approach may be unpractical because the size of the relaxed SDP quickly becomes unacceptable after a few steps. Nonetheless, the elegant theory still marks a milestone in the research field.

References

1. Ben-Tal, A., Nemirovski, A.: Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering applications. SIAM, Philadelphia, PA, USA (2001) available online: http://www2.isye.gatech.edu/~nemirovs/Lect_ModConvOpt.pdf
2. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, New York, NY (2004)
3. Boyd, S., Vandenberghe, L.: Convex Optimization Solution Manual (2006)
4. Hiriart-Urruty, J.B., Seeger, A.: A variational approach to copositive matrices. SIAM Rev. 52(4): 593-629 (2010)
5. Berman, A., Shaked-Monderer, N.: Completely Positive Matrices, World Scientific, Singapore (2003)
6. Bomze, I.M., Dür, M., de Klerk, E., Roos, C., Quist, A.J., Terlaky, T.: On copositive programming and standard quadratic optimization problems. J. Global Optim. 18(4), 301-320 (2000)
7. Burer, S.: On the copositive representation of binary and continuous nonconvex quadratic programs. Math. Program. 120(2): 479-495 (2009)
8. Povh, J.: Towards the optimum by semidefinite and copositive programming : New approach to approximate hard optimization problems. VDM Verlag (2009)
9. Parrilo, P.: Structured Semidefinite Programs and Semi-algebraic Geometry Methods in Robustness and Optimization. PhD thesis, California Institute of Technology (2000)
10. Lobo, M.S., Vandenberghe, L., Boyd, S., Lebret, H. Applications of second-order cone programming. Linear Algebra Appl. 284(1-3): 193-228 (1998)
11. Ben-Tal, A., Nemirovski, A. On polyhedral approximations of the second-order cone. Math. Oper. Res. 26(2): 193-205 (2001)
12. Vandenberghe, L., Boyd, S.: Semidefinite programming. SIAM Rev. 38(1): 49-95 (1996)
13. Burer, S., Saxena, A.: The MILP Road to MIQCP. In: Lee, J., Leyffer, S. (eds) Mixed Integer Nonlinear Programming. The IMA Volumes in Mathematics and its Applications, vol 154. Springer, New York, NY (2012)
14. Sherali, H.D., Adams, W.P.: A reformulation-linearization technique for solving discrete and continuous nonconvex problems. Springer (1999)
15. McCormick, G.P.: Computability of global solutions to factorable nonconvex programs: Part I - Convex underestimating problems. Math. Program. 10(1): 147-175 (1976)
16. Saxena, A., Bonami, P., Lee, J.: Convex relaxations of non-convex mixed integer quadratically constrained programs: extended formulations. Math. Program. 124(1): 383-411 (2010)
17. Lipp, T., Boyd, S.: Variations and extension of the convex-concave procedure. Optim. Eng. 17(2): 263-287 (2016)
18. Yang, W., Zhao, J., Xu, F.: An efficient method for convex constrained rank minimization problems based on DC programming. Math. Probl. Eng. Article ID 7473041, (2016)
19. Han, S.P., Mangasarian, O.L.: Exact penalty functions in nonlinear programming. Math. Program. 17(1):251-269 (1979)
20. Di Pillo, G., Grippo, L.: Exact penalty functions in constrained optimization. SIAM J. Control Optim. 27(6):1333-1360 (1989)
21. Dong, H., Luo, Y.: Compact disjunctive approximations to nonconvex quadratically constrained programs. arXiv:1811.08122v1 (2018)
22. Hu, J., Mitchell, J.E., Pang, J.S.: An LPCC approach to nonconvex quadratic programs. Math. Program. 133(1): 243-277 (2012)

23. Giannessi, F., Tomasin, E. : Nonconvex quadratic programs, linear complementarity problems, and integer linear programs. In: Conti, R., Ruberti, A. (eds.) Fifth Conference on Optimization Techniques (Rome 1973), Part I, Lecture Notes in Computer Science, vol. 3, pp. 437-449. Springer, Berlin (1973)
24. Chen, J., Burer, S.: Globally solving nonconvex quadratic programming problems via completely positive programming. *Math. Prog. Comp.* 4(1):33-52 (2012)
25. Rockafellar, R.T.: *Convex analysis*. Princeton University Press, Princeton, NJ (2015)
26. Bertsekas, D.P., Nedi, A., Ozdaglar, A.E.: *Convex Analysis and Optimization*. Athena Scientific, Belmont, MA, (2003)
27. Tuy, H.: *Convex Analysis and Global Optimization* (2nd Edition). Springer, (2016)
28. Gondzio, J., Yildirim, E. A.: Global solutions of nonconvex standard quadratic programs via mixed integer linear programming reformulations. *arXiv preprint arXiv:1810.02307* (2018)
29. Dantzig, G.: *Linear Programming and Extensions*. Princeton University Press, Princeton, NJ (2016)
30. Bertsimas, D., Tsitsiklis, J.N.: *Introduction to Linear Optimization*. Athena Scientific, Belmont, MA (1997)
31. Vanderbei, R.J.: *Linear Programming*. Springer, New York, NY, (2015)
32. Boyd, S., Kim, S.J., Vandenberghe, L., Hassibi, A.: A tutorial on geometric programming. *Optim. Eng.* 8(1): 67-127 (2007)
33. Dür, M.: Copositive Programming - a Survey. In: Diehl, M., Glineur, F., Jarlebring, E., Michiels, W. (eds) *Recent Advances in Optimization and its Applications in Engineering*. Springer, Berlin, Heidelberg (2010)
34. Gijben L. On approximations, complexity, and applications for copositive programming. *Rijksuniversiteit Groningen*, 2015.
35. Burer, S.: Copositive programming. In: *Handbook on Semidefinite, Conic and Polynomial Optimization*, 201-218, Springer, New York, NY, USA (2012)
36. Lasserre, J.B.: New approximations for the cone of copositive matrices and its dual. *Math. Program.* 144(1): 265-276 (2014)
37. Bomze I M. Copositive relaxation beats Lagrangian dual bounds in quadratically and linearly constrained quadratic optimization problems. *SIAM J. Optim.* 25(3): 1249-1275 (2015)
38. Berman A, Dur M, Shaked-Monderer N. Open problems in the theory of completely positive and copositive matrices. *Electronic Journal of Linear Algebra* 29(1): 46-58 (2015)
39. Grant, M., Boyd, S., Ye, Y.: *Disciplined convex programming*. In: *Global Optimization: From Theory to Implementation*. Springer US, 155-210 (2006)
40. Palomar, D.P., Eldar, Y.C.: *Convex Optimization in Signal Processing and Communications*. Cambridge University Press, Cambridge, UK (2010)
41. Luo, Z.Q., Ma, W.K., So, A.M.C., Ye, Y.: Semidefinite relaxation of quadratic optimization problems. *IEEE Signal Proc. Mag.* 27(3): 20-34 (2010)
42. Maher, A.: *Semidefinite programming: Methods and algorithms for energy management*. Working paper, available at: <https://tel.archives-ouvertes.fr/tel-00881025/document>.
43. Taylor, J.A.: *Convex Optimization of Power Systems*. Cambridge University Press, (2015)
44. Bao, X., Sahinidis, N.V., Tawarmalani, M.: Semidefinite relaxations for quadratically constrained quadratic programming: A review and comparisons. *Math. Program.* 129(1): 129-157 (2011)
45. Anstreicher, K.M.: On convex relaxations for quadratically constrained quadratic programming. *Math. Program.* 136(2): 233-251 (2012)

46. Bose, S., Gayme, D.F., Chandy, K.M., Low, S.H.: Quadratically constrained quadratic programs on acyclic graphs with application to power flow. *IEEE Trans. Control of Network Syst.* 2(3): 278-287 (2015)
47. Xu, Z., Hong, M., Luo, Z.Q.: Semidefinite approximation for mixed binary quadratically constrained quadratic programs. *SIAM J. Optim.* 24(3): 1265-1293 (2014)
48. Billionnet, A., Elloumi, S., Lambert, A.: Exact quadratic convex reformulations of mixed-integer quadratically constrained problems. *Math. Program.* 158(1): 235-266 (2016)
49. Burer, S., Kim, S., Kojima, M.: Faster, but weaker, relaxations for quadratically constrained quadratic programs. *Comput. Optim. Appl.* 59(1-2): 27-45 (2014)
50. Jiang, R., Li, D.: Convex relaxations with second-order cone constraints for nonconvex quadratically constrained quadratic programming. Working paper, arXiv:1608.02096 (2016)
51. Shtern, S., Ben-Tal, A.: Computational methods for solving nonconvex block-separable constrained quadratic problems. *SIAM J. Optimiz.* 26(2): 1174-1206 (2016)
52. Ben-Tal, A., Den Hertog, D.: Hidden conic quadratic representation of some nonconvex quadratic optimization problems. *Math. Program.* 143(1): 1-29 (2014)
53. Burer, S., Monteiro, R.D.C.: Local minima and convergence in low-rank semidefinite programming. *Math. Program.* 103(3): 427-444 (2005)
54. Burer, S., Choi, C.: Computational enhancements in low-rank semidefinite programming. *Optim. Method. Softw.* 21(3): 493-512 (2006)
55. Journée, M., Bach, F., Absil, P., Sepulchre, R.: Low-rank optimization on the cone of positive semidefinite matrices. *SIAM J. Optimiz.* 20(5): 2327-2351 (2010)
56. Madani, R., Sojoudi, S., Fazelnia, G., Lavaei, J.: Finding low-rank solutions of sparse linear matrix inequalities using convex optimization. *SIAM J. Optimiz.* 27(2): 725-758 (2017)
57. Lemon, A., So, A.M.C., Ye, Y.: Low-rank semidefinite programming: Theory and applications. *Foundations and Trends in Optimization* 2(1-2): 1-156 (2016)
58. Parrilo, P.A.: Semidefinite programming relaxations for semialgebraic problems. *Math. Program.* 96(2): 293-320 (2003)
59. Papachristodoulou, A., Anderson, J., Valmorbida, G., Prajna, S., Seiler, P., Parrilo, P.: SOSTOOLS version 3.00 sum-of-squares optimization toolbox for MATLAB. User's guide, available at: <https://arxiv.org/abs/1310.4716> 2013.
60. Lasserre, J.B.: *Moments, Positive Polynomials and Their Applications*. Imperial College Press, London UK (2009)
61. De Klerk, E., Laurent, M.: On the Lasserre hierarchy of semidefinite programming relaxations of convex polynomial optimization problems. *SIAM J. Optimiz.* 21(3): 824-832 (2011)
62. Lasserre, J.B., Toh, K.C., Yang, S.: A bounded degree SOS hierarchy for polynomial optimization. *EURO J. Comput. Optim.* 5(1): 87-117 (2017)
63. Jeyakumar, V., Lasserre, J.B., Li, G.: On polynomial optimization over non-compact semi-algebraic sets. *J. Optimiz. Theory Appl.* 163(3): 707-718 (2014)
64. Nie J. Optimality conditions and finite convergence of Lasserre's hierarchy. *Math. Program.* 146(1): 97-121 (2014)

Appendix B

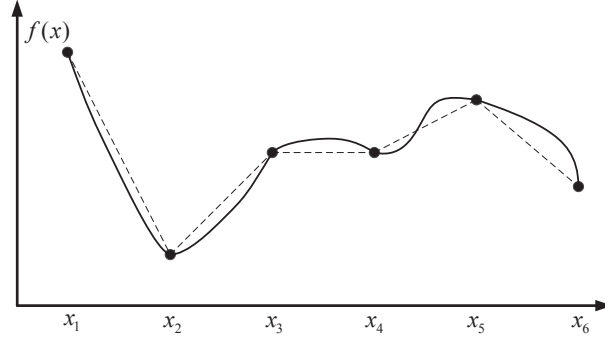
Formulation Recipes in Integer Programming

There is no problem in all mathematics that cannot be solved by direct counting. But with the present implements of mathematics many operations can be performed in a few minutes which without mathematical methods would take a lifetime.

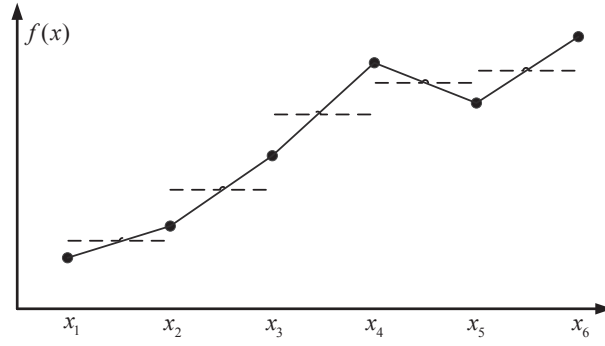
—Ernst Mach

As stated in Appendix A, generally speaking, convex optimization problems can be solved efficiently. However, the majority of optimization problems encountered in practical engineering are non-convex, and gradient based NLP solvers terminate at a local optimum, which may be far away from the global one. In fact, any nonlinear function can be approximated by a PWL function with adjustable errors by controlling the granularity of partitions. A PWL function can be expressed via a logic form or incorporating integer variables. Thanks to the latest progress in branch-and-cut algorithms and the development of state-of-the-art MILP solvers, a large-scale MILP can often be solved globally within reasonable computational efforts [1], although the MILP itself is proved to be NP-hard. In view of this fact, PWL/MILP approximation serves as a viable option to tackle real-world non-convex optimization problems, especially those with special structures.

This chapter introduces PWL approximation methods for nonlinear functions and linear representations of special non-convex constraints via integer programming techniques. When the majority of a problem at hand is linear or convex, while non-convexity arises from nonlinear functions with only one or two variables, linear complementarity constraints, logical inferences and so on, it is worth trying the methods in this chapter, in view of the fact that MILP solvers are becoming increasingly efficient to retrieve a solution with a pre-defined optimality gap.



(a) Piecewise linear approximation (dash line)



(b) Piecewise constant approximation (dash line)

Fig. B.1 Piecewise linear and piecewise constant approximations.

B.1 Piecewise Linear Approximation of Nonlinear Functions

B.1.1 Univariate Continuous Function

Considering a nonlinear continuous function $f(x)$ in a single variable x , we can evaluate the function values $f(x_0), f(x_1), \dots, f(x_n)$ at given breakpoints x_0, x_1, \dots, x_k , and replace $f(x)$ with the following PWL function

$$f(x) = \begin{cases} m_1x + c_1, & x \in [x_0, x_1] \\ m_2x + c_2, & x \in [x_1, x_2] \\ \vdots & \vdots \\ m_kx + c_k, & x \in [x_{k-1}, x_k] \end{cases} \quad (\text{B.1})$$

As an illustrative example, two curves of the original nonlinear function and its PWL approximation are portrayed in part (a), Fig. B.1. The PWL function in (B.1) is a finite union of line segments, but still non-convex. Moreover, the logic representation in (B.1) is not compatible with commercial solvers. Given the fact that any point on a line segment can be expressed as a convex combination of two terminal points, (B.1) can be written as

$$\begin{aligned}
 x &= \sum_i \lambda_i x_i \\
 y &= \sum_i \lambda_i f(x_i) \\
 \lambda &\geq 0, \quad \sum_i \lambda_i = 1 \\
 \lambda &\in \text{SOS}_2
 \end{aligned} \tag{B.2}$$

where SOS_2 stands for the special ordered set of type 2, describing a vector of variables with at most two adjacent ones being able to take nonzero values. The SOS_2 constraint on λ can be declared via the build-in module of commercial solvers such as CPLEX or GUROBI. Please note that if $f(x)$ is convex and to be minimized, then the last SOS_2 requirement is naturally met (thus can be relaxed), because the epigraph of $f(x)$ is a convex region. Otherwise, relaxing the last SOS_2 constraint in (B.2) gives rise to the convex hull of the sampled points $(x_0, y_0), \dots, (x_k, y_k)$. In general, the relaxation is inexact.

Branch-and-bound algorithms which directly working on SOS variables exhibit good performance [2], but it is desirable to explore equivalent MILP formulations to leverage the superiority of state-of-the-art solvers. To this end, we first provide an explicit form using additional integer variables.

$$\begin{aligned}
 \lambda_0 &\leq z_1 \\
 \lambda_1 &\leq z_1 + z_2 \\
 \lambda_2 &\leq z_2 + z_3 \\
 &\vdots \\
 \lambda_{k-1} &\leq z_{k-1} + z_k \\
 \lambda_k &\leq z_k \\
 z_i &\in \{0, 1\}, \quad \forall i, \quad \sum_{i=1}^k z_i = 1 \\
 \lambda_i &\geq 0, \quad \forall i, \quad \sum_{i=0}^k \lambda_i = 1
 \end{aligned} \tag{B.3}$$

Formulation (B.3) illustrates how integer variables can be used to enforce SOS_2 requirements on the weighting coefficients. This formulation does not

involve any manually supplied parameter, and often gives stronger bounds when the integrality of binary variables are relaxed.

Sometimes, it is more convenient to use a piecewise constant approximation, especially when the original function $f(x)$ is not continuous. An example is exhibited in part (b), Fig. B.1. In this approach, the feasible interval of x is partitioned into $S - 1$ segments (associated with binary variables θ_s , $s = 1, \dots, S - 1$) by S breakpoints x_1, \dots, x_S (associated with S continuous weight variables λ_s , $s = 1, \dots, S$); In the s -th interval between x_s and x_{s+1} , the function value $f(x)$ is approximated by the arithmetic mean $f_s = 0.5[f(x_s) + f(x_{s+1})]$, $s = 1, \dots, S - 1$, which is a constant as illustrated in Fig. B.1. With an appropriate number of partitions, an arbitrary function $f(x)$ can be approximated by a piecewise constant function as follows

$$x = \sum_{s=1}^S \lambda_s x_s, \quad y = \sum_{s=1}^{S-1} \theta_s f_s \quad (\text{B.4a})$$

$$\lambda_1 \leq \theta_1, \quad \lambda_S \leq \theta_{S-1} \quad (\text{B.4b})$$

$$\lambda_s \leq \theta_{s-1} + \theta_s, \quad s = 2, \dots, S - 1 \quad (\text{B.4c})$$

$$\lambda_s \geq 0, \quad s = 1, \dots, S, \quad \sum_{s=1}^S \lambda_s = 1 \quad (\text{B.4d})$$

$$\theta_s \in \{0, 1\}, \quad s = 1, \dots, S - 1, \quad \sum_{s=1}^{S-1} \theta_s = 1 \quad (\text{B.4e})$$

In (B.4), binary variable $\theta_s = 1$ indicates interval s is activated, and constraint (B.4e) ensures that only one interval will be activated; Furthermore, constraints (B.4b)-(B.4d) enforce weight coefficients α_s , $s = 1, \dots, S$ to be \mathbb{SO}_2 ; Finally, constraint (B.4a) expresses y and x via the linear combination of sampled values. The advantage of piecewise constant formulation (B.4) lies in the binary expression of function value y , such that the product of y and another continuous variable can be easily linearized via integer programming technique, which can be seen in Sect. B.2.3.

Clearly, the required number of binary variables introduced in formulation (B.3) is k , which grows linearly with respect to the number of breakpoints, and the final MILP model may suffer from computational overheads due to the presence of a large number of binary variables when more breakpoints are involved for improving accuracy. In what follows, we present a useful formulation that only engages a logarithmic number of binary variables and constraints. This technique is proposed in [3, 4, 5]. Consider the following constraints:

Gray codes						
λ^0					R_1	R_2 R_3
λ^1	G_1	0	0	0	R_1	R_2
λ^2	G_2	0	0	1	R_1	L_3
λ^3	G_3	0	1	1	R_1	L_2
λ^4	G_4	0	1	0		L_2 R_3
λ^5	G_5	1	1	0	L_1	R_3
λ^6	G_6	1	0	0	L_1	R_2
λ^7	G_7	1	0	1	L_1	L_3
λ^8	G_8	1	1	1	L_1	L_2 L_3

λ^0					R_1	R_2
λ^1	G_1	0	0		R_1	
λ^2	G_2	0	1			L_2
λ^3	G_3	1	1		L_1	
λ^4	G_4	1	0		L_1	R_2

Fig. B.2 Gray codes and sets L_n, R_n for two and three binary variables.

$$\begin{aligned}
\sum_{i \in L_n} \lambda_i &\leq z_n, \quad \forall n \in N \\
\sum_{i \in R_n} \lambda_i &\leq 1 - z_n, \quad \forall n \in N \\
z_n &\in \{0, 1\}, \quad \forall n \in N \\
\lambda &\geq 0, \quad \sum_{i=0}^k \lambda_i = 1
\end{aligned} \tag{B.5}$$

where L_n and R_n are index sets of weights λ_i , N is an index set corresponding to the number of binary variables. The dichotomy sequences $\{L_n, R_n\}_{n \in N}$ constitute a branching scheme on the indices of weights, such that constraint (B.5) guarantees that at most two adjacent elements of λ can take strictly positive values, so as to meet the SOS_2 requirement. The required number of binary variables z_n is $\lceil \log_2 k \rceil$, which is significantly smaller than that involved in formulation (B.3).

Next, we demonstrate how to design the sets L_n and R_n based on the concept of Gray codes. For notation brevity, we restrict the discussion to the instances with 2 and 3 binary variables (which are shown in Fig. B.2), indicating 5 and 9 breakpoints (or 4 and 8 intervals) in consequence.

As shown in Fig. B.2, Gray codes $G_1 - G_8$ form a binary system where any two adjacent numbers only differ in one bit. For example, G_4 and G_5 differ in the first bit, and G_5 and G_6 differ in the second bit. Such Gray codes are

used to describe which two adjacent weights are activated. In general, sets R_n and L_n are constructed as follows: the index $v \in L_n$ if the binary values of the n -th bit of two successive codes G_n and G_{n+1} are equal to 1, or $v \in R_n$ if they are equal to 0. This principle can be formally defined in a mathematical way as

$$L_n = \left\{ v \left| \begin{array}{l} (G_v^n = 1 \text{ and } G_{v+1}^n = 1) \\ \cup (v = 0 \text{ and } G_1^n = 1) \\ \cup (v = k \text{ and } G_k^n = 1) \end{array} \right. \right\} \quad (\text{B.6})$$

$$R_n = \left\{ v \left| \begin{array}{l} (G_v^n = 0 \text{ and } G_{v+1}^n = 0) \\ \cup (v = 0 \text{ and } G_1^n = 0) \\ \cup (v = k \text{ and } G_k^n = 0) \end{array} \right. \right\} \quad (\text{B.7})$$

where G_v^n stands for the n -th bit of code G_v .

For example, sets R_1, R_2, R_3 and L_1, L_2, L_3 for Gray codes G_1 - G_8 are shown in Fig. B.2. In such a way, we can establish the rule that only two adjacent weights can be activated via (B.5). To see this, consider that if $\lambda_i > 0$ for $i = 4, 5$ and $\lambda_i = 0$ for other indices, we let $z_1 = 1, z_2 = 1, z_3 = 0$, which leads to the following constraint set:

$$\left\{ \begin{array}{l} \lambda_0 + \lambda_1 + \lambda_2 + \lambda_3 \leq 1 - z_1 = 0 \\ \lambda_5 + \lambda_6 + \lambda_7 + \lambda_8 \leq z_1 = 1 \\ \lambda_0 + \lambda_1 + \lambda_6 \leq 1 - z_2 = 0 \\ \lambda_3 + \lambda_4 + \lambda_8 \leq z_2 = 1 \\ \lambda_0 + \lambda_4 + \lambda_5 \leq 1 - z_3 = 1 \\ \lambda_2 + \lambda_7 + \lambda_8 \leq z_3 = 0 \\ \lambda_i \geq 0, \forall i, \sum_{i=0}^8 \lambda_i = 1 \end{array} \right.$$

Thus we can conclude that

$$\begin{aligned} \lambda_4 + \lambda_5 &= 1, \lambda_4 \geq 0, \lambda_5 \geq 0, \\ \lambda_0 &= \lambda_1 = \lambda_2 = \lambda_3 = \lambda_6 = \lambda_7 = \lambda_8 = 0 \end{aligned}$$

This mechanism can be interpreted as follows: $z_1 = 1$ enforces $\lambda_i = 0$, $i = 0, 1, 2, 3$ through set R_1 ; $z_2 = 1$ further enforces $\lambda_6 = 0$ through set R_2 ; finally, $z_3 = 0$ enforces $\lambda_7 = \lambda_8 = 0$ through set L_3 . Then the remaining weights λ_4 and λ_5 constitute the positive coefficients. In this regard, only $\log_2 8 = 3$ binary variables and $2 \log_2 8 = 6$ additional constraints are involved. Compared with formulation (B.3), the gray code can be regarded as extra branching operation enabled by problem structure, so the number of binary variables in expression (B.5) is greatly reduced in the case with a large value of k .

As a special case, consider the following problem

$$\min \left\{ \sum_i f_i(x_i) \mid x \in X \right\} \quad (\text{B.8})$$

where $f_i(x_i), i = 1, 2, \dots$ are convex univariate functions, and X is a polytope. This problem is convex but nonlinear. The DCOPF problem, a fundamental issue in power market clearing, is given in this form, in which $f_i(x_i)$ is a convex quadratic function. Although any local NLP algorithm can find the global optimal solution of (B.8), there are still reasons to seek approximated LP formulations. One is that problem (B.8) may be embedded in another optimization problem and serve as its constraint. This is a pervasive modeling paradigm to study the strategic behaviors and market powers of energy providers, where the electricity market is cleared according to a DCOPF, and delivered energy of generation companies and nodal electricity prices are extracted from the optimal primal variables and dual variables associating with power balancing constraints, respectively. An LP representation allows to exploit the elegant LP duality theory for further analysis, and helps characterize optimal solution through primal-dual or KKT optimality conditions. To this end, we can opt to solve the following LP

$$\begin{aligned} \min_{x, y, \lambda} \quad & \sum_i y_i \\ \text{s.t.} \quad & y_i = \sum_k \lambda_{ik} f_i(x_{ik}), \quad \forall i \\ & x_i = \sum_k \lambda_{ik} x_{ik}, \quad \forall i, \quad x \in X \\ & \lambda \geq 0, \quad \sum_k \lambda_{ik} = 1, \quad \forall i \end{aligned} \quad (\text{B.9})$$

where $x_{ik}, k = 1, 2, \dots$ are break points (constants) for variable x_i , and the associated weights are λ_{ik} . Because $f_i(x_i)$ are convex functions, the SOS_2 requirement on the weight variable λ is naturally met, so it is relaxed from the constraints.

B.1.2 Bivariate Continuous Nonlinear Function

Consider a continuous nonlinear function $f(x, y)$ in two variables x and y . The entire feasible region is partitioned into $M \times N$ disjoint sub-rectangles by $M + N + 2$ breakpoints $x_n, n = 0, 1, \dots, N$ and $y_m, m = 0, 1, \dots, M$, as illustrated in Fig. B.3, and the corresponding function values are $f_{mn} = f(x_m, y_n)$. By introducing a planar weighting coefficient matrix $\{\lambda_{mn}\}$ for each grid point that satisfies

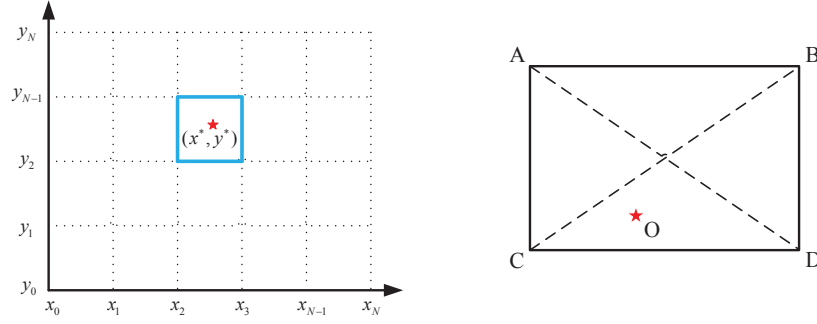


Fig. B.3 Breakpoints and active rectangle for PWL approximation.

$$\begin{aligned} \lambda_{mn} &\geq 0, \quad \forall m, \forall n \\ \sum_{m=0}^M \sum_{n=0}^N \lambda_{mn} &= 1 \end{aligned} \quad (\text{B.10a})$$

we can present any point (x, y) in the feasible region by a convex combination of the extreme points of the sub-rectangle it resides in:

$$\begin{aligned} x &= \sum_{m=0}^M \sum_{n=0}^N \lambda_{mn} x_n = \sum_{n=0}^N \left(\sum_{m=0}^M \lambda_{mn} \right) x_n \\ y &= \sum_{m=0}^M \sum_{n=0}^N \lambda_{mn} y_m = \sum_{m=0}^M \left(\sum_{n=0}^N \lambda_{mn} \right) y_m \end{aligned} \quad (\text{B.10b})$$

and its function value

$$f(x, y) = \sum_{m=0}^M \sum_{n=0}^N \lambda_{mn} f_{mn} \quad (\text{B.10c})$$

is also a convex combination of the function values at the corner points.

As we can see from Fig. B.3, in a valid representation, if (x^*, y^*) belongs to a sub-rectangle, only the weight parameter associated with the four corner points can be non-negative, while others should be forced at 0. In such a pattern, the sum of columns/rows of matrix $\Lambda = [\lambda_{mn}]$, $\forall m, n$, which remains a vector, should constitute an SOS_2 , and Λ is called a planar SOS_2 , which can be implemented via two SOS_1 constraints on the marginal weight vectors. In fact, at most three of the four corner points can be associated with uniquely determined non-negative weights. Consider point O and the active rectangle ABCD shown in Fig. B.3. The location of O can be expressed by a linear combination of the coordinates of corner points x_A, x_B, x_C, x_D associating with non-negative weights $\lambda_A, \lambda_B, \lambda_C, \lambda_D$ as:

$$x_O = \lambda_A x_A + \lambda_B x_B + \lambda_C x_C + \lambda_D x_D \quad (\text{B.11a})$$

In the first case

$$\lambda_A^1, \lambda_B^1, \lambda_C^1, \lambda_D^1 \geq 0, \lambda_A^1 + \lambda_B^1 + \lambda_C^1 + \lambda_D^1 = 1 \quad (\text{B.11b})$$

In the second case

$$\lambda_A^2, \lambda_C^2, \lambda_D^2 \geq 0, \lambda_B^2 = 0, \lambda_A^2 + \lambda_C^2 + \lambda_D^2 = 1 \quad (\text{B.11c})$$

In the third case

$$\lambda_B^3, \lambda_C^3, \lambda_D^3 \geq 0, \lambda_A^3 = 0, \lambda_B^3 + \lambda_C^3 + \lambda_D^3 = 1 \quad (\text{B.11d})$$

We use superscripts 1, 2, 3 to distinguish values of weights in different representations. According to Caratheodory theorem, the non-negative weights are uniquely determined in (B.11f) and (B.11g), and in the former (latter) case, we say ΔACD (ΔBCD) is activated or selected. Denote function values in these three cases by

$$f_1(x_O) = \lambda_A^1 f(x_A) + \lambda_B^1 f(x_B) + \lambda_C^1 f(x_C) + \lambda_D^1 f(x_D) \quad (\text{B.11e})$$

$$f_2(x_O) = \lambda_A^2 f(x_A) + \lambda_C^2 f(x_C) + \lambda_D^2 f(x_D) \quad (\text{B.11f})$$

$$f_3(x_O) = \lambda_B^3 f(x_B) + \lambda_C^3 f(x_C) + \lambda_D^3 f(x_D) \quad (\text{B.11g})$$

Suppose $f(x_A) < f(x_B)$, the plane defined by points B, C, D lies above that defined by points A, C, D, hence $f_2(x_O) < f_1(x_O) < f_3(x_O)$. If a smaller (larger) function value is in favor, then ΔACD (ΔBCD) will be activated at the optimal solution. Please bear in mind that as long as A, B, C, D are not in the same plane, $f_1(x_O)$ will be strictly less (greater) than $f_3(x_O)$ ($f_2(x_O)$). Therefore, (B.11e) will not become binding at the optimal solution, and the weights for active corners are uniquely determined. If rectangle ABCD is small enough, such a discrepancy can be neglected. Nonetheless, non-uniqueness of the corner weights has little injury on its application, because the optimal solution x_O and optimal value will be consistent with the original problem. The weights do not correspond to physical strategies that need to be deployed, and the linearization method can be considered as a black box to the decision maker, who provides function values at x_A, x_B, x_C, x_D and receives a unique solution x_O .

Detecting the active sub-rectangle that (x^*, y^*) resides in requires additional constraints on the weight parameter λ_{mn} . The aforementioned integer formulation is used to impose planar SOS_2 constraints. Let λ^n and λ^m be the aggregated weights for x and y , respectively, i.e.,

$$\begin{aligned}
\lambda^n &= \sum_{m=0}^M \lambda_{mn}, \quad \forall n \\
\lambda^m &= \sum_{n=0}^N \lambda_{mn}, \quad \forall m
\end{aligned} \tag{B.12}$$

which are also called the marginal weight vectors, and introduce the following constraints:

$$\text{For } x: \left\{ \begin{array}{l} \sum_{n \in L_k^1} \lambda^n \leq z_k^1 \\ \sum_{n \in R_k^1} \lambda^n \leq 1 - z_k^1 \\ z_k^1 \in \{0, 1\} \end{array} \right\}, \quad \forall k \in K_1 \tag{B.13}$$

$$\text{For } y: \left\{ \begin{array}{l} \sum_{m \in L_k^2} \lambda^m \leq z_k^2 \\ \sum_{m \in R_k^2} \lambda^m \leq 1 - z_k^2 \\ z_k^2 \in \{0, 1\} \end{array} \right\}, \quad \forall k \in K_2 \tag{B.14}$$

where L_k^1 , L_k^2 and R_k^2 , R_k^1 are index sets of weights λ^n and λ^m , K_1 and K_2 are index sets of binary variables. The dichotomy sequences $\{L_k^1, R_k^1\}_{k \in K_1}$ and $\{L_k^2, R_k^2\}_{k \in K_2}$ constitute a branching scheme on the indices of weights, such that constraints (B.13) and (B.14) would guarantee that at most two adjacent elements of λ^n and λ^m can take strictly positive values, so as to detect the active sub-rectangle. In this approach, the required number of binary variables is $\lceil \log_2 M \rceil + \lceil \log_2 N \rceil$. The construction of these index sets has been explained in the univariate case.

Likewise, the discussions for problems (B.8) and (B.9) can be easily extended if the objective function is the sum of bi-variate convex functions, implying that the planar SOS_2 condition is naturally met.

B.1.3 Approximation Error

This section answers a basic question: For a given function, how many intervals (break points) are needed to achieve certain error bound ε ? For the ease of understanding, we restrict our attention to univariate function, including the quadratic function $f(x) = ax^2$, and more generally, the continuous function $f(x)$ that is three times continuously differentiable. Let $\phi_f(x)$ be the PWL approximation for function $f(x)$ on $X = \{x | x_l \leq x \leq x_m\}$, the absolute maximum approximation error is defined by $\Delta = \max_{x \in X} |f(x) - \phi_f(x)|$.

First let us consider the quadratic function $f(x) = ax^2$, which has been thoroughly studied in [6]. The analysis is briefly introduced here. Choose an arbitrary interval $[x_{i-1}, x_i] \subset X$, the PWL approximation can be parameterize in a single variable $t \in [0, 1]$ as

$$\begin{aligned} x(t) &= x_{i-1} + t(x_i - x_{i-1}) \\ \phi_f(x(t)) &= ax_{i-1}^2 + at(x_i^2 - x_{i-1}^2) \end{aligned}$$

Clearly, $f(x(0)) = \phi_f(x(0)) = x_{i-1}^2$, $f(x(1)) = \phi_f(x(1)) = x_i^2$, and $\phi_f(x(t)) > f(x(t))$, $\forall t \in (0, 1)$. The maximal approximation error in the interval must be found at a critical point which satisfies

$$\begin{aligned} & \frac{d}{dt} (\phi_f(x(t)) - f(x(t))) \\ &= \frac{d}{dt} a [x_{i-1}^2 + t(x_i^2 - x_{i-1}^2) - (x_{i-1} + t(x_i - x_{i-1}))^2] \\ &= \frac{d}{dt} a(x_i - x_{i-1})^2(t - t^2) \\ &= a(x_i - x_{i-1})^2(1 - 2t) \\ &= 0 \Rightarrow t = \frac{1}{2} \end{aligned}$$

implying that $x(1/2)$ is always a critical point where the approximation error reaches maximum, regardless of the partition of intervals, and the error is given by

$$\begin{aligned} \Delta &= \phi \left(x \left(\frac{1}{2} \right) \right) - f \left(x \left(\frac{1}{2} \right) \right) \\ &= a \left[x_{i-1}^2 + \frac{1}{2}(x_i^2 - x_{i-1}^2) - \frac{1}{4}(x_{i-1} + x_i)^2 \right] \\ &= \frac{a}{4}(x_i - x_{i-1})^2 \end{aligned}$$

which is quadratic in the length of the interval and independent of its location. In this regard, the intervals must be evenly distributed with equal length in order to get the best performance. If X is divided into n intervals, the absolute maximum approximation error is

$$\Delta = \frac{a}{4n^2}(x_m - x_l)^2$$

Therefore, for a given tolerance ϵ , the number of intervals should satisfy

$$n \geq \sqrt{\frac{a}{\epsilon} \frac{x_m - x_l}{2}}$$

For quadratic function $f(x) = ax^2$, coefficient a determines its second-order derivative. For more general situations, above discussion implies that

the number of intervals needed to perform a PWL approximation for function $f(x)$ may depend on its second-order derivative. This problem has been thoroughly studied in [7]. The conclusion is: for a three times continuously differentiable $f(x)$ in the interval $[x_l, x_m]$, the optimal number of segments $s(\varepsilon)$ under given error tolerance ε can be selected as

$$s(\varepsilon) \propto \frac{c}{\sqrt{\varepsilon}}, \quad \varepsilon \rightarrow 0^+$$

where

$$c = \frac{1}{4} \int_{x_l}^{x_m} \sqrt{|f''(x)|}$$

The conclusion still holds if $\sqrt{|f''(x)|}$ has integrable singularities at the end-points.

B.2 Linear Formulation of Product Terms

Product of two variables, or a bilinear term, naturally arises in optimization models from various disciplines. For one example, in economic studies, if the price c and the quantity q of a commodity are variables, then the cost cq would be a bilinear term. For another, in circuit analysis, if both of the voltage v and the current i are variables, then the electric power vi would be a bilinear term. Bilinear terms are non-convex. Throughout history, linearizing bilinear terms using linear constraints and integer variables is a frequently used technique in optimization community. This section presents several techniques for the central question of product linearization: how to enforce constraint $z = xy$, depending on the types of x and y .

B.2.1 Product of Two Binary Variables

If $x \in \mathbb{B}$ and $y \in \mathbb{B}$, then $z = xy$ is equivalent to the following linear inequalities

$$\begin{aligned} 0 &\leq z \leq y \\ 0 &\leq x - z \leq 1 - y \\ x &\in \mathbb{B}, \quad y \in \mathbb{B}, \quad z \in \mathbb{B} \end{aligned} \tag{B.15}$$

It can be verified that if $x = 1$, $y = 1$, then $z = 1$ is achieved; if $x = 0$ or $y = 0$, then $z = 0$ is enforced, regardless of the value of y or x . This is equivalent to the requirement $z = xy$.

If $x \in \mathbb{Z}^+$ belongs to interval $[x^L, x^U]$, and $y \in \mathbb{Z}^+$ belongs to interval $[y^L, y^U]$, given the following binary expansion

$$\begin{aligned}
x &= x^L + \sum_{k=1}^{K_1} 2^{k-1} u_k \\
y &= y^L + \sum_{k=1}^{K_2} 2^{k-1} v_k
\end{aligned} \tag{B.16}$$

where $K_1 = \lceil \log_2(x^U - x^L) \rceil$, $K_2 = \lceil \log_2(y^U - y^L) \rceil$. To develop a vector expression, define vectors $b^1 = [2^0, 2^1, \dots, 2^{K_1-1}]$, $u = [u_1, u_2, \dots, u_{K_1}]^T$, $b^2 = [2^0, 2^1, \dots, 2^{K_2}]$, $v = [v_1, v_2, \dots, v_{K_2}]^T$, and matrices $B = (b^1)^T b^2$, $z = uv^T$, then

$$xy = x^L y^L + x^L b^2 v + y^L b^1 u + \langle B, z \rangle$$

where product matrix $z = uv^T$, and $\langle B, z \rangle = \sum_i \sum_j B_{ij} z_{ij}$. Relation among u , v , and z can be linearized via equation (B.15) element-wise. Its compact form is given by

$$\begin{aligned}
\mathbf{0}^{K_1 \times K_2} &\leq z \leq \mathbf{1}^{K_1 \times 1} v^T \\
\mathbf{0}^{K_1 \times K_2} &\leq u \mathbf{1}^{1 \times K_2} - z \leq \mathbf{1}^{K_1 \times K_2} - \mathbf{1}^{K_1 \times 1} v^T \\
u &\in \mathbb{B}^{K_1 \times 1}, \quad v \in \mathbb{B}^{K_2 \times 1}, \quad z \in \mathbb{B}^{K_1 \times K_2}
\end{aligned} \tag{B.17}$$

B.2.2 Product of Integer and Continuous Variables

We consider the binary-continuous case. If $x \in \mathbb{R}$ belongs to interval $[x^L, x^U]$, and $y \in \mathbb{B}$, and then $z = xy$ is equivalent to the following linear inequalities

$$\begin{aligned}
x^L y &\leq z \leq x^U y \\
x^L (1 - y) &\leq x - z \leq x^U (1 - y) \\
x &\in \mathbb{R}, \quad y \in \mathbb{B}, \quad z \in \mathbb{R}
\end{aligned} \tag{B.18}$$

It can be verified that if $y = 0$, then z is enforced to be 0 and $x^L \leq x \leq x^U$ is naturally met; if $y = 1$, then $z = x$ and $x^L \leq z \leq x^U$ must be satisfied, indicating the same relationship on x , y , and z . As for the integer-continuous case, the integer variable can be represented as (B.16) using binary variables, yielding a linear combination of binary-continuous products.

It should be mentioned that the upper bound x^U and the lower bound x^L are crucial for creating linearization inequalities. If explicit bounds are not available at hand, one can incorporate a constant M that is big enough. The value of M will have a notable impact on the computation time. To enhance efficiency, a desired value should be the minimal M that ensures that inequality $-M \leq x \leq M$ never becomes binding at optimum, as it leads to the strongest bound if integrality of binary variables is neglected, expediting the converge of the branch-and-bound procedure. However, such

a value is generally unclear before we solve the problem. Nevertheless, we do not actually need to find the smallest value M^* . Any $M \geq M^*$ produces the same optimal solution and is valid for linearization. Please bear in mind that an over-large M not only deteriorates the computation time, but also cause numeric instability due to a large conditional number. So a proper tradeoff must be made between efficiency and accuracy. A proper M can be determined from estimating the bound of x from certain heuristics, which is problem-dependent.

B.2.3 Product of Two Continuous Variables

If $x \in \mathbb{R}$ belongs to interval $[x^L, x^U]$, and $y \in \mathbb{R}$ belongs to interval $[y^L, y^U]$, there are three options for linearizing their product xy . The first one considers $z = xy$ as a bivariate function $f(x, y)$, and applies the planar SOS_2 method in Sect. B.1.2. The second one discretizes y , for example, as follows

$$\begin{aligned} y &= y^L + \sum_{k=1}^K 2^{k-1} u_k \Delta y \\ \Delta y &= \frac{y^U - y^L}{2^K}, \quad u_k \in \mathbb{B}, \quad \forall k \end{aligned} \tag{B.19}$$

and

$$xy = xy^L + \sum_{k=1}^K 2^{k-1} v_k \Delta y \tag{B.20}$$

where $v_k = u_k x$ can be linearized through equation (B.18) as

$$\begin{aligned} x^L u_k &\leq v_k \leq x^U u_k, \quad \forall k \\ x^L (1 - u_k) &\leq x - v_k \leq x^U (1 - u_k), \quad \forall k \\ x &\in \mathbb{R}, \quad u_k \in \mathbb{B}, \quad \forall k, \quad v_k \in \mathbb{R}, \quad \forall k \end{aligned} \tag{B.21}$$

In practical problems, bilinear terms often appear as the inner production of two vectors. For convenience, we present the compact linearization of $x^T y$ via binary expansion. Let y be the candidate vector variable to be discretized; perform (B.19) on each element of y

$$y_j = y_j^L + \sum_{k=1}^K 2^{k-1} u_{jk} \Delta y_j, \quad \forall j$$

and thus

$$x_j y_j = x_j y_j^L + \sum_{k=1}^K 2^{k-1} v_{jk} \Delta y_j, \quad \forall j, \quad v_{jk} = u_{jk} x_j, \quad \forall j, \forall k$$

Relation $v_{jk} = u_{jk} x_j$ can be expressed via linear constraints

$$x_j^L u_{jk} \leq v_{jk} \leq x_j^U u_{jk}, \quad x_j^L (1 - u_{jk}) \leq x_j - v_{jk} \leq x_j^U (1 - u_{jk}), \quad \forall j, \forall k$$

Denote by V and U are matrix variables consisting of v_{jk} and u_{jk} , respectively; 1_K stands for all-one column vector with a dimension of K ; Δ_Y is a diagonal matrix with Δy_j being non-zero entries; vector $\zeta = [2^0, 2^1, \dots, 2^{K-1}]$. Combining all above element-wise expressions together, we have the linear formulation of $x^T y$ in a compact matrix form

$$x^T y = x^T y^L + \zeta V^T \Delta y$$

in conjunction with

$$\begin{aligned} y &= y^L + \Delta_Y U \zeta^T \\ (x^L \cdot 1_K^T) \otimes U &\leq V \leq (x^U \cdot 1_K^T) \otimes U \\ (x^L \cdot 1_K^T) \otimes (1 - U) &\leq x \cdot 1_K^T - V \leq (x^U \cdot 1_K^T) \otimes (1 - U) \\ x &\in \mathbb{R}^J, \quad y \in \mathbb{R}^J, \quad U \in \mathbb{B}^{J \times K}, \quad V \in \mathbb{R}^{J \times K} \end{aligned} \tag{B.22}$$

where \otimes represents element-wise product of two matrices with the same dimension.

One possible drawback of this formulation is that the discretized variable is no longer continuous. The approximation accuracy can be improved by increasing the number of breakpoints without introducing too many binary variables, whose number is given by $\lceil \log_2(y^U - y^L) / \Delta y \rceil$. Furthermore, the variable to be discretized must have clear upper and lower bounds. This is not restrictive because decision variables of engineering problems are subject to physical operating limitations, such as the maximum and minimum output of a generator. Nevertheless, if x , for example, is unbounded in formulation, but the problem has a finite optimum, we can replace $x^U(x^L)$ in (B.21) with a large enough big-M parameter $M(-M)$, so that the true optimal solution remains feasible. It should be pointed out that the value of M may influence the computational efficiency of the equivalent MILP, as mentioned previously. The optimal choice of M in general cases remains an open problem, but there could be heuristic methods for specific instances. For example, if x stands for the marginal production cost, which is a dual variable whose bounds are unclear, one can alternatively determine a suitable bound from historical data or price forecast.

An alternative formulation for the second option deals with product term $x f(y)$, and xy is a special case when $f(y) = y$. By performing the piecewise constant approximation (B.4) on function $f(y)$, the product becomes $xy =$

$\sum_{s=1}^{S-1} x\theta_s y_s$, where y_s is constant, x is continuous, and θ_s is binary. The products $x\theta_s$, $s = 1, \dots, S-1$ can be readily linearized via the method in B.2.2. In this approach, the continuity of x and y are retained. However, the number of binary variables in the piecewise constant approximation for $f(y)$ grows linearly in the number of samples on y .

The third one converts the product into a separable form, and then performs PWL approximation for univariate nonlinear functions. To see this, consider a bilinear term xy . Introduce two continuous variables u and v defined as follows

$$\begin{aligned} u &= \frac{1}{2}(x + y) \\ v &= \frac{1}{2}(x - y) \end{aligned} \tag{B.23}$$

Now we have

$$xy = u^2 - v^2 \tag{B.24}$$

In (B.24), u^2 and v^2 are univariate nonlinear functions, and can be approximated by the PWL method presented in Sect. B.1.1. Furthermore, if $x_l \leq x \leq x_u$, $y_l \leq y \leq y_u$, then the lower and upper bounds of u and v are given by

$$\begin{aligned} \frac{1}{2}(x_l + y_l) &\leq u \leq \frac{1}{2}(x_u + y_u) \\ \frac{1}{2}(x_l - y_u) &\leq v \leq \frac{1}{2}(x_u - y_l) \end{aligned}$$

Formulation (B.24) has a connotative advantage. If xy appears in the objective function which is to be minimized and is not involved in constraints, we only need to approximate v^2 because u^2 is convex and $-v^2$ is concave. The minimum amount of binary variables in this method is a logarithmic function in the number of break points, as explained in Sect. B.1.1.

The bilinear term xy can be replaced by a single variable z in the following situation: 1) if the lower bounds x_l and y_l are nonnegative; 2) either x or y is not referenced anywhere else except in xy . For instance, y is such a variable, then the bilinear term xy can be replaced by variable z and constraint $xy_l \leq z \leq xy_u$. Once the problem is solved, y can be recovered by $y = z/x$ if $x > 0$, and the inequality constraint on z guarantees $y \in [y_l, y_u]$; otherwise if $x = 0$, then y is undetermined and has no impact on the optimum.

B.2.4 Monomial of Binary Variables

Previous cases discuss linearizing the product of two variables. Now we consider a binary monomial with n variables

$$z = x_1 x_2 \cdots x_n, \quad x_i \in \{0, 1\}, \quad i = 1, 2, \dots, n \tag{B.25}$$

Clearly, this monomial takes a binary value. Since the product of two binary can be expressed by a single one in light of (B.15), the monomial can be linearized recursively. Nevertheless, by making full use of the binary property of z , a smarter and concise way to represent (B.25) is given by

$$z \in \{0, 1\} \quad (\text{B.26})$$

$$z \leq \frac{x_1 + x_2 + \cdots + x_n}{n} \quad (\text{B.27})$$

$$z \geq \frac{x_1 + x_2 + \cdots + x_n - n + 1}{n} \quad (\text{B.28})$$

If at least one of x_i is equal to 0, because $\sum_{i=1}^n x_i - n + 1 \leq 0$, (B.28) becomes redundant; moreover, $\sum_{i=1}^n x_i/n \leq 1 - 1/n$, which removes $z = 1$ from the feasible region, so z will take a value of 0; otherwise, if all x_i are equal to 1, $\sum_{i=1}^n x_i/n = 1$, and the right-hand side of (B.28) is $1/n$, which removes $z = 0$ from the feasible region. Hence z is forced to be 1. In conclusion, linear constraints (B.26)-(B.28) have the same effect as (B.25).

In view of the above transformation technique, a binary polynomial program can always be reformulated as a binary linear program. Moreover, if a single continuous variable appears in the monomial, the problem can be reformulated as an MILP.

B.2.5 Product of Functions in Integer Variables

First, let us consider $z = f_1(x_1)f_2(x_2)$, where decision variables are positive integers, i.e., $x_i \in \{d_{i,1}, d_{i,2}, \dots, d_{i,r_i}\}, i = 1, 2$. Without particular tricks, f_1 and f_2 can be expressed as

$$\begin{aligned} f_1 &= \sum_{j=1}^{r_1} f_1(d_{1,j})u_{1,j}, \quad u_{1,j} \in \{0, 1\}, \quad \sum_{j=1}^{r_1} u_{1,j} = 1 \\ f_2 &= \sum_{j=1}^{r_2} f_2(d_{2,j})u_{2,j}, \quad u_{2,j} \in \{0, 1\}, \quad \sum_{j=1}^{r_2} u_{2,j} = 1 \end{aligned}$$

and the product of two binary variables can be linearized via (B.15). Above formulation introduces a lot of intermediary binary variables, and is not propitious to represent a product with more functions recursively.

Ref. [8] suggests another choice

$$z = \sum_{i=1}^{r_2} f_2(d_{2,i})\sigma_{2,i}, \quad \sum_{i=1}^{r_2} \sigma_{2,i} = f_1(x_1), \quad \sigma_{2,i} = f_1(x_1)u_{2,i} \quad (\text{B.29})$$

where $u_{2,j} \in \{0, 1\}$ and $\sum_{j=1}^{r_2} u_{2,j} = 1$. Although $f_1(x_1)u_{2,j}$ remains nonlinear because of decision variable x_1 , (B.29) can be used to linearize a product with more than two nonlinear functions.

To see this, Denote by $z_1 = f_1(x_1)$, $z_i = z_{i-1}f_i(x_i)$, $i = 2, \dots, n$; integer variable $x_i \in \{d_{i,1}, d_{i,2}, \dots, d_{i,r_i}\}$, $f_i(x_i) > 0$, $i = 1, \dots, n$. By using (B.29), z_i , $i = 1, \dots, n$ have the following expressions [8]

$$\begin{aligned}
z_1 &= \sum_{j=1}^{r_1} f_1(d_{1,j})u_{1,j} \\
z_2 &= \sum_{j=1}^{r_2} f_2(d_{2,j})\sigma_{2,j}, \quad \sum_{i=1}^{r_2} \sigma_{2,j} = z_1, \quad \dots \\
z_n &= \sum_{j=1}^{r_n} f_n(d_{n,j})\sigma_{n,j}, \quad \sum_{i=1}^{r_n} \sigma_{n,j} = z_{n-1} \\
0 &\leq z_{i-1} - \sigma_{i,j} \leq \bar{z}_{i-1}(1 - u_{i,j}) \quad \left. \vphantom{\sum_{j=1}^{r_i}} \right\} \quad j = 1, \dots, r_i, \\
0 &\leq \sigma_{i,j} \leq \bar{z}_{i-1}u_{i,j}, \quad u_{i,j} \in \{0, 1\} \quad \left. \vphantom{\sum_{j=1}^{r_i}} \right\} \quad i = 2, \dots, n \\
x_i &= \sum_{j=1}^{r_i} d_{i,j}u_{i,j}, \quad \sum_{j=1}^{r_i} u_{i,j} = 1, \quad i = 1, 2, \dots, n
\end{aligned} \tag{B.30}$$

In (B.30), the number of binary variables is $\sum_{i=1}^n r_i$, and grow linearly in the dimension of x and the interval length of each x_i . To reduce the number of auxiliary binary variable $u_{i,j}$, the dichotomy procedure in Sect. B.1.1 for SOS2 can be applied, which is discussed in [8].

B.2.6 Log-sum Functions

We consider log-sum function $\log(x_1 + x_2 + \dots + x_n)$, which arises from solving a signomial geometric programming problem. The basic element in such a problem has a form of

$$c_k \prod_{j=1}^l y_j^{a_{jk}} \tag{B.31}$$

where $y_j > 0$, c_k is a constant, and $a_{jk} \in \mathbb{R}$. Non-integer value of a_{jk} makes signomial geometric programming problem even harder than polynomial programs. Under some variable transformation, the non-convexity of a signomial geometric program can be concentrated in some log-sum functions [9]. In view of the form in (B.31), we discuss log-sum function in Sect. B.2.

We aim to represent function $\log(x_1 + x_2 + \dots + x_n)$ in terms of $\log x_1$, $\log x_2$, \dots , $\log x_n$. Following the method in [9], define a univariate function

$F(X) = \log(1 + e^X)$ and let $X_i = \log x_i$, $\Gamma_i = \log(x_1 + \cdots + x_i)$, $i = 1, \dots, n$. The relation between X_i and Γ_i can be revealed. Because

$$\begin{aligned} F(X_{i+1} - \Gamma_i) &= \log \left(1 + e^{\log x_{i+1} - \log(x_1 + \cdots + x_i)} \right) \\ &= \log \left(1 + \frac{x_{i+1}}{x_1 + \cdots + x_i} \right) = \Gamma_{i+1} - \Gamma_i \end{aligned}$$

By stipulating $W_i = X_{i+1} - \Gamma_i$, we have the following recursive equations

$$\begin{aligned} \Gamma_{i+1} &= \Gamma_i + F(W_i), \quad i = 1, \dots, n-1 \\ W_i &= X_{i+1} - \Gamma_i, \quad i = 1, \dots, n-1 \end{aligned} \tag{B.32}$$

Function $F(W_i)$ can be linearized using the method in Sect. B.1.1. Based on this technique, an outer-approximation approach is proposed in [9] to solve signomial geometric programming problem via MILP.

B.3 Other Frequently used Formulations

B.3.1 Minimum Values

Let x_1, x_2, \dots, x_n be continuous variables with known lower bound x_i^L and upper bound x_i^U , and $L = \min\{x_1^L, x_2^L, \dots, x_n^L\}$, then their minimum $y = \min\{x_1, x_2, \dots, x_n\}$ can be expressed via linear constraints

$$\begin{aligned} x_i^L &\leq x_i \leq x_i^U, \quad \forall i \\ y &\leq x_i, \quad \forall i \\ x_i - (x_i^U - L)(1 - z_i) &\leq y, \quad \forall i \\ z_i &\in \mathbb{B}, \quad \forall i, \quad \sum_i z_i = 1 \end{aligned} \tag{B.33}$$

The second inequality guarantees $y \leq \min\{x_1, x_2, \dots, x_n\}$; in addition, if $z_i = 1$, then $y \geq x_i$, hence y achieves the minimal value of $\{x_i\}$. According to the definition of L , $x_i - y \leq x_i^U - L, \forall i$ holds, thus the third inequality is inactive for the remaining $n - 1$ variables with $z_i = 0$.

B.3.2 Maximum Values

Let x_1, x_2, \dots, x_n be continuous variables with known lower bound x_i^L and upper bound x_i^U , and $U = \max\{x_1^U, x_2^U, \dots, x_n^U\}$, then their maximum $y = \max\{x_1, x_2, \dots, x_n\}$ can be expressed via linear constraints

$$\begin{aligned}
x_i^L &\leq x_i \leq x_i^U, \forall i \\
y &\geq x_i, \forall i \\
x_i + (U - x_i^L)(1 - z_i) &\geq y, \forall i \\
z_i &\in \mathbb{B}, \forall i, \sum_i z_i = 1
\end{aligned} \tag{B.34}$$

The second inequality guarantees $y \geq \max\{x_1, x_2, \dots, x_n\}$; in addition, if $z_i = 1$, then $y \leq x_i$, hence y achieves the maximal value of $\{x_i\}$. According to the definition of U , $y - x_i \leq U - x_i^L, \forall i$ holds, thus the third inequality is inactive for the remaining $n - 1$ variables with $z_i = 0$.

B.3.3 Absolute Values

Suppose $x \in \mathbb{R}$ and $|x| \leq U$, the absolute value function $y = |x|$, which is nonlinear, can be expressed via PWL function as

$$\begin{aligned}
0 &\leq y - x \leq 2Uz, \quad U(1 - z) \geq x \\
0 &\leq y + x \leq 2U(1 - z), \quad -Uz \leq x \\
-U &\leq x \leq U, \quad z \in \mathbb{B}
\end{aligned} \tag{B.35}$$

When $x > 0$, the first line yields $z = 0$ and $y = x$, while the second line is inactive. When $x < 0$, the second line yields $z = 1$ and $y = -x$, while the first line is inactive. When $x = 0$, either $z = 0$ or $z = 1$ gives $y = 0$. In conclusion, (B.35) has the same effect as $y = |x|$.

B.3.4 Linear Fractional of Binary Variables

A linear fractional of binary variables takes the form of

$$\frac{a_0 + \sum_{i=1}^n a_i x_i}{b_0 + \sum_{i=1}^n b_i x_i} \tag{B.36}$$

We assume $b_0 + \sum_{i=1}^n b_i x_i \neq 0$ for all $x \in \{0, 1\}^n$. Define a new continuous variable

$$y = \frac{1}{b_0 + \sum_{i=1}^n b_i x_i} \tag{B.37}$$

The lower bound and upper bound of y can be easily computed. Then the linear fractional shown in (B.36) can be replaced with a linear expression

$$a_0 y + \sum_{i=1}^n a_i z_i \quad (\text{B.38})$$

with constraints

$$b_0 y + \sum_{i=1}^n b_i z_i = 1 \quad (\text{B.39})$$

$$z_i = x_i y, \quad \forall i \quad (\text{B.40})$$

where (B.40) describes a product of a binary variable and a continuous variable, which can be linearized through equation (B.18).

B.3.5 Disjunctive Inequalities

Let $\{P^i\}, i = 1, 2, \dots, m$ be a finite set of bounded polyhedra. Disjunctive inequalities usually arise when the solution space is characterized by the union $S = \cup_{i=1}^m P^i$ of these polyhedra. Unlike intersection operator which preserves convexity, disjunctive inequalities form a non-convex region. It can be represented by MILP model using binary variables. We introduce three emblematic methods.

1. Big-M formulation

The hyperplane representations of polyhedra are given by $P^i = \{x \in \mathbb{R}^n | A^i x \leq b^i\}, i = 1, 2, \dots, m$. By introducing binary variables $z_i, i = 1, 2, \dots, m$, an MILP formulation for S can be written as

$$\begin{aligned} A^i x &\leq b^i + M^i(1 - z_i), \quad \forall i \\ z_i &\in \mathbb{B}, \quad \forall i, \quad \sum_{i=1}^m z_i = 1 \end{aligned} \quad (\text{B.41})$$

where M^i is a vector such that when $z_i = 0$, $A^i x \leq b^i + M^i$ holds. To show the impact of the value of M on the tightness of formulations (B.41) when integrality constraints $z_i \in \mathbb{B}, \forall i$ are relaxed as $z_i \in [0, 1], \forall i$, we contrivedly construct 4 polyhedra in \mathbb{R}^2 , which are depicted in Fig. B.4. The continuous relaxations of (B.41) with different values of M are illustrated in the same graph, showing that the smaller the value of M , the tighter the relaxation of (B.41).

From a computational perspective, the element in M should be as small as possible, because a huge constant without any insights about problem data will feature a bad conditional number. Furthermore, the continuous relaxation of MILP model will be very weak, resulting in poor objective value bounds and excessive branch-and-bound computation. The goal of big-M parameter selection is to create a model whose continuous relaxation is

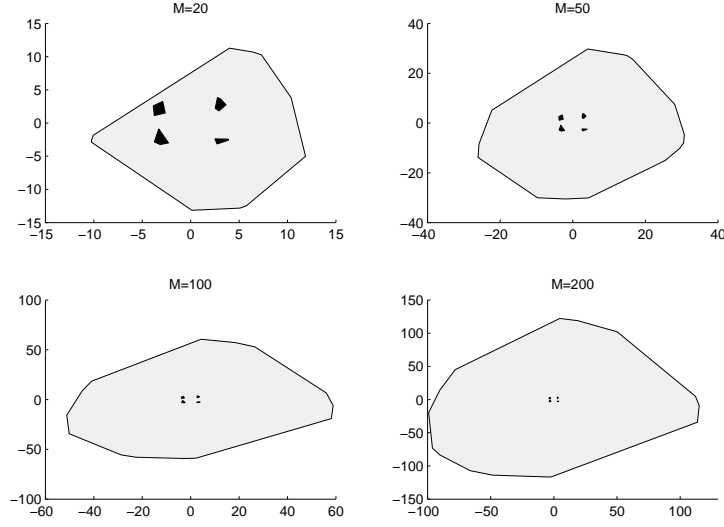


Fig. B.4 Big-M formulation and their relaxed regions.

close to the convex hull of the original constraint, i.e. the smallest convex set that contains the original feasible region. A possible selection of the big-M parameter is

$$M_l^i = \left(\max_{j \neq i} M_l^{ij} \right) - b_l^i \quad (\text{B.42})$$

$$M_l^{ij} = \max_x \{ [A^i x]_l : A^j x \leq b^j \}$$

where subscript l stands for the l -th element of a vector or l -th row of a matrix. As polyhedron P^i are bounded, all bound parameters in (B.42) are well defined.

However, even the tightest big-M parameter will yield a relaxed solution space that is generally larger than the convex hull of the original feasible set. In many applications, good variable bounds can be estimated from certain heuristic methods which explore specific problem data and structure.

2. Convex hull formulation

Let $\text{vert}(P^i) = \{v_l^i\}, l = 1, 2, \dots, L^i$ denote sets of vertices of polyhedra $\{P^i\}, i = 1, 2, \dots, m$, where L^i is the number of vertices of P^i . The set of extreme rays is empty since P^i is bounded. By introducing binary variables $z_i, i = 1, 2, \dots, m$, an MILP formulation for S is given by

$$\begin{aligned}
\sum_{i=1}^m \sum_{l=1}^{L^i} \lambda_l^i v_l^i &= x \\
\sum_{l=1}^{L^i} \lambda_l^i &= z_i, \quad \forall i \\
\lambda_l^i &\geq 0, \quad \forall i, \quad \forall l \\
z_i &\in \mathbb{B}, \quad \forall i, \quad \sum_{i=1}^m z_i = 1
\end{aligned} \tag{B.43}$$

Formulation (B.43) does not rely on manually supplied parameter. Instead, it requires enumerating all extreme points of polyhedra P^i . Although the vertex representation and hyperplane representation of a polyhedron are interchangeable, given the fact that vertex enumeration is time consuming for high-dimensional polyhedra, (B.43) is useful only if P^i are originally represented by extreme points.

3. Lifted formulation

A smarter formulation exploits the fact that bounded polyhedra P^i share the same recession cone $\{0\}$, i.e., equation $A^i x = 0$ has no non-zero solutions. Otherwise, suppose $A^i x^* = 0$, $x^* \neq 0$, and $y \in P^i$, then $y + \lambda x^* \in P^i$, $\forall \lambda > 0$, because $A^i(y + \lambda x^*) = A^i y \leq b^i$. As a result, P^i is unbounded. Bearing this in mind, an MILP formulation for S is given by

$$\begin{aligned}
A^i x^i &\leq b^i z^i, \quad \forall i \\
\sum_{i=1}^m x^i &= x \\
z_i &\in \mathbb{B}, \quad \forall i \\
\sum_{i=1}^m z_i &= 1
\end{aligned} \tag{B.44}$$

Formulation (B.44) is also parameter-free. Since it incorporates additional continuous variable for each polytope, we call it a lifted formulation. It is easy to see that the feasible region of x is the union of P^i : if $z_i = 0$, $x^i = 0$ as analyzed before; otherwise, if $z_i = 1$, $x = x^i \in P^i$.

4. Complementarity and slackness condition

Complementarity and slackness condition naturally arises in the KKT optimality condition of a mathematical programming problem, an equilibrium problem, a hierarchical optimization problem, and so on. It is a quintessential law to characterize the logic condition under which a rational decision-making progress must obey. Here we pay attention to the linear case and equivalent

MILP formulation, because nonlinear cases give rise to MINLPs, which are challenging to solve and not superior from the computational point of view.

A linear complementarity and slackness condition can be written as

$$0 \leq y \perp Ax - b \geq 0 \quad (\text{B.45})$$

where vectors x and y are decision variables; A and b are constant coefficients with compatible dimensions; notation \perp stands for the orthogonality of two vectors. In fact, (B.45) encompasses the following nonlinear constraints in traditional form

$$y \geq 0, Ax - b \geq 0, y^T(Ax - b) = 0 \quad (\text{B.46})$$

In view of the non-negativeness of y and $Ax - b$, the orthogonality condition is equivalent to the element-wise logic form $y_i = 0$ or $a_i x - b_i = 0$, $\forall i$, where a_i is the i -th row of A ; in other words, at most one of y_i and $a_i x - b_i$ can take a strictly positive value, implying that the feasible region is either the slice $y_i = 0$ or the slice $a_i x - b_i = 0$. Therefore, (B.45) can be regarded as a special case of the disjunctive constraints.

In practical application, (B.45) usually serves as constraints in an optimization problem. For example, in a sequential decision making or a linear bilevel program, the KKT condition of the lower-level LP appears in the form of (B.45), which is the constraint of the upper-level optimization problem. The main computation challenge arises from the orthogonality condition, which is nonlinear and non-convex, and violates the linear independent constraint qualification, see Appendix D.3 for an example. Nonetheless, in view of the switching logic between y_i and $a_i x - b_i$, we can introduce a binary variable z_i to select which slice is active [10]

$$\begin{aligned} 0 &\leq a_i x - b_i \leq M z_i, \quad \forall i \\ 0 &\leq y_i \leq M(1 - z_i), \quad \forall i \end{aligned} \quad (\text{B.47})$$

where M is a large enough constant. According to (B.47), if $z_i = 0$, then $(Ax - b)_i = 0$ must hold, and the second inequality is redundant; otherwise, if $z_i = 1$, then we have $y_i = 0$, and the first inequality becomes redundant. (B.47) can be written in a compact form as

$$\begin{aligned} 0 &\leq Ax - b \leq Mz \\ 0 &\leq y \leq M(1 - z) \end{aligned} \quad (\text{B.48})$$

It is worth mentioning that the big-M parameter M has a notable impact on the feasible region of the relaxed problem as well as the computational efficiency of the MILP model, as illustrated in Fig. B.4. One should make sure that (B.48) would not remove the optimal solution from the feasible set. If both x and y have clear bounds, then M can be easily estimated;

otherwise, we may prudently employ a large M , at the cost of sacrificing the computational efficiency.

Furthermore, if we are aiming to solve (B.45) without an objective function and other constraints, such a problem is called a linear complementarity problem (under some proper transformation), for which we can build parameter-free MILP models. More details can be found in Appendix D.4.2.

B.3.6 Logical Conditions

Logical conditions are associated with indicator constraints with a statement like “if event A then event B”. An event can be described in many ways. For example, a binary variable $a = 1$ can stand for event A happens, and otherwise $a = 0$; a point x belongs to a set X can denote a system is under secure operating condition, and otherwise $x \notin X$. In view of this, the disjunctive constraints discussed above is a special case of logical condition. In this section, we expatiate on how some usual logical conditions can be expressed via linear constraints. Let A, B, C, \dots associated with binary variables a, b, c, \dots represent events. Main results for linearizing typical logical conditions are summarized in B.1 [11].

Table B.1 Linear form of some typical logic conditions

If A then B	$b \geq a$
Not B	$1 - b$
If A then not B	$a + b \leq 1$
If not A then B	$a + b \geq 1$
A if and only if B	$a = b$
If A then B and C	$b + c \geq 2a$
If A then B or C	$b + c \geq a$
If B or C then A	$2a \geq b + c$
If B and C then A	$a \geq b + c - 1$
If M or more of N events then A	$(N - M + 1)a \geq b + c + \dots - M + 1$

Logical AND is formulated as a function of two binary inputs. Specifically, $c = a \text{ AND } b$ can be expressed as $c = \min\{a, b\}$ or $c = ab$. The former one can be linearized via (B.33) and the latter one through (B.15), and both of them renders

$$c \leq a, c \leq b, c \geq a + b - 1, c \geq 0 \quad (\text{B.49})$$

For the case with multiple binary inputs, i.e., $c = \min\{c_1, \dots, c_n\}$, or $c = \prod_{i=1}^n c_i$, (B.49) can be generalized as

$$c \leq c_i, \forall i, c \geq \sum_{i=1}^n c_i - n + 1, c \geq 0 \quad (\text{B.50})$$

Logical OR is formulated as a function of two binary inputs, i.e., $c = \max\{a, b\}$, which can be linearized via (B.34), yielding

$$c \geq a, c \geq b, c \leq a + b, c \leq 1 \quad (\text{B.51})$$

For the case with multiple binary inputs, i.e., $c = \max\{c_1, \dots, c_n\}$, (B.51) can be generalized as

$$c \geq c_i, \forall i, c \leq \sum_{i=1}^n c_i, c \leq 1 \quad (\text{B.52})$$

B.4 Further Reading

Throughout the half-century long research and development, MILP has become an indispensable and unprecedentedly powerful modeling tool in mathematics and engineering, thanks to the advent of efficient solvers that encapsulate many state-of-the-art techniques [12]. This chapter aims to provide an overview on formulation recipes that transform complicated conditions into MILPs, so as to take full advantages of off-the-shelf solvers. The paradigm is able to deal with a fairly broad class of hard optimization problems.

Readers who are interested in the strength of MILP model, may find in-depth discussions in [13] and references therein. For those interested in the PWL approximation of nonlinear functions, we refer to [14, 15, 16] and references therein, for various models and methods. The most promising one may be the convex combination model with a logarithmic number of binary variables, whose implementation has been thoroughly discussed in [3, 4, 5]. For those who are interested in the polyhedral study of single-term bilinear sets and MILP based methods for bilinear programs may find extensive information in [17, 18] and references therein. For those who need more knowledge about mathematical program with disjunctive constraints, in which constraint activity is controlled by logical conditions, we recommend [19]; specifically, the choice of big-M parameter is discussed in [20]. For those who wish to learn more about integer programming techniques, we refer to [21] for the formulation of union of polyhedra, [22] for the representability of MILP, and [23, 24] for the more general mixed-integer conic programming as well as its duality theory. To the best of our knowledge, dissertation [25] launches the most comprehensive and in-depth study on MILP approximation of non-convex optimization problems. State-of-the-art MILP formulations which balance problem size, strength, and branching behavior are developed and compared, including those mentioned above. The discussions in [25] offer insights on designing efficient MILP models that perform extremely well in practice, despite of their theoretically non-polynomial complexity in the worst case.

References

1. Chen, D.S., Batson, R.G., Dang, Y. Applied Integer Programming: Modeling and Solution. John Wiley & Sons Hoboken, NJ (2011)
2. Beale, E., Forrest, J.: Global optimization using special ordered sets. *Math. Program.* 10(1): 52-69 (1976)
3. Vielma, J.P., Nemhauser, G.L.: Modeling disjunctive constraints with a logarithmic number of binary variables and constraints. *Math. Program.* 128(1): 49-72 (2011)
4. Vielma J P, Ahmed S, Nemhauser G. Mixed-integer models for nonseparable piecewise-linear optimization: Unifying framework and extensions. *Oper. Res.* 58(2): 303-315 (2010)
5. Silva, T.L., Camponogara, E.: A computational analysis of multidimensional piecewise-linear models with applications to oil production optimization. *Eur. J. Oper. Res.* 232(3): 630-642 (2014)
6. Bernreuther, M: Solving mixed-integer programming problems using piecewise linearization methods. Tech. Rep. 01/897074, University of Konstanz (2017)
7. Frenzen, C.L., Sasao, T., Butler, J.T.: On the number of segments needed in a piecewise linear approximation. *J. Comput. Appl. Math.* 234(2):437-446 (2010)
8. Tsai, J., Lin, M.: An improved framework for solving NLIPs with signomial terms in the objective or constraints to global optimality. *Comput. Chem. Eng.* 53(53):44-54 (2013)
9. Zhan, Y., Zheng, Q., Tseng, C., Pasiliao, E.: An accelerated extended cutting plane approach with piecewise linear approximations for signomial geometric programming. *J. Global Optim.* 70(6):1-21 (2018)
10. Fortuny-Amat, J., McCarl, B.: A representation and economic interpretation of a two-level programming problem. *J. Oper. Res. Soc.* 32(9): 783-792 (1981)
11. FICO Xpress Optimization Suite. MIP formulations and linearizations. Quick reference, available at: <https://www.artelys.com/uploads/pdfs/Xpress/mipformref-1.pdf>
12. Jünger, M., Liebling, T.M., Naddef, D., Nemhauser, G.L., Pulleyblank, W., Reinelt, G., Rinaldi, G., Wolsey, L.: 50 Years of Integer Programming 1958-2008: From the Early Years to the State-of-the-Art, Springer-Verlag, New York, 2010
13. Vielma, J.P.: Mixed integer linear programming formulation techniques. *SIAM Rev.* 57(1): 3-57 (2015)
14. Misener, R., Floudas, C.A.: Piecewise-linear approximations of multidimensional functions. *J. Optimiz. Theory App.*, 145(1): 120-147 (2010)
15. Rovatti, R., D'Ambrosio, C., Lodi, A., Martello, S.: Optimistic MILP modeling of non-linear optimization problems. *Eur. J. Oper. Res.* 239(1): 32-45 (2014)
16. Sridhar, S., Linderoth, J., Luedtke, J.: Locally ideal formulations for piecewise linear functions with indicator variables. *Oper. Res. Lett.* 41(6): 627-632 (2013)
17. Gupte, A., Ahmed, S., Cheon, M.S., Dey, S.: Solving mixed integer bilinear problems using MILP formulations. *SIAM J. Optimiz.* 23(2): 721-744 (2013)
18. Kolodziej, S., Castro, P.M., Grossmann, I.E.: Global optimization of bilinear programs with a multiparametric disaggregation technique. *J. Global Optim.* 2013, 57(4): 1039-1063.
19. Bonami, P., Lodi, A., Tramontani, A., Wiese, S.: On mathematical programming with indicator constraints. *Math. Program.* 151(1): 191-223 (2015)

20. Trespalacios, F., Grossmann, I.E.: Improved Big-M reformulation for generalized disjunctive programs. *Comput. Chem. Eng.* 76: 98-103 (2015)
21. Vielma J.P.: Embedding formulations and complexity for unions of polyhedra. *Manage. Sci.* 64(10): 4721-4734 (2018)
22. Basu, A., Martin, K., Ryan, C.T., Wang, G.: Mixed-integer linear representability, disjunctions, and variable elimination. In *Proc. International Conference on Integer Programming and Combinatorial Optimization*. Springer, Cham, 75-85 (2017)
23. Vielma, J.P., Dunning, I., Huchette, J., Lubin, M.: Extended formulations in mixed integer conic quadratic programming. *Math. Prog. Comp.* 9(3): 369-418 (2017)
24. Morán, R., Dey, S.S., Vielma, J.P.: A strong dual for conic mixed-integer programs. *SIAM J. Optimiz.* 22(3): 1136-1150 (2012)
25. Huchette, J. A.: Advanced mixed-integer programming formulations: Methodology, computation, and application. Doctoral Dissertation, MIT (2018)

Appendix C

Basics of Robust Optimization

To be uncertain is to be uncomfortable, but to be certain is to be ridiculous.

Real-world decision-making models often involve unknown data. Reasons for data uncertainty could come from inexact measurements or forecast errors. For example, in power system operation, the wind power generation and system loads are barely known exactly at the time when the generation schedule should be made; in inventory management, market price and demand volatility is the main source of financial risks. In fact, optimal solutions to mathematical programming problems can be highly sensitive to parameter perturbations [1]. The optimal solution to the nominal problem may be highly suboptimal or even infeasible in reality due to parameter inaccuracy. Consequently, there is a great need of a systematic methodology that is capable of quantifying the impact of data inexactness on the solution quality, and is able to produce robust solutions that are insensitive to data uncertainty.

Optimization under uncertainty has been a focus of the operational research community for a long time. Two approaches are prevalent to deal with uncertain data in optimization, namely stochastic optimization (SO) and robust optimization (RO). They differ in the ways of modeling uncertainty. The former one assumes that the true probability distribution of uncertain data is known or can be estimated from available information, and minimizes the expected cost in its objective function. SO provides strategies that are optimal in the sense of statistics. However, the probability distribution itself may be inexact owing to the lack of enough data, and the performance of the optimal solution could be sensitive to the probability distribution chosen in the SO model. The latter one considers uncertain data resides in a pre-defined uncertainty set, and minimizes the cost in the worst-case scenario in its objective function. Constraint violation is not allowed for all possible data realizations in the uncertainty set. RO is popular because it relies on simple data and distribution-free. From the computational perspective, it is equivalent to convex optimization problems for a variety of uncertainty sets and problem types; for the intractable cases, it can be solved via systematic iteration algorithms. For more technical details about RO, we refer to [1, 3, 4, 5], survey articles [6, 7], and many references therein. Recently, distributionally

robust optimization (DRO), an emerging methodology that inherits the advantages of SO and RO, has attracted wide attention. In DRO, uncertain data are described by probability distribution functions which are not known exactly and restricted in a functional ambiguity set constructed from available information and structured properties. The expected cost associated with the worst-case distribution is minimized, and the probability of constraint violations can be controlled via robust chance constraints. In many cases, the DRO can be reformulated as a convex optimization problem, or solved iteratively via convex optimization. RO and DRO approaches are young and active research fields, and the challenge is to explore tractable reformulations with various kinds of uncertainties. SO is a relatively mature technique, and the current research is focusing on probabilistic modeling of uncertainty, chance constrained programming, multi-stage SO such as stochastic dual dynamic programming, as well as more efficient computational methods.

There are several ways to categorize robust optimization methods. According to how uncertainty is dealt with, they can be classified into static (single-stage) RO and dynamic (multi-stage) RO. According to how uncertainty is modeled, they can be divided into RO and DRO. In the latter category, the ambiguity set for probability distribution can be further classified into the moment based one and the divergence based one. We will shed light on each of them in this chapter. Specifically, RO will be discussed in Sect. C.1 and Sect. C.2, moment-based DRO will be presented in Sect. C.3, and divergence-based DRO, also called robust SO will be illuminated in Sect. C.4. In the operations research community, DRO and robust SO refer to the same thing: optimization problem with distributional uncertainty, and can be used interchangeably, although DRO is preferred by the majority of researchers. In this book, we intentionally distinguish them because the moment ambiguity set can be set up with little information and is more likely a RO; the divergence based set relies on an empirical distribution (may be inexact), so is more similar to an SO. In fact, the gap between SO and RO has been significantly narrowed by recent research progress in the sense of data-driven optimization.

C.1 Static Robust Optimization

For the purpose of clarity, we begin to explain the paradigm of static RO from LPs, the best known and most frequently used mathematical programming problem in engineering applications. It is relatively easy to derive tractable robust counterparts with various uncertainty sets. Nevertheless, most results can be readily generalized to robust conic programs. The general form of an LP with uncertain parameters can be written as follows:

$$\min_x \{c^T x \mid Ax \leq b\} : (A, b, c) \in W \quad (\text{C.1})$$

where x is the decision variable, A , b , c are coefficient matrices with compatible dimensions, and W denotes the set of all possible data realizations constructed from available information or historical data, or merely a rough estimation.

Without loss of generality, we can assume that the objective function and the constraint right-hand side in (C.1) are certain, and uncertainty only exists in coefficient matrix A . To see this, it is not difficult to observe that problem (C.1) can be written as an epigraph form

$$\min_{t,x,y} \{t \mid c^T x - t \leq 0, Ax - by \leq 0, y = 1\} : (A, b, c) \in W$$

By introducing additional scalar variables t and y , coefficients appearing in the objective function and constraint right-hand side are constants. With this transformation, it will be more convenient to define the feasible solution and the optimal solution to (C.1). Hereinafter, we neglect the uncertainty in cost coefficient vector c and constraint right-hand vector b without particular mention, and consider problem

$$\min_x \{c^T x \mid Ax \leq b\} : A \in W \quad (\text{C.2})$$

Next we present solution concepts of static RO under uncertain data.

C.1.1 Basic Assumptions and Formulations

Basic assumptions and definitions in static RO [1] are summarized as follows.

Assumption C.1 *Vector x represents “here-and-now” decisions: they should be determined without knowing exact values of uncertain parameters.*

Assumption C.2 *Once the decisions are made, constraints must be feasible when the actual data is within the uncertainty set W , and may be either feasible or not when the actual data step outside the uncertainty set W .*

These assumptions bring about the definition for a feasible solution of (C.2).

Definition C.1. A vector x is called a robust feasible solution to (C.2) if the following condition holds:

$$Ax \leq b, \forall A \in W \quad (\text{C.3})$$

To prescribe an optimal solution, the worst-case criterion is widely accepted in RO studies, leading to the following definition:

Definition C.2. The robust optimal value of (C.2) is the minimum value of the objective function over all possible x that satisfies (C.3).

After we have agreed on the meanings of feasibility and optimality of (C.2), we can seek the optimal solution among all robust feasible solutions to the problem. Now, the robust counterpart (RC) of the uncertain LP (C.2) can be described as:

$$\begin{aligned} \min_x \quad & c^T x \\ \text{s.t.} \quad & a_i^T x \leq b_i, \forall i, \forall A \in W \end{aligned} \quad (\text{C.4})$$

where a_i^T is the i -th row of matrix A , and b_i is the i -th element of vector b . We have two observations on the formulation of robust constraints in (C.4).

Proposition C.1. *Robust feasible solutions of (C.4) remain the same if we replace W with the Cartesian product $\hat{W} = W_1 \times \cdots \times W_n$, where $W_i = \{a_i | \exists A \in W\}$ is the projection of W on the coefficient space of i -th row of A .*

This is called the constraint-wise property in static RO [1]. The reason is

$$a_i^T x \leq b_i, \forall A \in W \Leftrightarrow \max_{A \in W} a_i^T x \leq b_i \Leftrightarrow \max_{a_i \in W_i} a_i^T x \leq b_i$$

As a result, problem (C.4) comes down to

$$\begin{aligned} \min_x \quad & c^T x \\ \text{s.t.} \quad & a_i^T x \leq b_i, \forall a_i \in W_i, \forall i \end{aligned} \quad (\text{C.5})$$

Proposition C.1 seems rather counter-intuitive. One may perceive that (C.4) will be less conservative with uncertainty set W since it is a subset of \hat{W} . In fact, later we will see that this intuition is true for adjustable robustness.

Proposition C.2. *Robust feasible solutions of (C.5) remain the same if we replace W_i with its convex hull $\text{conv}(W_i)$.*

To see this, let vector a_i^j , $j = 1, 2, \dots$ be the extreme points of W_i , then any point $\bar{a}_i \in \text{conv}(W_i)$ can be expressed by $\bar{a}_i = \sum_j \lambda_j a_i^j$, where $\lambda_j \geq 0$, $\sum_j \lambda_j = 1$ are weight coefficients. If x is feasible for all extreme points a_i^j , i.e., $a_i^j x \leq b_i, \forall j$, then

$$\bar{a}_i^T x = \sum_j \lambda_j a_i^j x \leq \sum_j \lambda_j b_i = b_i$$

which indicates that the constraint remains intact for all uncertain parameters reside in $\text{conv}(W_i)$.

Combining Propositions C.1 and C.2, we can conclude that the robust counterpart of an uncertain LP with a certain objective remains intact even if sets W_i of uncertain data are extended to their closed convex hulls, and W to the Cartesian product of the resulting sets. In other words, we can make a further assumption on the uncertainty set without loss of generality.

Assumption C.3 *The uncertainty set W is the Cartesian product of closed and convex sets.*

C.1.2 Tractable Reformulations

The constraint-wise property enables us to analyze the robustness of each constraint $a_i^T x \leq b_i$, $\forall a_i \in W_i$ separately. Without particular mention, we will omit the subscript i for brevity. To facilitate discussion, it is convenient to parameterize the uncertain vector as $a = \bar{a} + P\zeta$, where \bar{a} is the nominal value of a , P is a constant matrix, ζ is a new variable that is uncertain. This section will focus on how to derive tractable reformulation for robust constraints in the form of

$$(\bar{a} + P\zeta)^T x \leq b, \quad \forall \zeta \in Z \quad (\text{C.6})$$

where Z is the uncertainty set of variable ζ . For same reasons, we can assume that Z is closed and convex. A computationally tractable problem means that there are known solution algorithms which can solve the problem with polynomial running time in its input size even in the worst case. It has been shown in [1] that problem (C.5) is generally intractable even if each W_i is closed and convex. Nevertheless, tractability can be preserved for some special classes of uncertainty sets. Some well-known results are summarized in the following.

Condition (C.6) contains an infinite number of constraints due to the enumeration over set Z . Later we will see that for some particular uncertainty sets, the \forall quantifier as well as the uncertain parameter ζ can be eliminated by using duality theory, and the resulting constraint in variable x is still convex.

1. Polyhedral uncertainty set

We start with a commonly used uncertainty set: a polyhedron

$$Z = \{\zeta \mid D\zeta + q \geq 0\} \quad (\text{C.7})$$

where D and q are constant matrices with compatible dimensions.

To exclude the \forall quantifier for variable ζ , we investigate the worst case of the left-hand side and require

$$\bar{a}^T x + \max_{\zeta \in Z} (P^T x)^T \zeta \leq b \quad (\text{C.8})$$

For a fixed x , the second term is the optimum of an LP in variable ζ . Duality theory of LP says that the following relation holds

$$(P^T x)^T \zeta \leq q^T u, \quad \forall \zeta \in Z, \quad \forall u \in U \quad (\text{C.9})$$

where u is the dual variable, and $U = \{u \mid D^T u + P^T x = 0, u \geq 0\}$ is the feasible region of the dual problem. Please be cautious on the sign of u . We actually replace u with $-u$ in the original dual LP. Therefore, a necessary condition to validate (C.6) is

$$\exists u \in U : \bar{a}^T x + q^T u \leq b \quad (\text{C.10})$$

It is also sufficient if the second term takes its minimum value over U , because strong duality always holds for LPs, i.e. $(P^T x)^T \zeta = q^T u$ is satisfied at the optimal solution. In this regard, (C.8) is equivalent to

$$\bar{a}^T x + \min_{u \in U} q^T u \leq b \quad (\text{C.11})$$

In fact, the “min” operator in (C.11) can be omitted in a RC optimization problem that minimizes the objective function, and thus renders polyhedral constraints, although (C.8) is not given in a closed form and seems non-convex.

In summary, the RC problem of an uncertain LP with polyhedral uncertainty

$$\begin{aligned} \min_x \quad & c^T x \\ \text{s.t.} \quad & (\bar{a}_i + P_i \zeta_i)^T x \leq b_i, \quad \forall \zeta_i \in Z_i, \quad \forall i \\ & Z_i = \{\zeta_i \mid D_i \zeta_i + q_i \geq 0\}, \quad \forall i \end{aligned} \quad (\text{C.12})$$

can be equivalently formulated as

$$\begin{aligned} \min_x \quad & c^T x \\ \text{s.t.} \quad & \bar{a}_i^T x + q_i^T u_i \leq b_i, \quad \forall i \\ & D_i^T u_i + P_i^T x = 0, \quad u_i \geq 0, \quad \forall i \end{aligned} \quad (\text{C.13})$$

which is still an LP.

2. Cardinality constrained uncertainty set

Cardinality constrained uncertainty set is a special class of polyhedral uncertainty set which incorporates a budget constraint and defined as follows

$$Z(\Gamma) = \left\{ \zeta \mid -1 \leq \zeta_j \leq 1, \quad \forall j, \quad \sum_j |\zeta_j| \leq \Gamma \right\} \quad (\text{C.14})$$

where Γ is called the budget of uncertainty [2]. Motivated by the fact that each entry ζ_j is unlikely to reach 1 or -1 at the same time, the budget constraint controls the total data deviation from their forecast values. In other words, the decision maker can achieve a compromise between the level of solution robustness and the optimal cost by adjusting the value of Γ , which should

be less than the dimension of ζ , otherwise the budget constraint will be redundant.

Although the cardinality constrained uncertainty set $Z(\Gamma)$ is essentially a polyhedron, the number of its facets, or the number of linear constraints in (C.7), grows exponentially in the dimension of ζ , leading to a huge and dense coefficient matrix for the uncertainty set. To circumvent this difficulty, we can lift it into a higher dimensional space as follows by introducing auxiliary variables

$$Z(\Gamma) = \left\{ \zeta, \sigma \mid -\sigma_j \leq \zeta_j \leq \sigma_j, \sigma_j \leq 1, \forall j, \sum_j \sigma_j \leq \Gamma \right\} \quad (\text{C.15})$$

The first inequality naturally suggests $\sigma_j \geq 0, \forall j$. It is easy to see the equivalence of (C.14) and (C.15), and the numbers of variables and constraints in the latter one grows linearly in the dimension of ζ .

Following a similar paradigm, certifying constraint robustness with a cardinality constrained uncertainty set requires the optimal value function of the following LP in variables ζ and σ representing the uncertainty

$$\begin{aligned} & \max_{\zeta, \sigma} \quad (P^T x)^T \zeta \\ \text{s.t.} \quad & -\zeta_j - \sigma_j \leq 0, \forall j : u_j^n \\ & \zeta_j - \sigma_j \leq 0, \forall j : u_j^m \\ & \sigma_j \leq 1, \forall j : u_j^b \\ & \sum_j \sigma_j \leq \Gamma : u_r \end{aligned} \quad (\text{C.16})$$

where $u_j^n, u_j^m, u_j^b, \forall j$, and u_r following a colon are the dual variables associated with each constraint. The dual problem of (C.16) is given by

$$\begin{aligned} & \min_{u^n, u^m, u^b, u_r} \quad u_r \Gamma + \sum_j u_j^b \\ \text{s.t.} \quad & u_j^m - u_j^n = (P^T x)_j, \forall j \\ & -u_j^m - u_j^n + u_j^b + u_r = 0, \forall j \\ & u_j^m, u_j^n, u_j^b \geq 0, \forall j, u_r \geq 0 \end{aligned} \quad (\text{C.17})$$

In summary, the RC problem of an uncertain LP with cardinality constrained uncertainty

$$\begin{aligned} & \min_x \quad c^T x \\ \text{s.t.} \quad & (\bar{a}_i + P_i \zeta_i)^T x \leq b_i, \forall \zeta_i \in Z_i(\Gamma_i), \forall i \end{aligned} \quad (\text{C.18})$$

can be equivalently formulated as

Table C.1 Equivalent convex formulations with different uncertainty sets

Uncertainty	Z	Robust reformulation	Tractability
Box	$\ \zeta\ _\infty \leq 1$	$\bar{a}^T x + \ P^T x\ _1 \leq b$	LP
Ellipsoidal	$\ \zeta\ _2 \leq 1$	$\bar{a}^T x + \ P^T x\ _2 \leq b$	LP
p -norm	$\ \zeta\ _p \leq 1$	$\bar{a}^T x + \ P^T x\ _q \leq b$	Convex program
Proper cone	$D\zeta + q \in K$	$\begin{cases} \bar{a}^T x + q^T u \leq b \\ D^T u + P^T x = 0 \\ u \in K^* \end{cases}$	Conic LP
Convex constraints	$h_k(\zeta) \leq 0, \forall k$	$\begin{cases} \bar{a}^T x + \sum_k \lambda_k h_k^* \left(\frac{u^k}{\lambda_k} \right) \leq b \\ \sum_k u^k = P^T x \\ \lambda_k \geq 0, \forall k \end{cases}$	Convex program

$$\begin{aligned}
& \min_x \quad c^T x \\
& \text{s.t.} \quad \bar{a}_i^T x + u_{ri} \Gamma_i + \sum_j u_{ij}^b \leq b_i, \quad \forall i \\
& \quad \quad u_{ij}^m - u_{ij}^n = (P_i^T x)_j, \quad \forall i, \quad \forall j \\
& \quad \quad -u_{ij}^m - u_{ij}^n + u_{ij}^b + u_{ir} = 0, \quad \forall i, \quad \forall j \\
& \quad \quad u_{ij}^m, u_{ij}^n, u_{ij}^b \geq 0, \quad \forall i, \quad \forall j, \quad u_{ir} \geq 0, \quad \forall i
\end{aligned} \tag{C.19}$$

which is still an LP.

3. Several other uncertainty sets

Equivalent convex formulations of the uncertain constraint (C.6) with some other uncertainty sets are summarized in Table C.1 [4]. These outcomes are derived using the similar method described previously.

Table C.1 includes three cases: the p -norm uncertainty, the conic uncertainty, and general convex uncertainty. In the p -norm case, the Hölder's inequality is used, i.e.:

$$(P^T x)^T \zeta \leq \|P^T x\|_p \|\zeta\|_q \tag{C.20}$$

where $\|\cdot\|_p$ and $\|\cdot\|_q$ with $p^{-1} + q^{-1} = 1$ are a pair of dual norms. Since norm function of any order is convex [8], the resulting RC is a convex program. Moreover, if q is a positive rational number, the q -order cone constraints can be represented by a set of SOC inequalities [9], which is computationally more friendly. Box (∞ -norm) and ellipsoidal (2-norm) uncertainty sets are special kinds of p -norm ones.

In the general conic case, conic duality theory [10] is used. K^* stands for the dual cone of K , and the polyhedral uncertainty is a special kind of this case when K is the nonnegative orthant.

In the general convex case, Fenchel duality, a basic theory in convex analysis, is needed. Notation h^* stands for the convex conjugate function, i.e. $h^*(x) = \sup_y x^T y - h(y)$. The detailed proof of RC reformulations and more examples can be found in [11].

Above analysis focuses on the situation in which problem functions are linear in decision variables, and problem data are affine in some uncertain parameters, such as the form $a = \bar{a} + P\zeta$. For robust quadratic optimization, robust semidefinite optimization, robust conic optimization, and robust discrete optimization, in which the optimization problem is nonlinear and discontinuous, please refer to [1] and [3]; for quadratic type uncertainty, please refer to [1] (in Sect. 1.4) and [11].

C.1.3 Formulation Issues

To help practitioners build a well-defined and easy-to-solve robust optimization model, some important modeling issues and deeper insights are discussed in this section.

1. Choosing the uncertainty set

Since a robust solution remains feasible if the uncertain data does not step outside the uncertainty set, the level of robustness mainly depends on the shape and size of the uncertainty set. The more reliable, the higher the cost. One may wish to seek a trade-off between reliability and economy. This inspires the development of smaller uncertainty sets with a certain probability guarantee that the constraint violation is unlikely to happen. Such guarantees are usually described via a chance constraint

$$\Pr_{\zeta}[a(\zeta)^T x \leq b] \geq 1 - \varepsilon \quad (\text{C.21})$$

For $\varepsilon = 0$, chance constraint (C.21) is protected in the traditional sense of RO. When $\varepsilon > 0$, it becomes challenging to derive tractable reformulation for (C.21), especially when the probability distribution of uncertain data is unclear or inaccurate. In fact, this issue is closely related to the DRO that will be discussed later on. Here we provide some simple results which help the decision maker choose the parameter of the uncertainty set.

It is revealed that if $\mathbb{E}[\zeta] = 0$, the components of ζ are independent, and the uncertainty set takes the form

$$Z = \{\zeta \mid \|\zeta\|_2 \leq \Omega, \|\zeta\|_{\infty} \leq 1\} \quad (\text{C.22})$$

then chance constraint (C.21) holds with a probability of at least $1 - \exp(-\Omega^2/2)$ (see [1], Proposition 2.3.3).

Moreover, if the uncertainty set takes the form

$$Z = \{\zeta \mid \|\zeta\|_1 \leq \Gamma, \|\zeta\|_\infty \leq 1\} \quad (\text{C.23})$$

then chance constraint (C.21) holds with a probability of at least $1 - \exp(-\Gamma^2/2L)$, where L is the dimension of ζ (see [1], Proposition 2.3.4, and [2]).

It is proposed to construct uncertainty sets based on the central limit theorem. If each component of ζ is independent and identically distributed with mean μ and variance σ^2 , the uncertainty set can be built as [12]

$$Z = \left\{ \zeta \mid \left| \sum_{i=1}^L \zeta_i - L\mu \right| \leq \rho\sqrt{L}\sigma, \|\zeta\|_\infty \leq 1 \right\} \quad (\text{C.24})$$

where parameter ρ is used to control the probability guarantee. Variations of this formulation can take other distributional information into account, such as data correlation and long tail-effect. It is a special kind of polyhedral uncertainty, however, it is unbounded for $L > 1$, since the components can be arbitrarily large as long as their summation is relatively small. Unboundedness may prevent establishing tractable RCs.

Additional references are introduced in further reading.

2. How to solve a problem without a clear tractable reformulation?

The existence of a tractable reformulation for a static RO problem largely depends on the type of the uncertainty set. If the robust counterpart cannot be written as a tractable convex program, a smart remedy is to use an adaptive scenario generation procedure: first solve the problem with a smaller uncertainty set Z_S which is a subset of the original one Z , and the problem with Z_S has a known tractable reformulation. If the optimal solution x^* is robust against all scenarios in Z , it is also an optimal solution of the original problem. Otherwise, we have to identify a scenario $\zeta^* \in Z$ which leads to the most severe violation, which can be implemented by solving

$$\max \left\{ (P^T x^*)^T \zeta \mid \zeta \in Z \right\} \quad (\text{C.25})$$

where Z is a closed and convex set as validated in Assumption C.3, and then append a cutting plane

$$a(\zeta^*)^T x \leq b \quad (\text{C.26})$$

to the reformulation problem. (C.26) removes x that will cause infeasibility in scenario ζ^* , so is called a feasibility cut. It is linear and does not alter tractability. Then the updated problem is solved again. According to Proposition C.2, the new solution x^* will be robust for uncertain data in the convex hull of $Z_S \cup \zeta^*$. Above procedure continues until robustness is certified over the original uncertainty set Z .

This simple approach often converges quickly in a few number of iterations. Its advantage is that tractability is preserved. When we choose $Z_S = \zeta^0$, where ζ^0 is the nominal scenario or forecast, it could be more efficient than

using convex reformulations, because only LPs (whose sizes are almost equal to the problem without uncertainty, and grows slowly) and simple convex programs (C.25) are solved, see [13] for a comparison. This paradigm is an essential strategy for solving the adjustable RO problems in the next section.

3. How to deal with equality constraints?

Although the theory of static RO is relatively mature, it encounters difficulties in dealing with equality constraints. For example, consider $x + a = 1$ where $a \in [0, 0.1]$ is uncertain. However, one can seldom find a solution that makes the equality hold true for multiple values of a . The problem remains if you write a equality into a pair of opposite inequalities. In fact, this issue is inevitable in the static setting. In addition, this limitation will lead to completely different robust counterpart formulations for originally equivalent deterministic problems.

Consider the inequality $ax \leq 1$, which is equivalent to $ax + s = 1$, $s \geq 0$. Suppose a is uncertain and belongs to interval $[1, 2]$, their respective robust counterparts are given by

$$ax \leq 1, \forall a \in [1, 2] \quad (\text{C.27})$$

and

$$ax + s = 1, \forall a \in [1, 2], s \geq 0 \quad (\text{C.28})$$

The feasible set for (C.27) is $x \leq 1/2$, and is $x = 0$ for (C.28). By observing this difference, it is suggested that a static RO model should avoid using slack variables in constraints with uncertain parameters.

Sometimes, the optimization problem may contain state variables which can respond to parameter changes by adjusting their values. In such circumstance, equality constraint can be used to eliminate state variables. Nevertheless, such an action may lead to a problem that contains nonlinear uncertainties, which are challenging to solve. An example is taken from [4] to illustrate this issue. The constraints are

$$\begin{aligned} \zeta_1 x_1 + x_2 + x_3 &= 1 \\ x_1 + x_2 + \zeta_2 x_3 &\leq 5 \end{aligned} \quad (\text{C.29})$$

where ζ_1 and ζ_2 are uncertain.

If x_1 is a state variable and $\zeta_1 \neq 0$, substituting $x_1 = (1 - x_2 - x_3)/\zeta_1$ in the second inequality results in

$$\left(1 - \frac{1}{\zeta_1}\right) x_2 + \left(\zeta_2 - \frac{1}{\zeta_1}\right) x_3 \leq 5 - \frac{1}{\zeta_1}$$

in which the uncertainty becomes nonlinear in the coefficients.

If x_2 is a state variable, substituting $x_2 = 1 - \zeta_1 x_1 - x_3$ in the inequality yields

$$(1 - \zeta_1)x_1 + (\zeta_2 - 1)x_3 \leq 4$$

in which the uncertainty sustains linear in the coefficients.

If x_3 is a state variable, substituting $x_3 = 1 - \zeta_1 x_1 - x_2$ in the inequality gives

$$(1 - \zeta_1 \zeta_2)x_1 + (1 - \zeta_2)x_2 \leq 5 - \zeta_2$$

in which the uncertainty is nonlinear in the coefficients.

In conclusion, in the case that x_2 is a state variable, the problem is easier from a computational perspective. It is important to note that the physical interpretation of variable elimination is to determine the adjustable variable with exact information on the uncertain data. If no adjustment is allowed in (C.29), the only robust feasible solution is $x_1 = x_3 = 0$, $x_2 = 1$, which is rather restrictive. The adjustable RO will be elaborated in detail in the next section.

4. Pareto efficiency of the robust solution

The concept of Pareto efficiency in RO problems is proposed in [14]. If the optimal solution under the worst-case data realization is not unique, it is rational to compare their performances in non-worst-case scenarios: an alternative solution may give an improvement in the objective value for at least one data scenario without deteriorating the objective performances in all other scenarios. To present related concept tersely, we restrict the discussion on the following robust LP with objective uncertainty

$$\max\{p^T x \mid \text{s.t. } x \in X, \forall p \in W\} = \max_{x \in X} \left\{ \min_{p \in W} p^T x \right\} \quad (\text{C.30})$$

where $W = \{p \mid Dp \geq d\}$ is a polyhedral uncertainty set for the price vector p ; $X = \{x \mid Ax \leq b\}$ is the feasible region which is independent of the uncertainty. More general cases are elaborated in [14]. We consider this form because it is easy to discuss related issues, although objective uncertainty can be moved into constraints.

For a given strategy x , the worst-case uncertainty is

$$\min\{p^T x \mid \text{s.t. } p \in W\} = \max\{d^T y \mid \text{s.t. } y \in Y\} \quad (\text{C.31})$$

where $Y = \{y \mid D^T y = x, y \geq 0\}$ is the feasible set for dual variable y . Substituting (C.31) in (C.30) gives

$$\max\{d^T y \mid \text{s.t. } D^T y = x, y \geq 0, x \in X\} \quad (\text{C.32})$$

which is an LP. Its solution x is the robust optimal one to (C.30), and the worst-case price p can be found by solving the left-hand side LP in (C.31). Let z^{RO} be the optimal value of (C.32), and then the set of robust optimal solutions for (C.30) can be expressed via

$$X^{RO} = \{x \mid x \in X : \exists y \in Y \text{ such that } y^T d \geq z^{RO}\} \quad (\text{C.33})$$

If (C.32) has a unique optimal solution, X^{RO} is a singleton; otherwise, a Pareto optimal robust solution can be formally defined.

Definition C.3. [14] $x \in X^{RO}$ is a Pareto optimal solution for problem (C.30) if there is no other $\bar{x} \in X$ such that $p^T \bar{x} \geq p^T x$, $\forall p \in W$ and $\bar{p}^T \bar{x} > \bar{p}^T x$ for some $\bar{p} \in W$.

The terminology “Pareto optimal” is borrowed from multi-objective optimization theory: RO problem (C.30) is viewed as a multi-objective LP with infinitely many objectives, each of which corresponds to a particular $p \in W$. Some interesting problems are elaborated.

a. Pareto efficiency test

In general, it is not clear whether X^{RO} contains multiple solutions, at least before a solution $x \in X^{RO}$ is found. To test whether a given solution x is a robust optimal one or not, it is proposed to solve a new LP

$$\begin{aligned} \max_y \quad & \bar{p}^T y \\ \text{s.t.} \quad & y \in W^* \\ & x + y \in X \end{aligned} \tag{C.34}$$

where \bar{p} is a relative interior of the polyhedral uncertainty set W , which is usually set to the nominal scenario, and $W^* = \{y \mid \exists \lambda : d^T \lambda \geq 0, D^T \lambda = y, \lambda \geq 0\}$ is the dual cone of W . Please refer to Sect. A.2.1 and equation (A.27) for the dual cone of a polyhedral set. Since $y = 0, \lambda = 0$ is always feasible in (C.34), the optimal value is either zero or strictly positive. In the former case, x is also a Pareto optimal solution; in the latter case, $\bar{x} = x + y^*$ dominates x and itself is Pareto optimal for any y^* that solves LP (C.34) [14]. The interpretation of (C.34) is clear: since $y \in W^*$, $y^T p$ must be non-negative for all $p \in W$. If we can find y that leads to a strict objective improvement for \bar{p} , then $x + y$ would be Pareto optimal.

In view of the above interpretation, it is a direct conclusion that for an arbitrary relative interior point $\bar{p} \in W$, the optimal solutions to the problem

$$\max \{ \bar{p}^T x \mid x \in X^{RO} \} \tag{C.35}$$

are Pareto optimal.

b. Characterizing the set of Pareto optimal solutions

It is interesting to characterize the Pareto optimal solution set X^{PRO} . After we get z^{RO} and X^{RO} , solve the following LP

$$\begin{aligned} \max_{x, y, \lambda} \quad & \bar{p}^T y \\ \text{s.t.} \quad & d^T \lambda \geq 0, D^T \lambda = y, \lambda \geq 0 \\ & x \in X^{RO}, x + y \in X \end{aligned} \tag{C.36}$$

and we can conclude $X^{PRO} = X^{RO}$ if and only if the optimal value of (C.36) is equal to 0 [14]. If this is true, the decision maker would not have to worry about Pareto efficiency, as any solution in X^{RO} is also Pareto optimal. More broadly, the set X^{PRO} is shown to be non-convex and is contained in the boundary of X^{RO} .

c. Optimization over Pareto optimal solutions

In the case that X^{PRO} is not a singleton, one may consider to optimize a linear secondary objective over X^{PRO} , i.e.:

$$\max\{r^T x \mid \text{s.t. } x \in X^{PRO}\} \quad (\text{C.37})$$

It is demonstrated in [14] that if r lies in the relative interior of W , the decision maker can simply replace X^{PRO} with X^{RO} in (C.37) without altering the problem solution, due to the property revealed in (C.35). In more general cases, problem (C.37) can be formulated as an MILP [14]

$$\begin{aligned} \max_{x, \mu, \eta, z} \quad & r^T x \\ \text{s.t.} \quad & x \in X^{RO} \\ & \mu \leq M(1 - z) \\ & b - Ax \leq Mz \\ & DA^T \mu - d\eta \geq D\bar{p} \\ & \mu, \eta \geq 0, z \in \{0, 1\}^m \end{aligned} \quad (\text{C.38})$$

where M is a sufficiently large number, m is the dimension of vector z . To show their equivalence, it is revealed that the feasible set of (C.38) depicts an optimal solution of (C.34) with a zero objective value [14]. In other words, the constraints of (C.38) contain the KKT optimality condition of (C.34). To see this, the binary vector z imposes the complementarity and slackness condition $\mu^T(b - Ax) = 0$, which ensures λ, μ, η are the optimal solution of the following primal-dual LP pair

$$\begin{array}{ll} \max_{\lambda} & \bar{p}^T D^T \lambda \\ \text{Primal :} & \text{s.t. } \lambda \geq 0 \\ & d^T \lambda \geq 0 \\ & AD^T \lambda \leq b - Ax \\ \min_{\mu, \eta} & \mu^T (b - Ax) \\ \text{Dual :} & \text{s.t. } \mu \geq 0 \\ & \eta \geq 0 \\ & DA^T \mu - d\eta \geq D\bar{p} \end{array}$$

The original variable y in (C.34) is eliminated via equality $D^T \lambda = y$ in the dual cone W^* . According to strong duality, the optimal value of the primal LP (C.34) is $\bar{p}^T D^T \lambda = \bar{p}^T y = 0$, and Pareto optimality is guaranteed.

In practice, Pareto inefficiency is not a contrived phenomenon, see various examples in [14] and power market examples in [15, 16].

5. On max-min and min-max formulations

In many literatures, the robust counterpart problem of (C.2) is written as a min-max form

$$\text{Opt-1} = \min_x \max_{A \in W} \{c^T x \mid \text{s.t. } Ax \leq b\} \quad (\text{C.39})$$

which means x is determined before A takes a value in W , and the decision maker can foresee the worst consequence of deploying x brought by the perturbation of A . To make a prudent decision that is insensitive to data perturbation, the decision maker resorts to minimizing the maximal objective.

The max-min formulation

$$\text{Opt-2} = \max_{A \in W} \min_x \{c^T x \mid \text{s.t. } Ax \leq b\} \quad (\text{C.40})$$

has a different interpretation: the decision maker can first observe the realization of uncertainty, and then recovers the constraints by deploying a corrective action x as a response to the observed A . Certainly, this specific x may not be feasible for other $A \in W$. On the other hand, the uncertainty, like a rational player, can foresee the optimal action taken by the human decision maker, and select a strategy that will yield a maximal objective value even an optimal corrective action is deployed.

From above analysis, the feasible region of x in (C.39) is a subset of that in (C.40), because (C.40) only accounts for a special scenario in W . As a result, their optimal values satisfy $\text{Opt-1} \geq \text{Opt-2}$.

Consider the following problem in which the uncertainty is not constraint-wise

$$\begin{aligned} \min_x \quad & x_1 + x_2 \\ \text{s.t.} \quad & x_1 \geq a_1, \quad x_2 \geq a_2, \quad \forall a \in W \end{aligned} \quad (\text{C.41})$$

where $W = \{a \mid a \geq 0, \|a\|_2 \leq 1\}$.

For the min-max formulation, since x should be feasible for all possible values of a , it is necessary to require $x_1 \geq 1$ and $x_2 \geq 1$, and $\text{Opt-1} = 2$ for problem (C.41).

As for the max-min formulation, as x is determined in response to the value of a , it is clear that the optimal choice is $x_1 = a_1$ and $x_2 = a_2$, so the problem becomes

$$\begin{aligned} \max_a \quad & a_1 + a_2 \\ \text{s.t.} \quad & a_1^2 + a_2^2 \leq 1 \end{aligned}$$

whose optimal value is $\text{Opt-2} = \sqrt{2} < \text{Opt-1}$.

As a short conclusion, static RO models discussed in this section are used to immunize against constraint violation or objective volatility caused by data perturbations, without jeopardizing computational tractability. General approaches involve reformulating the original uncertainty dependent constraints into deterministic convex ones without uncertain data, such that feasible so-

lutions of the robust counterpart program remain feasible for all data realizations in the pre-specified uncertainty set, which interprets the meaning of robustness.

C.2 Adjustable Robust Optimization

Several reasons call for developing new decision-making mechanisms to overcome limitations of the static RO approach: 1) Equality constraints often give rise to infeasible robust counterpart problems in the static setting; 2) real-world decision-making process may involve multiple stages, in which some decisions indeed can be made after the uncertain data has been known or can be predicted accurately. Take power system operation for an example, the on-off status of generation units must be made several hours before real-time dispatch when the renewable power is unclear; however, the output of some units (called AGC units) can change in response to the real values of system demands and renewable generations. This section will be devoted to the adjustable robust optimization (ARO) with two stages, which leverages the adaptability in the second stage. We still focus our attention on the linear case.

C.2.1 Basic Assumptions and Formulations

The essential difference between static RO and ARO approaches stems from the manner of decision making.

Assumption C.4 *In an ARO problem, some variables are “here-and-now” decisions, whereas the rest are “wait-and-see” decisions: they can be made at a later moment according to the observed data.*

In analogy to the static case, the decision-making mechanism can be explained.

Assumption C.5 *Once the here-and-now decisions are made, there must be at least one valid wait-and-see decision which is able to recover constraints in response to the observed data realization, if the actual data is within the uncertainty set.*

In this regard, we can say here-and-now decisions are robust against the uncertainty, and wait-and-see decisions are adaptive to the uncertainty. These terminologies are borrowed from two-stage SO models. In fact, there is a close relation between two-stage SO and two-stage RO [17, 18].

Now we are ready to post the compact form of a linear ARO problem with an uncertain constraint right-hand side:

$$\min_{x \in X} \left\{ c^T x + \max_{w \in W} \min_{y(w) \in Y(x, w)} d^T y(w) \right\} \quad (\text{C.42})$$

where x is the here-and-now decision variable (or the first-stage decision variable), and X is the feasible region of x ; w is the uncertain parameter, and W is the uncertainty set, which has been discussed in the previous section; $y(w)$ is the wait-and-see decision variable (or second-stage decision variable), which can be adjusted according to the actual data of w , so it is represented as a function of w ; Y is the feasible region of y given the values of x and w , because the here-and-now decision is not allowed to change in this stage, and the exact value of w is known. It has a polyhedral form

$$Y(x, w) = \{y \mid Ax + By + Cw \leq b\} \quad (\text{C.43})$$

where A , B , C , and b are constant matrices and vector with compatible dimensions. It is clear that both of the here-and-now decision x and the data uncertainty w can influence the feasible region Y in the second stage. We define $w = 0$ the nominal scenario and assume 0 is a relative interior of W . Otherwise, we can decompose the uncertainty as $w = w^0 + \Delta w$ and merge the constant term Cw^0 into the right-hand side as $b \rightarrow b - Cw^0$, where w^0 is the predicted or expected value of w , and Δw is the forecast error, which is the real uncertain parameter.

It should be pointed out that x , w , and y may contain discrete decision variables. Later we will see, integer variables in x and w do not significantly alter the solution algorithm of ARO. However, because integrality in y prevents the use of LP duality theory, the computation will be greatly challenged. Although we assume coefficient matrices are constants in (C.43), most results in this section can be generalized if matrix A is a linear function in w ; the situation would be complicated in matrix B is uncertainty-dependent. The purpose for the specific form in (C.43) is that it is more dedicated to the problems considered in this book: uncertainties originates from renewable/load volatility can be modeled by term Cw in (C.43), and the coefficients representing component and network parameters are constants.

Assumption C.5 inspires the definition for a feasible solution of ARO (C.42).

Definition C.4. A first-stage decision x is called robust feasible in (C.42) if the feasible region $Y(x, w)$ is non-empty for all $w \in W$, and the set of robust feasible solutions are given by:

$$X_R = \{x \mid x \in X : \forall w \in W, Y(x, w) \neq \emptyset\} \quad (\text{C.44})$$

Please be aware of the sequence in (C.44): x takes its value first, and then parameter w chooses a value in W before some $y \in Y$ does. The non-emptiness of Y is guaranteed by the selection of x for an arbitrary w . If we swap the latter two terms and write $Y(x, w) \neq \emptyset, \forall w \in W$, like the form in a static RO, it sometimes cause confusion that both x and y are here-and-now

type decisions, the adaptiveness vanishes, and thus X_R may become empty if uncertainty appears in an equality constraint, as analyzed in the previous section.

The definition of an optimal solution depends on the decision maker's attitude towards the cost in the second stage. In (C.42), we adopt the following definition.

Definition C.5. (Min-max cost criterion) An optimal solution of (C.42) is a pair of here-and-now decision $x \in X_R$ and wait-and-see decision $y(w^*)$ corresponding to the worst-case scenario $w^* \in W$, such that the total cost in scenario w^* is minimal, where the worst-case scenario w^* means that for the fixed x , the optimal second-stage cost is maximized over W .

Other criteria may give different robust formulations. For example, the minimum nominal cost formulation and min-max regret formulation.

Definition C.6. (Minimum nominal cost criterion) An optimal solution under the minimum nominal cost criterion is a pair of here-and-now decision $x \in X_R$ and wait-and-see decision y^0 corresponding to the nominal scenario $w^0 = 0$, such that the total cost in scenario w^0 is minimal.

The minimum nominal cost criterion leads to the following robust formulation

$$\begin{aligned} \min \quad & c^T x + d^T y \\ \text{s.t.} \quad & x \in X_R \\ & y \in Y(x, w^0) \end{aligned} \quad (\text{C.45})$$

where robustness is guaranteed by X_R .

To explain the concept of regret, the minimum perfect-information total cost is

$$C_P(w) = \min \{c^T x + d^T y \mid x \in X, y \in Y(x, w)\}$$

where w is known to the decision maker. For a fixed first-stage decision x , the maximum regret is defined as

$$\text{Reg}(x) = \max_{w \in W} \left\{ \min_{y \in Y(x, w)} \{c^T x + d^T y\} - C_P(w) \right\}$$

Definition C.7. (Min-max regret criterion) An optimal solution under the min-max regret criterion is a pair of here-and-now decision x and wait-and-see decision $y(w)$, such that the worst-case regret under all possible scenarios $w \in W$ is minimized.

The min-max regret cost criterion leads to the following robust formulation

$$\min_{x \in X} \left\{ c^T x + \max_{w \in W} \left\{ \min_{y \in Y(x, w)} d^T y - \min_{x' \in X, y' \in Y(x, w)} \{c^T x' + d^T y'\} \right\} \right\} \quad (\text{C.46})$$

In an ARO problem, we can naturally assume that the uncertainty set is a polyhedron. To see this, if x is a robust solution under an uncertainty set consists of discrete scenarios, i.e., $W = \{w^1, w^2, \dots, w^S\}$, according to Definition C.4, there exist corresponding $\{y^1, y^2, \dots, y^S\}$ such that

$$\begin{aligned} By^1 &\leq b - Ax - Cw^1 \\ By^2 &\leq b - Ax - Cw^2 \\ &\vdots \\ By^S &\leq b - Ax - Cw^S \end{aligned}$$

For non-negative weighting parameters $\lambda_1, \lambda_2, \dots, \lambda_S \geq 0$, $\sum_{s=1}^S \lambda_s = 1$, we have

$$\sum_{s=1}^S \lambda_s (By^s) \leq \sum_{s=1}^S \lambda_s (b - Ax - Cw^s)$$

or equivalently

$$B \sum_{s=1}^S \lambda_s y^s \leq b - Ax - C \sum_{s=1}^S \lambda_s w^s$$

indicating that for any $w = \sum_{s=1}^S \lambda_s w^s \in \text{conv}(\{w^1, w^2, \dots, w^S\})$, the wait-and-see decision $y = \sum_{s=1}^S \lambda_s y^s$ can recover all constraints, and thus $Y(x, w) \neq \emptyset$. This property inspires the following proposition that is in analogy to Proposition C.2

Proposition C.3. *Suppose x is a robust feasible solution for a discrete uncertainty set $\{w^1, w^2, \dots, w^S\}$, then it remains robust feasible if we replace the uncertainty set with its convex hull.*

Proposition C.3 also implies that in order to ensure the robustness of x , it is sufficient to consider the extreme points of a bounded polytope. Suppose the vertices of the polyhedral uncertainty set are w^s , $s = 1, 2, \dots, S$. Consider the following set

$$\Xi = \{x, y^1, y^2, \dots, y^S \mid Ax + By^s \leq b - Cw^s, s = 1, 2, \dots, S\} \quad (\text{C.47})$$

Robust feasible region X_R is the projection of polyhedron Ξ on x -space, which is also a polyhedron (Theorem B.2.5 in [10]).

Proposition C.4. *If the uncertainty set has a finite number of extreme points, set X_R is a polytope.*

Despite the nice theoretical properties, it is still difficult to solve an ARO problem in its general form (C.42). There have been considerable efforts spent on developing different approximations and approaches to tackle the computational challenges. We leave the solution methods of ARO problems to the

next subsection. Here we demonstrate the benefit from postponing some decisions to the second stage via a simple example taken from [1].

Consider an uncertain LP

$$\begin{aligned} \min_x \quad & x_1 \\ \text{s.t.} \quad & x_2 \geq 0.5\xi x_1 + 1 \quad (a_\xi) \\ & x_1 \geq (2 - \xi)x_2 \quad (b_\xi) \\ & x_1 \geq 0, \quad x_2 \geq 0 \quad (c_\xi) \end{aligned}$$

where $\xi \in [0, \rho]$ is an uncertain parameter and ρ is a constant (level of uncertainty) which may take a value in the open interval $(0, 1)$.

In a static setting, both x_1 and x_2 must be independent of ξ . When $\xi = \rho$, constraint (a_ξ) suggests $x_2 \geq 0.5\rho x_1 + 1$; when $\xi = 0$, constraint (b_ξ) indicates $x_1 \geq 2x_2$; as a result, we arrive at the conclusion $x_1 \geq \rho x_1 + 2$, so the optimal value in the static case satisfies

$$\text{Opt} \geq x_1 \geq \frac{2}{1 - \rho}$$

Thus the optimal value tends to infinity when ρ approaches 1.

Now consider the adjustable case, in which x_2 is a wait-and-see decision. Let $x_2 = 0.5\xi x_1 + 1$, (a_ξ) is always satisfied; substituting x_2 in constraint (b_ξ) yields:

$$x_1 \geq (2 - \xi)\left(\frac{1}{2}\xi x_1 + 1\right), \quad \forall \xi \in [0, \rho]$$

Substituting $x_1 = 4$ into above inequality we have

$$4 \geq 2(2 - \xi)\xi + 2 - \xi, \quad \forall \xi \in [0, \rho]$$

This inequality can be certified by the fact that $\xi \geq 0$ and $\xi(2 - \xi) \leq 1$, $\forall \xi \in \mathbb{R}$, indicating that $x_1 = 4$ is a robust feasible solution. Therefore, the optimal value should be no greater than 4 in the adjustable case for any ρ . The difference of optimal values in two cases can go arbitrarily large, depending on the value of ρ .

C.2.2 Affine Policy Based Approximation Model

ARO problem (C.42) is difficult to solve because the functional dependence of the wait-and-see decision on w is arbitrary, and there lacks a closed-form formula to characterize the optimal solution function $y(w)$ or certify whether $Y(x, w)$ is empty or not. At this point, we consider to approximate the recourse function $y(w)$ using a simpler one, naturally, an affine function

$$y(w) = y^0 + Gw \quad (\text{C.48})$$

where y^0 is the action in the second stage for the nominal scenario $w = 0$, and G is the gain matrix to be designed. (C.48) is called a linear decision rule or affine policy. It explicitly characterizes the wait-and-see decisions as an affine function in the revealed uncertain data. The rationality for employing an affine policy instead of other parametric ones is that it yields computationally tractable robust counterpart reformulations. This finding is firstly reported in [19].

To validate (C.44) under the linear decision rule, substituting (C.48) in (C.43)

$$Ax + By^0 + (BG + C)w \leq b, \quad \forall w \in W \quad (\text{C.49})$$

In (C.49), decision variables are x , y^0 , and G , which should be made before w is known, and thus are here-and-now decisions. The wait-and-see decision (or the incremental part) is naturally determined from (C.48) without further optimization, and cost reduction is considered in the determination of gain matrix G . (C.49) is in form of (C.6), and hence its robust counterpart can be derived via the methods in Appendix C.1.2. Here we just provide the results of polyhedral uncertainty as an example.

Suppose the uncertainty set is described by

$$W = \{w \mid Sw \leq h\}$$

If we assume that y^0 is the optimal second stage decision when $w = 0$, then we have

$$Ax + By^0 \leq b$$

Furthermore, (C.49) must hold if

$$\max_{w \in W} (BG + C)_i w \leq 0, \quad \forall i \quad (\text{C.50})$$

where $(\cdot)_i$ stands for the i -th row of the input matrix. According to LP duality theory,

$$\max_{w \in W} (BG + C)_i w = \min_{\Lambda_i \in \Pi_i} \Lambda_i h, \quad \forall i \quad (\text{C.51})$$

where Λ is a matrix consists of the dual variables, Λ_i is the i -th row of Λ and also the dual variable of the i -th LP in (C.51), and the set

$$\Pi_i = \{\Lambda_i \mid \Lambda_i \geq 0, \Lambda_i S = (BG + C)_i\}$$

is the feasible region of the i -th dual LP.

The minimization operator in the right-hand side of (C.51) can be omitted if the objective is to seek a minimum. Moreover, if we adopt the minimum nominal cost criterion, the ARO problem with a linear decision rule in the second stage can be formulated as an LP

$$\begin{aligned}
\min \quad & c^T x + d^T y^0 \\
\text{s.t.} \quad & Ax + By^0 \leq b, \Lambda h \leq 0 \\
& \Lambda \geq 0, \Lambda S = BG + C
\end{aligned} \tag{C.52}$$

In (C.52), decision variables are vectors x and y^0 , gain matrix G and dual matrix Λ . The constraints actually constitute a lifted formulation for X_R in (C.44). If the min-max cost criterion is employed, the objective can be transformed into a linear inequality constraint with uncertainty via an epigraph form, whose robust form can be derived using similar procedures shown above.

Affine policy based method is attractive because it reduces the conservatism in the static RO approach by incorporating corrective actions, and sustains computational tractability. In theory, the affine assumption more or less restricts the adaptability in the recourse stage. Nevertheless, research work in [20, 21, 22] shows that linear decision rules are indeed optimal or near optimal for many practical problems.

For more information on other decision rules and their reformulations, please see [1] (Chapter 14.3) for the quadratic decision rule, [23] for the extended linear decision rule, [24, 25] for the piecewise constant decision rule (finite adaptability), [26, 27] for the piecewise linear decision rule, and [28] for generalized decision rules. The methods in [24, 27] can be used to cope with integer wait-and-see decision variables. See also [4].

C.2.3 Algorithms for Fully Adjustable Models

Fully adjustable models are generally NP-hard [29]. To find the solution in Definition C.2, the model is decomposed into a master problem and a subproblem, which are solved iteratively, and a sequence of lower bound and upper bound of the optimal values are generated, until they get close enough to each other. To explain the algorithm for ARO problems, we discuss two instances.

1. Second-stage problem is an LP

Now we consider problem (C.42) without specific functional assumptions on the wait-and-see variables. We start from the second-stage LP with fixed x and w :

$$\begin{aligned}
\min_{y^0} \quad & d^T y^0 \\
\text{s.t.} \quad & By \leq b - Ax - Cw : u
\end{aligned} \tag{C.53}$$

where u is the dual variable, and the dual LP of (C.53) is

$$\begin{aligned}
& \max_u \quad u^T(b - Ax - Cw) \\
& \text{s.t.} \quad B^T u = d, \quad u \leq 0
\end{aligned} \tag{C.54}$$

If the primal LP (C.53) has a finite optimum, the dual LP (C.54) is also feasible and has the same optimum; otherwise, if (C.53) is infeasible, then (C.54) will be unbounded. Sometimes, an improper choice of x indeed leads to an infeasible second-stage problem. To detect infeasibility, consider the following LP with slack variables

$$\begin{aligned}
& \min_{y,s} \quad 1^T s \\
& \text{s.t.} \quad s \geq 0 \\
& \quad \quad By - Is \leq b - Ax - Cw : u
\end{aligned} \tag{C.55}$$

Its dual LP is

$$\begin{aligned}
& \max_u \quad u^T(b - Ax - Cw) \\
& \text{s.t.} \quad B^T u = 0, \quad -1 \leq u \leq 0
\end{aligned} \tag{C.56}$$

(C.55) and (C.56) are always feasible and have the same finite optimum. If the optimal value is equal to 0, then LP (C.53) is feasible; otherwise, if the optimal value is strictly positive, then LP (C.53) is infeasible.

For notation brevity, define feasible sets for the dual variable

$$\begin{aligned}
U_O &= \{u \mid B^T u = d, \quad u \leq 0\} \\
U_F &= \{u \mid B^T u = 0, \quad -1 \leq u \leq 0\}
\end{aligned}$$

The former one is associated with the dual form (C.54) of the second-stage optimization problem (C.53); the latter one corresponds to the dual form (C.56) of the second-stage feasibility test problem (C.55).

Next, we proceed to the middle level with fixed x :

$$R(x) = \max_{w \in W} \min_{y \in Y(x,w)} d^T y \tag{C.57}$$

which is a linear max-min problem that identifies the worst-case uncertainty. If LP (C.53) is feasible for an arbitrarily given value of $w \in W$, then we conclude $x \in X_R$ defined in (C.44); otherwise, if LP (C.53) is infeasible for some $w \in W$, then $x \notin X_R$ and $R(x) = +\infty$.

To check whether $x \in X_R$ or not, we investigate the following problem

$$\begin{aligned}
& \max_w \quad \min_{y,s} 1^T s \\
& \text{s.t.} \quad w \in W, \quad s \geq 0 \\
& \quad \quad By - Is \leq b - Ax - Cw : u
\end{aligned} \tag{C.58}$$

It maximizes the minimum of (C.55) over all possible values of $w \in W$. Since the minimums of (C.55) and (C.56) are equal, problem (C.58) is equivalent to maximizing the optimal value of (C.56) over the uncertainty set W , leading to a bilinear program

$$\begin{aligned} r(x) = \max_{u,w} \quad & u^T(b - Ax - Cw) \\ \text{s.t.} \quad & w \in W, \quad u \in U_F \end{aligned} \quad (\text{C.59})$$

Because both W and U_F are bounded, (C.59) must have a finite optimum. Clearly, $0 \in U_F$, so $r(x)$ must be non-negative. In fact, if $r(x) = 0$, then $x \in X_R$; if $r(x) > 0$, then $x \notin X_R$. With the duality transformation, the opposite optimization operators in (C.58) come down to a traditional NLP.

For similar reasons, by replacing the second-stage LP (C.53) with its dual LP (C.54), problem (C.57) is equivalent to the following bilinear program

$$\begin{aligned} r(x) = \max_{u,w} \quad & u^T(b - Ax - Cw) \\ \text{s.t.} \quad & w \in W, \quad u \in U_O \end{aligned} \quad (\text{C.60})$$

The fact that a linear max-min problem can be transformed as a bilinear program using LP duality is reported in [30]. Bilinear programs can be locally solved by general purpose NLP solvers, but the non-convexity prevents a global optimal solution from being found easily. In what follows, we introduce some methods that exploit specific features of the uncertainty set and are widely used by the research community. In view that (C.59) and (C.60) only differ in the dual feasibility set, we will use set U to refer either U_F or U_O in the unified solution method.

a. General polytope

Suppose that the uncertainty set is described by

$$W = \{w \mid Sw \leq h\}$$

An important feature in (C.59) and (C.60) is that the constraint set W and U are separated and there is no constraint that involves w and u simultaneously, so the bilinear program can be considered in the following format

$$\begin{aligned} \max_{u \in U} \quad & u^T(b - Ax) + \max_w \quad (-u^T Cw) \\ \text{s.t.} \quad & Sw \leq h : \xi \end{aligned} \quad (\text{C.61})$$

The bilinear term $u^T Cw$ is non-convex. If we treat the second part $\max_{w \in W} (-u^T Cw)$ as an LP in w where u is a parameter, whose KKT optimality condition is given by

$$\begin{aligned} 0 &\leq \xi \perp h - Sw \leq 0 \\ S^T \xi + C^T u &= 0 \end{aligned} \quad (\text{C.62})$$

The stationary point of LCP (C.62) gives the optimal primal and dual solutions simultaneously. As the uncertainty set is a bounded polyhedron, the optimal solution must be bounded, and strong duality holds, so we can replace $-u^T Cw$ in the objective with a linear term $h^T \xi$ and additional constraints in (C.62). Moreover, the complementarity and slackness condition in (C.62) can be linearized via the method in Appendix B.3.5. In summary, problem (C.61) can be solved via an equivalent MILP

$$\begin{aligned}
& \max_{u, w, \xi} \quad u^T(b - Ax) + h^T \xi \\
& \text{s.t.} \quad u \in U, \theta \in \{0, 1\}^m \\
& \quad S^T \xi + C^T u = 0 \\
& \quad 0 \leq \xi \leq M(1 - \theta) \\
& \quad 0 \leq h - Sw \leq M\theta
\end{aligned} \tag{C.63}$$

where m is the dimension of θ , and M is a large enough constant. Compared with (C.61), non-convexity migrates from the objective function to the constraints with binary variables. The number of binary variables in (C.63) only depends on the number of constraints in set W , and is independent of the dimension of x .

Another heuristic method for bilinear programs in the form of (C.59) and (C.60) is the mountain climbing method in [31], which is summarized in Algorithm C.1

Algorithm C.1 : Mountain climbing

- 1: Choose a convergence tolerance $\varepsilon > 0$, and an initial $w^* \in W$;
- 2: Solve the following LP with current w^*

$$R_1 = \max_{u \in U} u^T(b - Ax - Cw^*) \tag{C.64}$$

The optimal solution is u^* and the optimal value is R_1 ;

- 3: Solve the following LP with current u^*

$$R_2 = \max_{w \in W} (b - Ax - Cw)^T u^* \tag{C.65}$$

The optimal solution is w^* and the optimal value is R_2 ;

- 4: If $R_2 - R_1 \leq \varepsilon$, report the optimal value R_2 as well as the optimal solution w^*, u^* , and terminate; otherwise, go to step 2.
-

The optimal solutions of LPs must be found at one of the vertices of its feasible region, hence $w^* \in \text{vert}(W)$ and $u^* \in \text{vert}(U)$ hold. As its name implies, the sequence of objective values generated by Algorithm C.1 is monotonically increasing, until a local maximum is found [31]. The convergence

is guaranteed by the finiteness of $\text{vert}(U)$ and $\text{vert}(W)$. If we try multiple initial points that are chosen elaborately and pick up the best one among the returned results, the solution quality is often satisfactory. The key point is, these initial points should span along most directions in the w -subspace. For example, one may search the $2m$ points on the boundary of W in directions $\pm e_i^m$, $i = 1, 2, \dots, m$, where m is the dimension of w , and e_i^m is the i -th column of an $m \times m$ identity matrix. As LPs can be solved very efficiently, Algorithm C.1 is especially suitable for the instances with very complicated U and W , and usually outperforms general NLP solvers for bilinear programs with disjoint constraints.

Algorithm C.1 is also valid if W is other convex set, say, an ellipsoid, and converges to a local optimum in a finite number of iterations for a given precision [32].

b. Cardinality constrained uncertainty set

A continuous cardinality constrained uncertainty set in the form of (C.14) is a special class of the polyhedral case, see the transformation in (C.15). Therefore, the previous method can be applied, and the number of inequalities in the polyhedral form is $3m + 1$, which is equal to the number of binary variables in MILP (C.63). As revealed in Proposition C.3, for a polyhedral uncertainty set, we can merely consider the extreme points.

Consider a discrete cardinality constrained uncertainty set

$$W = \left\{ w \left| \begin{array}{l} w_j = w_j^0 + w_j^+ z_j^+ - w_j^- z_j^-, \forall j \\ \exists z^+, z^- \in Z \end{array} \right. \right\} \quad (\text{C.66a})$$

$$Z = \left\{ z^+, z^- \left| \begin{array}{l} z^+, z^- \in \{0, 1\}^m \\ z_j^+ + z_j^- \leq 1, \forall j \\ 1^T(z^+ + z^-) \leq \Gamma \end{array} \right. \right\} \quad (\text{C.66b})$$

where the budget of uncertainty $\Gamma \leq m$ is an integer. In (C.66a), each element w_j takes one of three possible values: w_j^0 , $w_j^0 + w_j^+$, and $w_j^0 - w_j^-$, and at most Γ of the m elements w_j can take a value that is not equal to w_j^0 . If the forecast error is symmetric, i.e., $w_j^+ = w_j^-$, then (C.66) is called symmetric as the nominal scenario locates at the center of W . We discuss this case separately because this representation allows to linearize the non-convexity in (C.59) and (C.60) with fewer binary variables.

Expanding the bilinear term $u^T C w$ in an element-wise form

$$u^T C w = u^T C w^0 + \sum_i \sum_j (c_{ij} w_j^+ u_i z_j^+ - c_{ij} w_j^- u_i z_j^-)$$

where c_{ij} is the element of matrix C . Let

$$v_{ij}^+ = u_i z_j^+, \quad v_{ij}^- = u_i z_j^-, \quad \forall i, \forall j$$

the bilinear term can be expressed via a linear function. The product involving a binary variable and a continuous variable can be linearized via the method illuminated in Appendix B.2.2.

In conclusion, bilinear subproblems (C.59) and (C.60) can be solved via MILP

$$\begin{aligned}
\max \quad & u^T(b - Ax) - u^T C w^0 - \sum_i \sum_j (c_{ij} w_j^+ v_{ij}^+ - c_{ij} w_j^- v_{ij}^-) \\
\text{s.t.} \quad & u \in U, \{z^+, z^-\} \in Z \\
& 0 \leq v_{ij}^+ - u_j \leq M(1 - z_j^+), -M z_j^+ \leq v_{ij}^+ \leq 0, \forall i, \forall j \\
& 0 \leq v_{ij}^- - u_j \leq M(1 - z_j^-), -M z_j^- \leq v_{ij}^- \leq 0, \forall i, \forall j
\end{aligned} \tag{C.67}$$

where $M = 1$ for problem (C.59) since $-1 \leq u \leq 0$, and M is a sufficiently large number for problem (C.60), because there is no clear bounds for the dual variable u . The number of binary variables in MILP (C.67) is $2m$, which is less than that in (C.63) if the uncertainty set is replaced by its convex hull. The number of additional continuous variables v_{ij}^+ and v_{ij}^- is also moderate since the matrix C is sparse.

Finally, we are ready to give the decomposition algorithm which is proposed in [33]. In light of Proposition C.3, it is sufficient to consider the extreme points w^1, w^2, \dots, w^S in the uncertainty set, inspiring the following epigraph formulation which is equivalent to (C.42)

$$\begin{aligned}
\min_{x, y^s, \eta} \quad & c^T x + \eta \\
\text{s.t.} \quad & x \in X \\
& \eta \geq d^T y^s, \forall s \\
& Ax + B y^s \leq b - C w^s, \forall s
\end{aligned} \tag{C.68}$$

Recall (C.47), the last constraint is in fact a lifted formulation for X_R . For polytope and cardinality constrained uncertainty sets, the number of extreme points are finite, but may grow exponentially in the dimension of uncertainty. Actually, it is difficult and also unnecessary to enumerate every extreme point, because most of them actually provide redundant constraints. A smart method is to identify active scenarios which contribute binding constraints in X_R . This motivation has been widely used in complex optimization problems and formalized in Sect. C.1.3. The procedure of the adaptive scenario generation algorithm for ARO is summarized in Algorithm C.2.

Algorithm C.2 converges in a finite number of iterations, which is bounded by the number of extreme points of the uncertainty set. In practice, this algorithm often converges in a few iterations, because problems (C.59) and (C.60) always identify the most critical scenario that should be considered. This is why we name the algorithm “adaptive scenario generation”. It is called “constraint-and-column generation algorithm” in [33], because the number-

Algorithm C.2 : Adaptive scenario generation

-
- 1: Choose a tolerance $\varepsilon > 0$, set $LB = -\infty$, $UB = +\infty$, iteration index $k = 0$, and the critical scenario set $O = w^0$;
 - 2: Solve the following master problem

$$\begin{aligned}
& \min_{x, y^s, \eta} \quad c^T x + \eta \\
& \text{s.t.} \quad x \in X, \quad \eta \geq d^T y^s, \quad s = 0, \dots, k \\
& \quad \quad Ax + By^s \leq b - Cw^s, \quad \forall w^s \in O
\end{aligned} \tag{C.69}$$

- The optimal solution is x^{k+1} , η^{k+1} , and update $LB = c^T x^{k+1} + \eta^{k+1}$;
- 3: Solve bilinear feasibility testing problem (C.59) with x^{k+1} , the optimal solution is w^{k+1} , u^{k+1} ; if the optimal value $r^{k+1} > 0$, update $O = O \cup w^{k+1}$, and add a scenario cut

$$\eta \geq d^T y^{k+1}, \quad Ax + By^{k+1} \leq b - Cw^{k+1} \tag{C.70}$$

- with a new variable y^{k+1} to the master problem (C.69), update $k \leftarrow k+1$, and go to Step 2;
- 4: Solve bilinear optimality testing problem (C.60) with x^{k+1} , the optimal solution is w^{k+1} , u^{k+1} , and the optimal value is R^{k+1} ; update $O = O \cup w^{k+1}$ and $UB = c^T x^{k+1} + R^{k+1}$, create scenario cut (C.70) with a new variable y^{k+1} .
 - 5: If $UB - LB \leq \varepsilon$, report the optimal solution, terminate; otherwise, add the scenario cut in step 4 to the master problem (C.69), update $k \leftarrow k+1$, and go to step 2;
-

s of decision variables (columns) and constraints increase simultaneously. Please note that the scenario cut streamlines the feasibility cut and optimality cut used in the existing literature.

Bilinear subproblems (C.59) and (C.60) can be solved by the methods discussed previously, according to the form of the uncertainty set. In Algorithm C.2, we utilize w to create scenario cuts, which are also called primal cuts. In fact, the optimal dual variable u of (C.59) and (C.60) provides sensitivity information, and can be used to construct dual cuts, which is a single inequality in the first-stage variable x . See Benders decomposition algorithm in [29]. Since scenario cuts are much tighter than Benders cuts, Algorithm C.2 is the most prevalent method for solving ARO problems.

If matrix A is uncertainty-dependent, the scenario constraints in the master problem (C.69) becomes $A(w^s)x + By^s \leq b - Cw^s$, $\forall w^s$, where $A(w^s)$ is constant but varies in different scenarios; the objective function of bilinear subproblems changes to $u^T[b - A(w)x - Cw]$, where x is given in the subproblem. If A can be expressed as a linear function in w , the problem structure remains the same, and previous methods are still valid. Even if the

second-stage problem is an SOCP, the adaptive scenario generation framework remains applicable, and the key procedure is to solve a max-min SOCP. Such a model originates from the robust operation of a power distribution network with uncertain generation and demand. By dualizing the inner-most SOCP, the max-min SOCP is cast as a bi-convex program, which can be globally or locally solved via an MISOCP or the mountain climbing method. An example can be found in Sect. ??.

Recently, the duality theory of fully-adjustable robust optimization problem has been proposed in [34]. It has been shown that this kind of problem is self-dual, i.e., the dual problem remains an ARO. However, solving the dual problem may enjoy better efficiency. An extended CCG algorithm which always produces a feasible first-stage decision (if one exists) is proposed in [35].

2. Second-stage problem is an MILP

Now we consider the case in which some of the wait-and-see decisions are discrete. As what can be observed from the previous case, the most important tasks in solving an ARO problem is to validate feasibility and optimality, which can boil down to solving a linear max-min problem. When the wait-and-see decisions are continuous and the second-stage problem is linear, LP duality theory is applied such that the linear max-min problem is cast as a traditional bilinear program. However, discrete variables appearing in the second stage make the recourse problem a mixed-integer linear max-min problem with a non-convex inner level, preventing the use of LP duality theory. As a result, validating feasibility and optimality becomes more challenging.

The compact form of an ARO problem with integer wait-and-see decisions can be written as

$$\min_{x \in X} \left\{ c^T x + \max_{w \in W} \min_{y, z \in Y(x, w)} d^T y + g^T z \right\} \quad (\text{C.71})$$

where z is binary and depends on the exact value of w ; the feasible region

$$Y(x, w) = \left\{ y, z \left| \begin{array}{l} By + Gz \leq b - Ax - Cw \\ y \in \mathbb{R}^{m_1}, z \in \Phi \end{array} \right. \right\}$$

where feasible set $\Phi = \{z | z \in \mathbb{B}^{m_2}, Tz \leq v\}$; m_1 and m_2 are dimensions of y and z ; T and v are constant coefficients; all coefficient matrices have compatible dimensions. We assume that the uncertainty set W can be represented by a finite number of extreme points. This kind of problem is studied in [36]. A nested constraint-and-column generation algorithm is proposed.

Different from the mainstream idea that directly solves a linear max-min program as a bilinear program, the mixed-integer max-min program in (C.71) is expanded to a tri-level problem

$$\begin{aligned}
& \max_{w \in W} \min_{z \in \Phi} g^T z + \min_y d^T y \\
& \text{s.t. } By \leq b - Ax - Cw - Gz
\end{aligned} \tag{C.72}$$

For the ease of discussion, we assume all feasible sets are bounded, because decision variables of practical problems have physical bounds. By replacing the innermost LP in variable y with its dual LP, problem (C.72) becomes

$$\max_{w \in W} \left\{ \min_{z \in \Phi} \left\{ g^T z + \max_{u \in U} u^T (b - Ax - Cw - Gz) \right\} \right\} \tag{C.73}$$

where u is the dual variable, and set $U = \{u \mid u \leq 0, B^T u = d\}$. Because both w and z are expressed via binary variables, bilinear terms $u^T Cw$ and $u^T Gz$ have linear representations by using the method in Appendix B.2.2. Since Φ has a countable number of elements, problem (C.73) (in its linearized version) has the same form as ARO problem (C.42), and can be solved by Algorithm C.2. More exactly, write (C.73) into an epigraph form by enumerating all possible elements $z \in \Phi$, then perform Algorithm C.2 and identify binding elements. In this way, the minimization operator in the middle level is eliminated.

The nested adaptive scenario generation algorithm for ARO problem (C.71) with mixed-integer recourse is summarized in Algorithm C.3. Because both W and Φ are finite sets with countable elements, Algorithm C.3 converges in a finite number of iterations. Notice that we do not distinguish feasibility and optimality subproblems in above algorithm due to their similarities. One can also introduce slack here-and-now variables in the second stage and penalty terms in the objective function, such that the recourse problem is always feasible. It should be pointed out that Algorithm C.3 incorporates double loops, and an MILP should be solved in each iteration in the inner loop, so we'd better not expect too much on its efficiency. Nonetheless, it is the first systematic method to solve an ARO problem with integer variables in the second stage. Another concept which should be clarified is that although the second-stage discrete variable z is treated as scenario and enumerated on the fly when solving problem (C.73) in step 3 (the inner loop), it is a decision variable of the master problem (C.74) in the outer loop.

As a short conclusion, to overcome the limitation of traditional static RO approaches which require all decisions should be made without exact information on the underlying uncertainty, ARO employs a two-stage decision-making framework and allows a subset of decision variables to be made after the uncertain data are revealed. Under some special decision rules, computational tractability can be preserved. In fully adjustable cases, the ARO problem can be solved by a decomposition algorithm. The subproblem comes down to a (mixed-integer) linear max-min problem, which is generally challenging to solve. We introduce MILP reformulations for special classes of uncertainty sets, which are compatible with commercial solvers, and help solve an engineering optimization problem in a systematic way.

Algorithm C.3 : Nested adaptive scenario generation

-
- 1: Choose a tolerance $\varepsilon > 0$, set $LB = -\infty$, $UB = +\infty$, iteration index $k = 0$, and the critical scenario set $O = w^0$;
 - 2: Solve the following master problem

$$\begin{aligned}
& \min_{x, y, z, \eta} \quad c^T x + \eta \\
& \text{s.t.} \quad x \in X \\
& \quad \eta \geq d^T y^s + g^T z^s, \quad z^s \in \Phi, \quad s = 0, \dots, k \\
& \quad Ax + By^s + Gz^s \leq b - Cw^s, \quad \forall w^s \in O
\end{aligned} \tag{C.74}$$

- The optimal solution is x^{k+1} , η^{k+1} , and update $LB = c^T x^{k+1} + \eta^{k+1}$;
- 3: Solve problem (C.73) with x^{k+1} , the optimal solution is $(z^{k+1}, w^{k+1}, u^{k+1})$, and optimal value is R^{k+1} ; update $O = O \cup w^{k+1}$, $UB = \min\{UB, c^T x^{k+1} + R^{k+1}\}$, create new variables (y^{k+1}, z^{k+1}) and scenario cuts

$$\begin{aligned}
& \eta \geq d^T y^{k+1} + g^T z^{k+1}, \quad z^{k+1} \in \Phi \\
& Ax + By^{k+1} + Gz^{k+1} \leq b - Cw^{k+1}
\end{aligned} \tag{C.75}$$

- 4: If $UB - LB \leq \varepsilon$, terminate and report the optimal solution and optimal value; otherwise, add scenario cuts (C.75) to the master problem (C.74), update $k \leftarrow k + 1$, and go to step 2;
-

C.3 Distributionally Robust Optimization

Static and adjustable RO models presented in Sect. C.1 and Sect. C.2 do not rely on specifying probability distributions of the uncertain data, which are used in SO approaches for generating scenarios, evaluating probability of constraint violation, or deriving analytic solutions for some specific problems. Instead, RO design principle aims to cope with the worst-case scenario in a pre-defined uncertainty set in the space of uncertain variables, which is a salient distinction between these two approaches. If the exact probability distribution is precisely known, optimal solutions to SO models would be less conservative than the robust ones from the statistical perspective. However, the optimal solution to SO models could have poor statistical performances if the actual distribution is not identical to the designated one [37]. As for the RO approach, as it hedges against the worst-case scenario, which rarely happens in reality, the robust strategy could be conservative thus suboptimal in most cases.

A method which aims to build a bridge connecting SO and RO approaches is the DRO, whose optimal solutions are designed for the worst-case probability distribution within a family of candidate distributions, which are described by statistic information, such as moments, and structure properties, including

symmetry, unimodality, and so on. This approach is generally less conservative than the traditional RO because dispersion effect of uncertainty is taken into account, i.e., the probability of an extreme event is low. Meanwhile, the statistic performances of the solution is less sensitive to the perturbation in probability distributions than that of an SO model, as it hedges against the worst distribution. Publications on this method have been proliferating rapidly in the past few years. This section only sheds light on some most representative methods which have been used in energy system studies.

C.3.1 Static Distributionally Robust Optimization

In analogy with the terminology used in Sect C.1, “static” means that all decision variables are here-and-now type. Theoretical outcomes in this part mainly come from [38]. A static DRO problem can be formulated as

$$\begin{aligned} \min_x \quad & c^T x \\ \text{s.t.} \quad & x \in X \\ & \Pr(a_i(\xi)^T x \leq b_i(\xi), i = 1, \dots, m) \geq 1 - \varepsilon, \forall f(\xi) \in \mathcal{P} \end{aligned} \quad (\text{C.76})$$

where x is the decision variable, X is a closed and convex set that is independent of the uncertain parameter, c is a deterministic vector, and ξ is the uncertain data, whose probability density function $f(\xi)$ is not known exactly, and belongs to \mathcal{P} , a set comprised of candidate distributions. Robust chance constraint in (C.76) requires a finite number of linear inequalities depending on ξ to be met with a probability of at least $1 - \varepsilon$, regardless of the true probability density function of ξ . We assume uncertain coefficients a_i and b_i are linear functions in ξ , i.e.

$$\begin{aligned} a_i(\xi) &= a_i^0 + \sum_{j=1}^k a_i^j \xi_j \\ b_i(\xi) &= b_i^0 + \sum_{j=1}^k b_i^j \xi_j \end{aligned}$$

where a_i^0, a_i^j are constant vectors and b_i^0, b_i^j are constant scalars. Define

$$y_i^j(x) = (a_i^j)^T x - b_i^j, \forall i, \forall j$$

the chance constraint in (C.76) can be expressed via

$$\Pr(y_i^0(x) + y_i(x)^T \xi \leq 0, i = 1, \dots, m) \geq 1 - \varepsilon, \forall f(\xi) \in \mathcal{P} \quad (\text{C.77})$$

where vector $y_i(x) = [y_i^1(x), \dots, y_i^k(x)]^T$ is affine in x . Since the objective is certain and constraint violation is bounded by a small probability, problem (C.76) is also called a robust chance-constrained program.

Chance constraints can be transformed into tractable ones that are convex in variable x only for a few special cases. For example, if ξ follows a Gaussian distribution, $\varepsilon \leq 0.5$, and $m = 1$, then the individual chance constraint without distribution uncertainty is equivalent to a single SOC constraint [39]. For $m > 1$, joint chance constraints form convex feasible region when the right-hand side terms $b_i(\xi)$ are uncertain and follow a log-concave distribution [38, 40], while coefficients a_i , $i = 1, \dots, m$ are deterministic.

Constraint (C.77) is even more challenging at first sight: not only the random vector ξ , but also the probability distribution function $f(\xi)$ itself is uncertain. Because in many practical situations, probability distribution must be estimated from enough historical data, which may not be available at hand. Typically, one may only have access to some statistical indicators about $f(\xi)$, e.g. its mean value, covariance, and support set. Using a specific $f(\xi) \in \mathcal{P}$ may lead to over-optimistic solutions which fail to satisfy the probability guarantee under the true distribution.

Similar to the paradigm in static RO, a prudent way to immunize a chance constraint against uncertain probability distribution is to investigate the situation in the worst case, inspiring the following distributionally robust chance constraint, which is equivalent to (C.77)

$$\inf_{f(\xi) \in \mathcal{P}} \Pr(y_i^0(x) + y_i(x)^T \xi \leq 0, i = 1, \dots, m) \geq 1 - \varepsilon \quad (\text{C.78})$$

Clearly, if x satisfies (C.78), the probability of constraint violation is upper bounded by ε for the true probability distribution of ξ .

This section introduces convex optimization models for approximating robust chance constraints under uncertain probability distributions, whose first- and second-order moments as well as the support set (or equivalently the feasible region) of random variable are known. More precisely, we let $\mathbb{E}_P(\xi) = \mu \in \mathbb{R}^k$ be the mean value and $\mathbb{E}_P((\xi - \mu)(\xi - \mu)^T) = \Sigma \in \mathbb{S}_{++}^k$ be the covariance matrix of random variable ξ under the true distribution P . We define the moment matrix

$$\Omega = \begin{bmatrix} \Sigma + \mu\mu^T & \mu \\ \mu^T & 1 \end{bmatrix}$$

for ease of notation.

To help readers understand the fundamental ideas in DRO, we briefly introduce the worst-case expectation problem, which will be used throughout this section. Recall that \mathcal{P} represents the set of all probability distributions on \mathbb{R}^k with mean vector μ and covariance matrix $\Sigma \succ 0$, the problem is formulated by

$$\theta_P^m = \sup_{f(\xi) \in \mathcal{P}} \mathbb{E}[(g(\xi))^+]$$

where $g : \mathbb{R}^k \rightarrow \mathbb{R}$ is a function of ξ ; $(g(\xi))^+$ means the maximum between 0 and $g(\xi)$. Write the problem into an integral format

$$\begin{aligned}
\theta_P^m = \sup_{f(\xi) \in \mathcal{P}} \quad & \int_{\xi \in \mathbb{R}^k} \max\{0, g(\xi)\} f(\xi) d\xi \\
\text{s.t.} \quad & f(\xi) \geq 0, \forall \xi \in \mathbb{R}^k \\
& \int_{\xi \in \mathbb{R}^k} f(\xi) d\xi = 1 : \lambda_0 \\
& \int_{\xi \in \mathbb{R}^k} \xi f(\xi) d\xi = \mu : \lambda \\
& \int_{\xi \in \mathbb{R}^k} \xi \xi^T f(\xi) d\xi = \Sigma + \mu \mu^T : \Lambda
\end{aligned} \tag{C.79}$$

In problem (C.79), the decision variables are the values of $f(\xi)$ over all possible $\xi \in \mathbb{R}^k$, so there are infinitely many decision variables, and problem (C.79) is an infinite-dimensional LP. The former two constraints enforce $f(\xi)$ to be a valid distribution function; the latter two ensure consistent first- and second-order moments. The optimal solution gives the worst-case distribution. However, it is difficult to solve (C.79) in its primal form. We now associate dual variables $\lambda_0 \in \mathbb{R}$, $\lambda \in \mathbb{R}^k$, and $\Lambda \in \mathbb{S}^k$ with each integral constraint, and the dual problem of (C.79) can be constructed following the duality theory of conic LP, which is given by

$$\begin{aligned}
\theta_D^m = \inf_{\lambda_0, \lambda, \Lambda} \quad & \lambda_0 + \mu^T \lambda + \text{tr}[\Lambda^T (\Sigma + \mu \mu^T)] \\
\text{s.t.} \quad & \lambda_0 + \xi^T \lambda + \text{tr}[\Lambda^T (\xi \xi^T)] \\
& \geq \max\{0, g(\xi)\}, \forall \xi \in \mathbb{R}^k
\end{aligned} \tag{C.80}$$

To understand this dual form in (C.80), we can image a discrete version of (C.79), in which ξ_1, \dots, ξ_n are sampled scenarios of the uncertain parameter, and their associated probabilities $f(\xi_1), \dots, f(\xi_n)$ are decision variables of (C.79). Moreover, if we replace the integral arithmetic in the constraints with the summation arithmetic, (C.79) comes down to a traditional LP, and its dual is also an LP, where the constraint becomes

$$\lambda_0 + \xi_i^T \lambda + \text{tr}[\Lambda^T (\xi_i \xi_i^T)] \geq \max\{0, g(\xi_i)\}, \quad i = 1, \dots, n$$

Let $n \rightarrow +\infty$ and ξ spread over \mathbb{R}^k , we can get the dual problem (C.80).

Unlike the primal problem (C.79) that has infinite decision variables, the dual problem (C.80) has finite variables and an infinite number of constraints. In fact, we are optimizing over the coefficients of a polynomial in ξ . Because $\Sigma \succ 0$, Slater condition is met, and thus strong duality holds (this conclusion can be found in many other literatures, such as [43]), i.e., $\theta_P^m = \theta_D^m$. In the following, we will eliminate ξ and reduce the constraint into convex ones in

dual variables λ_0 , λ , and Λ . Recall the definition of matrix Ω , the compact form of problem (C.80) can be expressed as

$$\begin{aligned} \inf_{M \in \mathbb{S}^{k+1}} \quad & \text{tr}[\Omega^T M] \\ \text{s.t.} \quad & [\xi^T \ 1] M [\xi^T \ 1]^T \geq 0, \forall \xi \in \mathbb{R}^k \\ & [\xi^T \ 1] M [\xi^T \ 1]^T \geq g(\xi), \forall \xi \in \mathbb{R}^k \end{aligned} \quad (\text{C.81})$$

where the matrix decision variable is

$$M = \begin{bmatrix} \Lambda & \frac{\lambda}{2} \\ \frac{\lambda^T}{2} & \lambda_0 \end{bmatrix}$$

and the first constraint is equivalent to an LMI $M \succeq 0$.

A special case of the worst-case expectation problem is

$$\theta_P^m = \sup_{f(\xi) \in \mathcal{P}} \Pr[\xi \in S] \quad (\text{C.82})$$

which quantifies the maximum probability of the event $\xi \in S$, where S is a Borel measurable set. This problem has a close relationship with generalized probability inequalities discussed in [43] and the generalized moments problem studied in [44]. By defining an indicator function as

$$\mathbb{I}_S(\xi) = \begin{cases} 1 & \text{if } \xi \in S \\ 0 & \text{otherwise} \end{cases}$$

The dual problem of (C.82) can be written as

$$\begin{aligned} \inf_{M \in \mathbb{S}^{k+1}} \quad & \text{tr}[\Omega^T M] \\ \text{s.t.} \quad & M \succeq 0, \quad [\xi^T \ 1] M [\xi^T \ 1]^T \geq 1, \forall \xi \in S \end{aligned} \quad (\text{C.83})$$

which is a special case of (C.81) when $g(\xi) = \mathbb{I}_S(\xi)$.

Next we present how to formulate a robust chance constraint (C.78) as convex constraints that can be recognized by convex optimization solvers.

1. Individual chance constraints

Consider a single robust chance constraint

$$\inf_{f(\xi) \in \mathcal{P}} \Pr(y^0(x) + y(x)^T \xi \leq 0) \geq 1 - \varepsilon \quad (\text{C.84})$$

The feasible set in x is denoted by X_R^S .

To eliminate the optimization over function $f(\xi)$, we leverage the concept of conditional value-at-risk (CVaR) introduced by [45]. For a given loss

function $L(\xi)$ and tolerance $\varepsilon \in (0, 1)$, the CVaR at level ε is defined as

$$\text{CVaR}(L(\xi), \varepsilon) = \inf_{\beta \in \mathbb{R}} \beta + \frac{1}{\varepsilon} \mathbb{E}_{f(\xi)} \left([L(\xi) - \beta]^+ \right) \quad (\text{C.85})$$

where the expectation is taken over a given probability distribution $f(\xi)$. CVaR is the conditional expectation of loss greater than the $(1 - \varepsilon)$ -quantile of the loss distribution. Indeed, condition

$$\Pr[L(\xi) \leq \text{CVaR}(L(\xi), \varepsilon)] \geq 1 - \varepsilon$$

holds regardless of the probability distribution and loss function $L(\xi)$ [38]. Therefore, to certify $\Pr(L(\xi) \leq 0) \geq 1 - \varepsilon$, a sufficient condition without probability evaluation is $\text{CVaR}(L(\xi), \varepsilon) \leq 0$, or more precisely:

$$\begin{aligned} & \sup_{f(\xi) \in \mathcal{P}} \text{CVaR}(y^0(x) + y(x)^T \xi, \varepsilon) \leq 0 \\ \implies & \inf_{f(\xi) \in \mathcal{P}} \Pr(y^0(x) + y(x)^T \xi \leq 0) \geq 1 - \varepsilon \end{aligned} \quad (\text{C.86})$$

According to (C.85), above worst-case CVaR can be expressed by

$$\begin{aligned} & \sup_{f(\xi) \in \mathcal{P}} \text{CVaR}(y^0(x) + y(x)^T \xi, \varepsilon) \\ &= \sup_{f(\xi) \in \mathcal{P}} \inf_{\beta \in \mathbb{R}} \left\{ \beta + \frac{1}{\varepsilon} \mathbb{E}_{f(\xi)} \left([y^0(x) + y(x)^T \xi - \beta]^+ \right) \right\} \\ &= \inf_{\beta \in \mathbb{R}} \left\{ \beta + \frac{1}{\varepsilon} \sup_{f(\xi) \in \mathcal{P}} \mathbb{E}_{f(\xi)} \left([y^0(x) + y(x)^T \xi - \beta]^+ \right) \right\} \end{aligned} \quad (\text{C.87})$$

The maximization and minimization operators are interchangeable because of the saddle point theorem in [46]. Recall previous analysis; the worst-case expectation can be computed from problem

$$\begin{aligned} & \inf_{\beta, M \in \mathbb{S}^{k+1}} \text{tr}[\Omega^T M] \\ & \text{s.t. } M \succeq 0, \\ & \begin{bmatrix} \xi^T & 1 \end{bmatrix} M \begin{bmatrix} \xi^T & 1 \end{bmatrix} \geq y^0(x) + y(x)^T \xi - \beta, \quad \forall \xi \in \mathbb{R}^k \end{aligned}$$

The semi-infinite constraint has a matrix quadratic form

$$\begin{bmatrix} \xi \\ 1 \end{bmatrix}^T \left(M - \begin{bmatrix} 0 & \frac{y(x)}{2} \\ \frac{y(x)^T}{2} & y^0(x) - \beta \end{bmatrix} \right) \begin{bmatrix} \xi \\ 1 \end{bmatrix} \geq 0, \quad \forall \xi \in \mathbb{R}^k$$

which is equivalent to

$$M - \begin{bmatrix} 0 & \frac{y(x)}{2} \\ \frac{y(x)^T}{2} & y^0(x) - \beta \end{bmatrix} \succeq 0$$

As a result, the worst-case CVaR can be calculated from an SDP

$$\begin{aligned} & \sup_{f(\xi) \in \mathcal{P}} \text{CVaR}(y^0(x) + y(x)^T \xi, \varepsilon) \\ &= \inf_{\beta, M} \beta + \frac{1}{\varepsilon} \text{tr}(\Omega^T M) \\ & \quad \text{s.t. } M \succeq 0 \\ & \quad M \succeq \begin{bmatrix} 0 & \frac{y(x)}{2} \\ \frac{y(x)^T}{2} & y^0(x) - \beta \end{bmatrix} \end{aligned} \quad (\text{C.88})$$

It is shown that the indicator \Rightarrow in (C.86) is in fact an equivalence \Leftrightarrow [38] in static DRO. In conclusion, robust chance constraint (C.84) can be written as a convex set in variable x , β , and M as follows

$$X_R^S = \left\{ x \left| \begin{array}{l} \exists \beta \in \mathbb{R}, M \succeq 0 \text{ such that} \\ \beta + \frac{1}{\varepsilon} \text{tr}(\Omega^T M) \leq 0 \\ M \succeq \begin{bmatrix} 0 & \frac{y(x)}{2} \\ \frac{y(x)^T}{2} & y^0(x) - \beta \end{bmatrix} \end{array} \right. \right\} \quad (\text{C.89})$$

2. Joint chance constraints

Now consider the joint robust chance constraints

$$\inf_{f(\xi) \in \mathcal{P}} \Pr(y_i^0(x) + y_i(x)^T \xi \leq 0, i = 1, \dots, m) \geq 1 - \varepsilon \quad (\text{C.90})$$

The feasible set in x is denoted by X_R^J .

Let α be the vector of strictly positive scaling parameters, and $\mathcal{A} = \{\alpha \mid \alpha > 0\}$. It is clear that constraint

$$\inf_{f(\xi) \in \mathcal{P}} \Pr \left[\max_{i=1, \dots, m} \{\alpha_i (y_i^0(x) + y_i(x)^T \xi)\} \leq 0 \right] \geq 1 - \varepsilon \quad (\text{C.91})$$

imposes the same feasible region in variable x as (C.90). Nonetheless, it turns out that parameter α_i can be co-optimized to improve the quality of the convex approximation for X_R^J . (C.91) is a single robust chance constraint, and can be conservatively approximated by a worst-case CVaR constraint

$$\sup_{f(\xi) \in \mathcal{P}} \text{CVaR} \left[\max_{i=1, \dots, m} \{ \alpha_i (y_i^0(x) + y_i(x)^T \xi) \}, \varepsilon \right] \leq 0 \quad (\text{C.92})$$

It defines a feasible region in variable x with auxiliary parameter $\alpha \in \mathcal{A}$, which is denoted by $X_R^J(\alpha)$. Clearly, $X_R^J(\alpha) \subseteq X_R^J$, $\forall \alpha \in \mathcal{A}$. Unlike (C.88), condition (C.92) is α -dependent.

By observing the fact that

$$\begin{aligned} [\xi^T \ 1] M [\xi^T \ 1] &\geq \max_{i=1, \dots, m} \{ \alpha_i (y_i^0(x) + y_i(x)^T \xi) \} - \beta, \quad \forall \xi \in \mathbb{R}^k \\ \iff [\xi^T \ 1] M [\xi^T \ 1] &\geq \alpha_i (y_i^0(x) + y_i(x)^T \xi) - \beta, \quad \forall \xi \in \mathbb{R}^k, \quad i = 1, \dots, m \\ \iff M - \begin{bmatrix} 0 & \frac{\alpha_i y_i(x)}{2} \\ \frac{\alpha_i y_i(x)^T}{2} & \alpha_i y_i^0(x) - \beta \end{bmatrix} &\succeq 0, \quad i = 1, \dots, m \end{aligned}$$

and employing the optimization formulation of the worst-case expectation problem, the worst-case CVaR in (C.92) can be calculated by

$$\begin{aligned} J(x, \alpha) &= \sup_{f(\xi) \in \mathcal{P}} \text{CVaR} \left[\max_{i=1, \dots, m} \{ \alpha_i (y_i^0(x) + y_i(x)^T \xi) \}, \varepsilon \right] \\ &= \inf_{\beta \in \mathbb{R}} \left\{ \beta + \frac{1}{\varepsilon} \sup_{f(\xi) \in \mathcal{P}} \mathbb{E}_{f(\xi)} \left(\left[\max_{i=1, \dots, m} \{ \alpha_i (y_i^0(x) + y_i(x)^T \xi) \} - \beta \right]^+ \right) \right\} \\ &= \inf_{\beta, M} \left\{ \beta + \frac{1}{\varepsilon} \text{tr}(\Omega^T M) \mid \text{s.t. } M \succeq 0, \quad M \succeq \begin{bmatrix} 0 & \frac{\alpha_i y_i(x)}{2} \\ \frac{\alpha_i y_i(x)^T}{2} & y_i^0(x) - \beta \end{bmatrix}, \forall i \right\} \end{aligned} \quad (\text{C.93})$$

In conclusion, for any fixed $\alpha \in \mathcal{A}$, the worst-case CVaR constraint (C.92) can be written as a convex set in variables x , β , and M as follows

$$X_R^J(\alpha) = \left\{ x \mid \begin{array}{l} \exists \beta \in \mathbb{R}, \quad M \succeq 0 \text{ such that} \\ \beta + \frac{1}{\varepsilon} \text{tr}(\Omega^T M) \leq 0 \\ M \succeq \begin{bmatrix} 0 & \frac{\alpha_i y_i(x)}{2} \\ \frac{\alpha_i y_i(x)^T}{2} & y_i^0(x) - \beta \end{bmatrix}, \forall i \end{array} \right\} \quad (\text{C.94})$$

Moreover, it is revealed in [38] that the union $\bigcup_{\alpha \in \mathcal{A}} X_R^J(\alpha)$ gives an exact description of X_R^J , which indicates that the original robust chance constrained program

$$\min_x \{ c^T x \mid \text{s.t. } x \in X \cap X_R^J \} \quad (\text{C.95})$$

and the worst-case CVaR formulation

$$\min_{x, \alpha} \{c^T x \mid \text{s.t. } x \in X \cap X_R^J(\alpha), \alpha \in \mathcal{A}\}$$

or equivalently

$$\min_{x, \alpha} \{c^T x \mid \text{s.t. } x \in X, \alpha \in \mathcal{A}, J(x, \alpha) \leq 0\} \quad (\text{C.96})$$

have the same optimal value. The constraints of (C.96) contain bilinear matrix inequalities, which means that if either x or α is fixed, $J(x, \alpha) \leq 0$ in (C.96) can come down to LMIs, however, when both x and α are variables, the constraint is non-convex, making problem (C.96) difficult to solve. In view of the biconvex feature [32], a sequential convex optimization procedure is presented to find an approximated solution.

Algorithm C.4

- 1: Choose a convergence tolerance $\varepsilon > 0$; Let the iteration counter $k = 1$, $x^0 \in X \cap X_R^J(\alpha)$ be a feasible solution for some α and $f^0 = c^T x^0$;
- 2: Solve the following subproblem with input x^{k-1}

$$\min_{\alpha} \{J(x, \alpha) \mid \text{s.t. } \alpha \geq \delta \mathbf{1}\} \quad (\text{C.97})$$

where $\mathbf{1}$ denotes the all-one vector with a compatible dimension, and $\delta > 0$ is a small constant; the worst-case CVaR functional is defined in (C.93). The optimal solution is α^k ;

- 3: Solve the following master problem with input α^k

$$\min_x \{c^T x \mid \text{s.t. } x \in X, J(x, \alpha^k) \leq 0\} \quad (\text{C.98})$$

The optimal solution is x^k and the optimal value is f^k ;

- 4: If $|f^k - f^{k-1}|/|f^{k-1}| \leq \varepsilon$, terminate and report the optimal solution x^k ; otherwise, update $k \leftarrow k + 1$, and go to step 2.
-

The main idea of this algorithm is to identify the best feasible region $X_R^J(\alpha)$ through successively solving the subproblem (C.97), and therefore improving the objective value. The performance of Algorithm C.4 is intuitively explained below.

Because parameter α is optimized in the subproblem (C.97) given the value x^k , there must be $J(x^k, \alpha^{k+1}) \leq J(x^k, \alpha^k) \leq 0$, $\forall k$, demonstrating that x^k is a feasible solution of the master problem (C.98) in iteration $k + 1$; therefore, the optimal values of (C.98) in two consecutive iterations satisfy $c^T x^{k+1} \leq c^T x^k$, as the objective evaluated at the optimal solution x^{k+1} in iteration $k + 1$ deserves a value no greater than that is incurred at any feasible solution. In this regard, the optimal value sequence f^k , $k = 1, 2, \dots$ is monotonically decreasing. If X is bounded, the optimal solution sequence

x^k is also bounded, and the optimal value converges. Algorithm C.4 does not necessarily find the global optimum of problem (C.96). Nevertheless, it is desired by practical problems due to its robustness since it involves only convex optimization.

In many practical applications, the uncertain data ξ is known to be within a strict subset of \mathbb{R}^k , which is called the support set. We briefly outline how to incorporate the support set in the distributionally robust chance constraints. We assume the support set Ξ is the intersection of a finite number of ellipsoids, i.e.

$$\Xi = \{\xi \in \mathbb{R}^k \mid \xi^T W_i \xi \leq 1, \ i = 1, \dots, l\} \quad (\text{C.99})$$

where $W_i \in \mathbb{S}_+^k$, $i = 1, \dots, l$, and we have $\Pr(\xi \in \Xi) = 1$. Let \mathcal{P}_Ξ be the set of all candidate probability distributions supported on Ξ which have identical first- and second-order moments.

Consider the worst-case expectation problem (C.79). If we replace \mathcal{P} with \mathcal{P}_Ξ , the constraints of the dual problem (C.80) become

$$[\xi^T \ 1] M [\xi^T \ 1]^T \geq 0, \quad \forall \xi \in \Xi \quad (\text{C.100})$$

$$[\xi^T \ 1] M [\xi^T \ 1]^T \geq g(\xi), \quad \forall \xi \in \Xi \quad (\text{C.101})$$

According to (C.99), $1 - \xi^T W_i \xi$ must be non-negative if and only if $\xi \in \Xi$, and hence a sufficient condition for (C.100) is the existence of constants $\tau_i \geq 0$, $i = 1, \dots, l$, such that

$$[\xi^T \ 1] M [\xi^T \ 1]^T - \sum_{i=1}^l \tau_i (1 - \xi^T W_i \xi) \geq 0 \quad (\text{C.102})$$

Under this condition, as long as $\xi \in \Xi$, we have

$$[\xi^T \ 1] M [\xi^T \ 1]^T \geq \sum_{i=1}^l \tau_i (1 - \xi^T W_i \xi) \geq 0$$

Arrange (C.102) as a matrix quadratic form

$$[\xi^T \ 1] \left(M - \sum_{i=1}^l \tau_i \begin{bmatrix} -W_i & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix} \right) \begin{bmatrix} \xi \\ 1 \end{bmatrix} \geq 0, \quad \forall \xi \in \mathbb{R}^k$$

As a result, (C.100) can be reduced to an LMI in variables M and τ

$$M - \sum_{i=1}^l \tau_i \begin{bmatrix} -W_i & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix} \succeq 0 \quad (\text{C.103})$$

For similar reasons, by letting $g(\xi) = y^0(x) + y(x)^T \xi - \beta$, (C.101) can be conservatively approximated by the following LMI

$$M - \sum_{i=1}^l \tau_i \begin{bmatrix} -W_i & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix} \succeq \begin{bmatrix} 0 & \frac{1}{2}y(x) \\ \frac{1}{2}y(x)^T & y^0(x) - \beta \end{bmatrix} \quad (\text{C.104})$$

In fact, (C.103) and (C.104) are special cases of S-Lemma. Based upon these outcomes, most formulations in this section can be extended to consider the bounded support set Ξ in the form of (C.99). For polyhedral and some special classes of convex support sets, one may utilize the nonlinear Farkas lemma (Lemma 2.2 in [38]) to derive tractable reformulations.

C.3.2 Adjustable Distributionally Robust Optimization

As explained in Appendix C.1, the traditional static RO encounters difficulties in dealing with equality constraints. This plight remains in the DRO approach following a static setting. Consider $x + \xi = 1$ where $\xi \in [0, 0.1]$ is uncertain, while its mean and variance are known. For any given x^* , the worst-case probability $\inf_{f(\xi) \in \mathcal{P}} \Pr[x^* + \xi = 1] = 0$, because one can always find a feasible probability distribution function $f(\xi)$ that satisfies the first- and second-order moment constraints, whereas $f(1 - x^*) = 0$.

To vanquish this difficulty, it is necessary to incorporate wait-and-see decisions. A simple remedy is to impose an affine recourse policy without involving optimization in the second stage, giving rise to an affine-adjustable RO with distributional uncertainty and linear decision rule, which can be solved by the method in Appendix C.3.1.

This section aims to investigate the following adjustable DRO with completely flexible wait-and-see decisions

$$\min_{x \in X} \left\{ c^T x + \sup_{f(w) \in \mathcal{P}} \mathbb{E}_{f(w)} Q(x, w) \right\} \quad (\text{C.105})$$

where x is the first-stage (here-and-now) decision, and X is its feasible set; the uncertain parameter is denoted by w ; the probability distribution $f(w)$ belongs to the Chebyshev ambiguity set (whose first- and second-order moments are known)

$$\mathcal{P} = \left\{ f(w) \left| \begin{array}{l} f(w) \geq 0, \forall w \in W \\ \int_{w \in W} f(w) dw = 1 \\ \int_{w \in W} w f(w) dw = \mu \\ \int_{w \in W} w w^T f(w) dw = \Theta \end{array} \right. \right\} \quad (\text{C.106})$$

supported on $W = \{w \mid (w - \mu)^T Q(w - \mu) \leq \Gamma\}$, where matrix $\Theta = \Sigma + \mu\mu^T$ represents the second-order moment; μ is the mean value and Σ is the covariance matrix. The expectation in (C.105) is taken over the worst-case $f(w)$ in \mathcal{P} , and the second-stage problem under fixed x and w is an LP

$$Q(x, w) = \min_{y \in Y(x, w)} d^T y \quad (\text{C.107})$$

$Q(x, w)$ is its optimal value function under fixed x and w . The feasible set of the second-stage problem is

$$Y(x, w) = \{y \mid By \leq b - Ax - Cw\}$$

Matrices A, B, C and vectors b, c, d are constant coefficients in the model. We assume that the second-stage problem is always feasible, i.e., $\forall x \in X, \forall w \in W : Y(x, w) \neq \emptyset$ and is bounded, and thus $Q(x, w)$ has a finite optimal value. This can be implemented by introducing wait-and-see type slack variables and adding penalties in the objective of (C.107).

The difference between problems (C.42) and (C.105) stems from the descriptions of uncertainty and the criteria in the objective function: more information of the dispersion effect, such as the covariance matrix, is taken into account in the latter one, and the objective function in (C.105) is an expectation reflecting the statistical behavior of the second-stage cost, rather than the one in (C.42) which is associated with only a single worst-case scenario, and leaves the performances in all other scenarios un-optimized. Because the probability distribution is uncertain, it is prudent to investigate the worst-case outcome in which the expected cost of the second stage is maximized. This formulation is advantageous in several ways: first, the requirement on the exact probability distribution is not necessary, and the optimal solution is insensitive to the family of distributions with common mean and covariance; second, the dispersion of the uncertainty is also taken into account, which helps reduce model conservatism: since the variance is fixed, a scenario that leaves far away from the forecast would have a low probability; finally, it is often important to tackle the tail effect, which indicates that the occurrence of a rare event may induce heavy losses in spite of its low probability. Such phenomenon is naturally taken into account in (C.105). In what follows, we outline the method proposed in [48] to solve the adjustable DRO problem (C.105). A slight modification is that an ellipsoid support set is considered.

1. The worst-case expectation problem

We consider the following worst-case expectation problem with a fixed x

$$\sup_{f(w) \in \mathcal{P}} \mathbb{E}_{f(w)} Q(x, w) \quad (\text{C.108})$$

According to the discussions for problem (C.79), the dual problem of (C.108) is

$$\begin{aligned}
& \min_{H, h, h_0} \quad \text{tr}(H^T \Theta) + \mu^T h + h_0 \\
& \text{s.t.} \quad w^T H w + h^T w + h_0 \geq Q(x, w), \quad \forall w \in W
\end{aligned} \tag{C.109}$$

where H , h , h_0 are dual variables. Nevertheless, the optimal value function $Q(x, w)$ is not given in a closed form. From the LP duality theory

$$Q(x, w) = \max_{u \in U} u^T (b - Ax - Cw)$$

where u is the dual variable of LP (C.107), and its feasible set is given by

$$U = \{u \mid B^T u = d, u \leq 0\}$$

Because we have assumed that $Q(x, w)$ is bounded, the optimal solution of the dual problem can be found at one of the extreme points of U , i.e.,

$$\exists u^* \in \text{vert}(U) : Q(x, w) = (b - Ax - Cw)^T u^* \tag{C.110}$$

where $\text{vert}(U) = \{u^1, u^2, \dots, u^{N_E}\}$ stands for the vertices of polyhedron U , and $N_E = |\text{vert}(U)|$ is the cardinality of $\text{vert}(U)$. In view of this, the constraint of (C.109) can be expressed as

$$w^T H w + h^T w + h_0 \geq (b - Ax - Cw)^T u^i, \quad \forall w \in W, i = 1, \dots, N_E$$

Recall the definition of W ; a certification for above condition is

$$\begin{aligned}
& w^T H w + h^T w + h_0 - (b - Ax - Cw)^T u^i \\
& \geq \lambda[\Gamma - (w - \mu)^T Q(w - \mu)] \geq 0, \quad \forall w \in \mathbb{R}^k, i = 1, \dots, N_E
\end{aligned}$$

which has the following compact matrix form

$$\begin{bmatrix} w \\ 1 \end{bmatrix}^T M^i \begin{bmatrix} w \\ 1 \end{bmatrix} \geq 0, \quad \forall w \in \mathbb{R}^k, i = 1, \dots, N_E \tag{C.111}$$

where

$$M^i = \begin{bmatrix} H + \lambda Q & \frac{h - C^T u^i}{2} - \lambda Q \mu \\ \frac{h^T - (u^i)^T C}{2} - \lambda \mu^T Q & h_0 - (b - Ax)^T u^i - \lambda(\Gamma - \mu^T Q \mu) \end{bmatrix} \tag{C.112}$$

and (C.111) simply reduces to $M^i \succeq 0$, $i = 1, \dots, N_E$.

Finally, problem (C.109) comes down to the following SDP

$$\begin{aligned}
& \min_{H, h, h_0, \lambda} \quad \text{tr}(H^T \Theta) + \mu^T h + h_0 \\
& \text{s.t.} \quad M^i(H, h, h_0, \lambda) \succeq 0, \quad i = 1, \dots, N_E \\
& \quad \lambda \in \mathbb{R}^+
\end{aligned} \tag{C.113}$$

where $M^i(H, h, h_0, \lambda)$ is defined in (C.112). Above results can be readily extended if the support set is the intersection of ellipsoids.

2. Adaptive constraint generation algorithm

Due to the positive semi-definiteness of the covariance matrix Σ , the duality gap between problems (C.108) and (C.109) is zero [48], and hence we can replace the worst-case expectation in (C.105) with its dual form, yielding

$$\begin{aligned} \min \quad & c^T x + \text{tr}(H^T \Theta) + \mu^T h + h_0 \\ \text{s.t.} \quad & M^i(H, h, h_0, \lambda) \succeq 0, \quad i = 1, \dots, N_E \\ & x \in X, \quad \lambda \in \mathbb{R}^+ \end{aligned} \quad (\text{C.114})$$

Problem (C.114) is an SDP. However, the number of vertices in set U ($|\text{vert}(U)|$) may increase exponentially in the dimension of U . It is non-trivial to enumerate all of them. However, because of weak duality, only the one which is optimal in the dual problem provides an active constraint, as shown in (C.110), and the rest are redundant inequalities. To identify the critical vertex in (C.110), we solve problem (C.114) in iterations: in the master problem, a subset of $\text{vert}(U)$ is used to formulate a relaxation, then check whether the following constraint

$$w^T H w + h^T w + h_0 \geq (b - Ax - Cw)^T u, \quad \forall w \in W, \quad \forall u \in U \quad (\text{C.115})$$

is fulfilled. If yes, the relaxation is exact and the optimal solution is found; otherwise, find a new vertex of U at which constraint (C.115) is violated, and then add a cut to the master problem so as to tighten the relaxation, till constraint (C.115) is satisfied. The flowchart is summarized in Algorithm C.5.

Algorithm C.5 terminates in a finite number of iterations which is bounded by $|\text{vert}(U)|$. Actually, it will converge within a few iterations, because the sub-problem (C.117) in step 3 always identifies the most critical vertex in $\text{vert}(U)$. It is worth mentioning that the subproblem (C.117) is a non-convex program. Despite that it can be solved by general NLP solvers, we suggest three approaches with different computational complexity and optimality guarantees.

1. If the support set $W = \mathbb{R}^k$, it can be verified that matrix M_i becomes

$$\begin{bmatrix} H & \frac{h + C^T u^i}{2} \\ \frac{(h + C^T u^i)^T}{2} & h_0 - (b - Ax)^T u^i \end{bmatrix} \succeq 0$$

Then there must be $H \succeq 0$, and non-convexity appears in the bilinear term $u^T C w$. In such circumstance, problem (C.117) can be solved via a mountain climbing method similar to Algorithm C.1 (but here the mountain is actually a pit because the objective is to be minimized).

Algorithm C.5

-
- 1: Choose a convergence tolerance $\epsilon > 0$ and an initial vertex set $V_E \subseteq \text{vert}(U)$.
 - 2: Solve the following master problem

$$\begin{aligned}
\min \quad & c^T x + \text{tr}(H^T \Theta) + \mu^T h + h_0 \\
\text{s.t.} \quad & M^i(H, h, h_0, \lambda) \succeq 0, \quad \forall u^i \in V_E \\
& x \in X, \quad \lambda \in \mathbb{R}^+
\end{aligned} \tag{C.116}$$

The optimal value is R^* , and the optimal solution is (x^*, H, h, h_0) .

- 3: Solve the following sub-problem with obtained (x^*, H, h, h_0)

$$\begin{aligned}
\min_{w, u} \quad & w^T H w + h^T w + h_0 - (b - A x^* - C w)^T u \\
\text{s.t.} \quad & w \in W, \quad u \in U
\end{aligned} \tag{C.117}$$

The optimal value is r^* , and the optimal solution is u^* and w^* .

- 4: If $r^* \geq -\epsilon$, terminate and report the optimal solution x^* and the optimal value R^* ; otherwise, $V_E = V_E \cup u^*$, add an LMI cut $M(H, h, h_0, \lambda) \succeq 0$ associated with the current u^* to the master problem (C.116), and go to step 2.
-

2. In the case that W is an ellipsoid, above iterative approach is still applicable; however, the w -subproblem in which w is to be optimized may become non-convex because H may be indefinite. Since u is fixed in the w -subproblem, non-convex term $w^T H w$ can be decomposed as the difference of two convex functions as $w^T (H + \alpha I) w - \alpha w^T w$, where α is a constant such that $H + \alpha I$ is positive-definite, and the w -subproblem can be solved by the convex-concave procedure elaborated in [47], or any existing NLP solver.

3. As a non-convex QP, problem (C.117) can be globally solved by the MILP method presented in Appendix A.4. This method could be time consuming with the growth in problem sizes.

C.4 Data-driven Robust Stochastic Program

Most classical SO methods assume that the probability distribution of uncertain factors is exactly known, which is an input of the problem. However, such information heavily relies on historical data, and may not be available at hand or accurate enough. Using an inaccurate distribution in a classical SO model could lead to biased results. To cope with ambiguous probability distributions, a natural way is to consider a set of possible candidates derived from available data, instead of a single distribution, just as the moment-inspired

ambiguity set used in DRO. In this section, we investigate some useful SO models with distributional uncertainty described by divergence ambiguity sets, which is referred to as robust SO. When the distribution is discrete, the distributional uncertainty is interpreted by the perturbation of probability value associated with each scenario; when the distribution is continuous, the distance of two density functions should be specified first. In this section, we consider Φ -divergence and Wasserstein metric based ambiguity sets.

C.4.1 Robust Chance Constrained Stochastic Program

We introduce robust chance-constrained stochastic programs with distributional robustness. The ambiguous PDF is modeled based on ϕ -divergence, and the optimal solution provides constraint feasibility guarantee with desired probability even in the worst-case distribution. In short, the underlying problem possesses the following features:

- 1) The PDF is continuous and the constraint violation probability is a functional.
 - 2) Uncertain parameters do not explicitly appear in the objective function.
- Main results of this section come from [49].

1. Problem formulation

In a traditional chance-constrained stochastic linear program, the decision maker seeks a cost-minimum solution at which some certain constraints can be met with a given probability, yielding:

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & \Pr[C(x, \xi)] \geq 1 - \alpha \\ & x \in X \end{aligned} \tag{C.118}$$

where x is the vector of decision variables; ξ is the vector of uncertain parameters, and the exact (joint) probability distribution is apparent to the decision maker; vector c represents the cost coefficients; X is a polyhedron that is independent of ξ ; α is the risk level or the maximum allowed probability of constraint violation; $C(x, \xi)$ collects all uncertainty dependent constraints, whose general form is given by

$$C(x, \xi) = \{\xi \mid \exists y : A(\xi)x + B(\xi)y \leq b(\xi)\} \tag{C.119}$$

where A , B , b are constant coefficient matrices that may contain uncertain parameters; y is a recourse action that can be made after ξ is known. In the presence of y , we call (C.118) a two-stage problem; otherwise, it is a single-stage problem if y is null. We don't consider the cost of recourse actions in the objective function in its current form. In case of need, we can add the

second-stage cost $d^T y(\xi)$ in the objective function, and ξ is a specific scenario which $y(\xi)$ corresponds to; for instance, robust optimization may consider a max-min cost scenario or a max-min regret scenario; traditional SO often tackles the expected second-stage cost $\mathbb{E}[d^T y(\xi)]$. We leave it to the end of this section to discuss how to deal with the second-stage cost in the form of worst-case expectation like (C.108), and show that the problem can be convexified under some technical assumptions.

In the chance constraint, for a given x , the probability of constraint satisfaction can be evaluated for a particular probability distribution of ξ . Traditional studies on chance-constrained programs often assume that the distribution of ξ is perfectly known. However, this assumption can be very strong because it requires a lot of historical data. Moreover, the optimal solution may be sensitive to the true distribution and thus highly suboptimal in practice. To overcome these difficulties, a prudent method is to consider a set of probability distributions belonging to a pre-specified ambiguity set D , and require that the chance constraint should be satisfied under all possible distributions in D , resulting in the following robust chance-constrained programming problem:

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & \inf_{f(\xi) \in D} \Pr[C(x, \xi)] \geq 1 - \alpha \\ & x \in X \end{aligned} \tag{C.120}$$

where $f(\xi)$ is the probability density function of random variable ξ .

The ambiguity set D in (C.120) which includes distributional information can be constructed in a data-driven fashion, such as the moment based ones used in Appendix C.3. Please see [57] for more information on establishing D based on moment data and other structural properties, such as symmetry and unimodality. The tractability of (C.120) largely depends on the form of D . For example: if D is built on the mean value and covariance matrix (which is called a Chebyshev ambiguity set), a single robust chance constraint can be reformulated as an LMI and a set of joint robust chance constraints can be approximated by BMIs [38]; probability of constraint violation under more general moment based ambiguity sets can be evacuated by solving conic optimization problems [57].

A shortcoming of moment description is that it does not provide a direct measure on the distance between the candidate PDFs in D and a reference distribution. Two PDFs with the same moments may differ a lot in other aspects. Furthermore, the worst-case distribution corresponding to a Chebyshev ambiguity set always puts more weights away from the mean value, subject to the variance. As such, the long-tail effect is a source of conservatism. In this section, we consider the confidence set built around a reference distribution. The motivation is: the decision maker may have some knowledge on what distribution the uncertainty follows, although such a distribution could be inexact, and the true density function would not deviate far away from it.

To describe distributional ambiguity in term of a PDF, the first problem is how to characterize the distance between two functions. One common measure on the distance between density functions is the ϕ -divergence, which is defined as [58]

$$D_\phi(f\|f_0) = \int_{\Omega} \phi\left(\frac{f(\xi)}{f_0(\xi)}\right) f_0(\xi) d\xi \quad (\text{C.121})$$

where f and f_0 stand for the particular density function and the estimated one (or the reference distribution), respectively; function ϕ satisfies:

$$(C1) \quad \phi(1) = 0$$

$$(C2) \quad 0\phi(x/0) = \begin{cases} x \lim_{p \rightarrow +\infty} \phi(p)/p & \text{if } x > 0 \\ 0 & \text{if } x = 0 \end{cases}$$

$$(C3) \quad \phi(x) = +\infty \text{ for } x < 0$$

$$(C4) \quad \phi(x) \text{ is a convex function on } \mathbb{R}^+$$

It is proposed in [58] that the ambiguity set can be built as:

$$D = \{P : D_\phi(f\|f_0) \leq d, f = dP/d\xi\} \quad (\text{C.122})$$

where the tolerance d can be adjusted by the decision maker according to their attitudes towards risks. The ambiguity set in (C.122) can be denoted as D_ϕ , without causing confusion with the definition of ϕ -divergence $D_\phi(f\|f_0)$. Compared to the moment-based ambiguity sets, especially the Chebyshev ambiguity set, where only the first- and second-order moments are involved, the density based description captures the overall profile of the ambiguous distribution, so may hopefully provide less conservative solutions. However, it hardly guarantees consistent moments. Which one is better depends on data availability: if we are more confident on the reference distribution, (C.122) may be better; otherwise, if we only have limited statistic information such as mean and variance, then the moment-based ones are more straightforward.

Many commonly seen divergence measures are special cases of ϕ -divergence, coinciding with a particular choice of function ϕ . Some examples are given in Table C.2 [59]. In what follows, we will use the KL-divergence. According to its corresponding function ϕ , the KL-divergence is given by

$$D_\phi(f\|f_0) = \int_{\Omega} \log\left(\frac{f(\xi)}{f_0(\xi)}\right) f(\xi) d\xi \quad (\text{C.123})$$

Before presenting the main results in [49], the definition of conjugate duality is given. For a univariate function $g : \mathbb{R} \rightarrow \mathbb{R} \cup \{+\infty\}$, its conjugate function $g^* : \mathbb{R} \rightarrow \mathbb{R} \cup \{+\infty\}$ is defined as

$$g^*(t) = \sup_{x \in \mathbb{R}} \{tx - g(x)\}$$

Table C.2 Instances of ϕ -divergences

Divergence	function $\phi(x)$
KL-divergence	$x \log x - x + 1$
reverse KL-divergence	$-\log x$
Hellinger distance	$(\sqrt{x} - 1)^2$
Variation distance	$ x - 1 $
J-divergence	$(x - 1) \log x$
χ^2 divergence	$(x - 1)^2$
α -divergence	$\begin{cases} \frac{4}{1 - \alpha^2} (1 - x^{(1+\alpha)/2}) & \text{If } \alpha \neq \pm 1 \\ x \ln x & \text{If } \alpha = 1 \\ -\ln x & \text{If } \alpha = -1 \end{cases}$

For a valid function ϕ for ϕ -divergence satisfying (C1)-(C4), its conjugate function ϕ^* is convex, nondecreasing, and the following condition holds [49]

$$\phi^*(x) \geq x \quad (\text{C.124})$$

Besides, if ϕ^* is a finite constant on a closed interval $[a, b]$, then it is a finite constant on the interval $(-\infty, b]$.

2. Equivalent formulation

It is revealed in [49] that when the confidence set D is constructed based on ϕ -divergence, robust chance constrained program (C.120) can be easily transformed into a traditional chance-constrained program (C.118) at the reference distribution by calibrating the confidence tolerance α .

Theorem C.1. [49] *Let \mathbb{P}_0 be the cumulative distribution function generated by density function f_0 , then the robust chance constraint*

$$\inf_{\mathbb{P}(\xi) \in \{D_\phi(f \| f_0) \leq d\}} \Pr[C(x, \xi)] \geq 1 - \alpha \quad (\text{C.125})$$

constructed based on ϕ -divergence is equivalent to a traditional chance constraint

$$\Pr_0[C(x, \xi)] \geq 1 - \alpha'_+ \quad (\text{C.126})$$

where \Pr_0 means that the probability is evaluated at the reference distribution \mathbb{P}_0 , $\alpha'_+ = \max\{\alpha', 0\}$, and α' can be computed by

$$\alpha' = 1 - \inf_{z \in Z} \left\{ \frac{\phi^*(z_0 + z) - z_0 - \alpha z + d}{\phi^*(z_0 + z) - \phi^*(z_0)} \right\}$$

where

$$Z = \left\{ z \left| \begin{array}{l} z > 0, \quad z_0 + \pi z \leq l_\phi \\ \underline{m}(\phi^*) \leq z + z_0 \leq \overline{m}(\phi^*) \end{array} \right. \right\}$$

In above formula, constants $l_\phi = \lim_{x \rightarrow +\infty} \phi(x)/x$, $\overline{m}(\phi^*) = \inf\{m : \phi^*(m) = +\infty\}$, $\underline{m}(\phi^*) = \sup\{m : \phi^* \text{ is a finite constant on } (-\infty, m]\}$, Table C.3 summarizes the values of these parameters for typical ϕ -divergence measures, and

$$\pi = \begin{cases} -\infty & \text{if } \text{Leb}\{[f_0 = 0]\} = 0 \\ 0 & \text{if } \text{Leb}\{[f_0 = 0]\} > 0 \text{ and } \text{Leb}\{[f_0 = 0] \setminus C(x, \xi)\} = 0 \\ 1 & \text{otherwise} \end{cases}$$

where $\text{Leb}\{\cdot\}$ is the Lebesgue measure on $\mathbb{R}^{\text{Dim}(\xi)}$.

Table C.3 Values of l_ϕ , $\underline{m}(\phi^*)$, and $\overline{m}(\phi^*)$ for ϕ -divergences

ϕ -Divergence	l_ϕ	$\underline{m}(\phi^*)$	$\overline{m}(\phi^*)$
KL-divergence	$+\infty$	$-\infty$	$+\infty$
Hellinger distance	1	$-\infty$	1
Variation distance	1	-1	1
J-divergence	$+\infty$	$-\infty$	$+\infty$
χ^2 divergence	$+\infty$	-2	$+\infty$

The values of α' for the Variation distance and the χ^2 divergence have analytical expressions; for the KL divergence, α' can be computed from one-dimensional line search. Results are shown in Table C.4.

Table C.4 Values of α' for some ϕ -divergences

ϕ -Divergence	α'
χ^2 divergence	$\alpha' = \alpha - \frac{\sqrt{d^2 + 4d(\alpha - \alpha^2)} - (1 - 2\alpha)d}{2d + 2}$
Variation distance	$\alpha' = \alpha - \frac{1}{2}d$
KL-divergence	$\alpha' = 1 - \inf_{x \in (0,1)} \left\{ \frac{e^{-d}x^{1-\alpha} - 1}{x - 1} \right\}$

For the KL divergence, calculating α' entails solving $\inf_{x \in (0,1)} h(x)$ where

$$h(x) = \frac{e^{-d}x^{1-\alpha} - 1}{x - 1}$$

Its first-order derivative is given by

$$h'(x) = \frac{1 - \alpha e^{-d} x^{1-\alpha} - (1-\alpha)e^{-d} x^{-\alpha}}{(x-1)^2}, \quad \forall x \in (0, 1)$$

To claim the convexity of $h(x)$, we need to show that $h'(x)$ is an increasing function in $x \in (0, 1)$. To this end, first notice that the denominator $(x-1)^2$ is a decreasing function in x on the open interval $(0, 1)$; then we can show the numerator is an increasing function in x , because its first-order derivative gives

$$(1 - \alpha e^{-d} x^{1-\alpha} - (1-\alpha)e^{-d} x^{-\alpha})'_x = \alpha(1-\alpha)e^{-d}(x^{-\alpha-1} - x^{-\alpha}) > 0, \quad \forall x \in (0, 1)$$

Hence $h'(x)$ is monotonically increasing, and $h(x)$ is a convex function in x . Moreover, because $h'(x)$ is continuous in $(0, 1)$, and $\lim_{x \rightarrow 0^+} h'(x) = -\infty$, $\lim_{x \rightarrow 1^-} h'(x) = +\infty$, there must be some $x^* \in [\delta, 1-\delta]$ such that $h'(x^*) = 0$, i.e., the infimum of $h(x)$ is attainable. The minimum of $h(x)$ can be calculated by solving a nonlinear equation $h'(x) = 0$ via Newton's method, or a derivative-free line search, such as the golden section search algorithm. Either scheme is computationally inexpensive.

Finally, we discuss the connection between the modified tolerance α' and its original value α . Because a set of distributions are considered in (C.125), the threshold in (C.126) should be greater than the original one, i.e., $1 - \alpha' \geq 1 - \alpha$ must hold. To see this, recall inequality (C.124) of conjugate function, we have

$$\alpha\phi^*(z_0 + z) + (1 - \alpha)\phi(z_0) \geq \alpha(z_0 + z) + (1 - \alpha)z_0$$

The right-hand side gives $\alpha z + z_0$; in the ambiguity set (C.123), d is strictly positive, therefore

$$\alpha\phi^*(z_0 + z) + (1 - \alpha)\phi(z_0) \geq \alpha z + z_0 - d$$

which gives

$$\phi^*(z_0 + z) - z_0 - \alpha z + d \geq (1 - \alpha)(\phi(z_0 + z) - \phi(z_0))$$

Recall the expression of α' in Theorem C.1, we arrive at

$$1 - \alpha' = \frac{\phi^*(z_0 + z) - z_0 - \alpha z + d}{\phi(z_0 + z) - \phi(z_0)} \geq 1 - \alpha$$

which is the desired conclusion.

Theorem C.1 concludes that the complexity of handling a robust chance constraint is almost the same as that of tackling a traditional chance constraint associated with the reference distribution \mathbb{P}_0 , except for the efforts on computing α' . If \mathbb{P}_0 belongs to the family of log-concave distributions,

then the chance constraint is convex. As a special case, if \mathbb{P}_0 is the Gaussian distribution or a uniform distribution on ellipsoidal support, a single chance constraint can boil down to a second-order cone [60]. For more general cases, the chance constraint is non-convex in x . In such circumstance, we will use risk based reformulation and the sampling average approximation (SAA) approach.

3. Risk and SAA based reformulation

Owing to the different descriptions on dispersion ambiguity and presence of the wait-and-see decision y , unlike DRO problem (C.76) with static robust chance constraint (C.78) which can be transformed into an SDP, constraint (C.125) is treated in a different way, as demonstrated in Theorem C.1: it comes down to a traditional chance constraint (C.126) while the dispersion ambiguity is taken into account by a modification in the confidence level. The remaining task is to express (C.126) as a solver-compatible form.

1) Loss function

For given x and ξ , constraints in $C(x, \xi)$ cannot be met if no y satisfying $A(\xi)x + B(\xi)y \leq b(\xi)$ exists. To quantify the constraint violation under scenario ξ and first-stage decision x , define the following loss function $L(x, \xi)$

$$\begin{aligned} L(x, \xi) = \min_{y, \sigma} \quad & \sigma \\ \text{s.t.} \quad & A(\xi)x + B(\xi)y \leq b(\xi) + \sigma \mathbf{1} \end{aligned} \quad (\text{C.127})$$

where $\mathbf{1}$ is an all-one vector with compatible dimension. If $L(x, \xi) \geq 0$, the minimum of slackness σ under the joint efforts of the recourse action y is defined as the loss; otherwise, demands are satisfiable after the uncertain parameter is known. As we assume $C(x, \xi)$ is a bounded polytope, problem (C.127) is always feasible and bounded below. Therefore, the loss function $L(x, \xi)$ is well-defined, and the chance constraint (C.126) can be written as

$$\Pr_0[L(x, \xi) \leq 0] \geq 1 - \alpha'_+ \quad (\text{C.128})$$

In this way, the joint chance constraints are consolidated into a single one, just like what has been done in (C.90) and (C.91).

2) VaR based reformulation: An MILP

For a given probability tolerance β and a first-stage decision x , the β -VaR for loss function $L(x, \xi)$ under the reference distribution PDF \mathbb{P}_0 is defined as

$$\beta\text{-VaR}(x) = \min \left\{ a \in \mathbb{R} \left| \int_{L(x, \xi) \leq a} f_0(\xi) d\xi \geq \beta \right. \right\} \quad (\text{C.129})$$

which interprets the threshold a such that the loss is no greater than a will hold with a probability no less than β . According to (C.129), an equivalent

expression of chance constraint (C.128) is

$$(1 - \alpha'_+)\text{-VaR}(x) \leq 0 \quad (\text{C.130})$$

So that probability evaluation is obviated. Furthermore, if SAA is used, (C.128) and (C.130) indicate that the scenarios which will lead to $L(x, \xi) > 0$ account for a fraction of α_{1+} among all sampled data.

Let $\xi_1, \xi_2, \dots, \xi_q$ be q scenarios sampled from random variable ξ . We use q binary variables z_1, z_2, \dots, z_q to identify possible infeasibility: $z_k = 1$ implies that constraints cannot be satisfied in scenario ξ_k . To this end, let M be a large enough constant, consider inequality

$$A(\xi_k)x + B(\xi_k)y_k \leq b(\xi_k) + Mz_k \quad (\text{C.131})$$

In (C.131), if $z_k = 0$, recourse action y_k will recover all constraints in scenario ξ_k , and thus $C(x, \xi_k)$ is non-empty; otherwise, if no such a recourse action y_k exists, then constraint violation will take place. To reconcile infeasibility, $z_k = 1$ so that (C.131) becomes redundant, and there is actually no constraint for scenario ξ_k . The fraction of sampled scenarios which will incur inevitable constraint violations is counted by $\sum_{k=1}^q z_k/q$. So we can write out the following MILP reformulation for robust chance-constrained program (C.120) based on VaR and SAA

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & x \in X \\ & A(\xi_k)x + B(\xi_k)y_k \leq b(\xi_k) + Mz_k, \quad k = 1, \dots, q \\ & \sum_{k=1}^q z_k \leq q\alpha'_+, \quad z_k \in \{0, 1\}, \quad k = 1, \dots, q \end{aligned} \quad (\text{C.132})$$

In MILP (C.132), constraint violation can happen in at most $q\alpha_{1+}$ out of q scenarios in the reference distribution, according to Theorem C.1, and the reliability requirement (C.125) under all possible distributions in ambiguity set D_ϕ can be guaranteed by the selection of α'_+ . Improved MILP formulations of chance constraints which do not rely on the specific big-M parameter are comprehensively studied in [52], and some structure properties of the feasible region are revealed.

3) CVaR based reformulation: An LP

The number of binary variables in MILP (C.132) is equal to the number of sampled scenarios. To guarantee the accuracy of SAA, a large number of scenarios are required, preventing MILP (C.132) from being solved efficiently. To ameliorate this plight, we provide a conservative LP approximation for problem (C.120) based on the properties of CVaR revealed in [45].

The β -CVaR for the loss function $L(x, \xi)$ is defined as

$$\beta\text{-CVaR}(x) = \frac{1}{1-\beta} \int_{L(x,\xi) \geq \beta\text{-VaR}(x)} L(x,\xi) f(\xi) d\xi \quad (\text{C.133})$$

which interprets the conditional expectation of loss that is no less than β -VaR; therefore, relation

$$\beta\text{-VaR} \leq \beta\text{-CVaR} \quad (\text{C.134})$$

always holds, and a conservative approximation of constraint (C.130) is

$$(1 - \alpha'_+) \text{-CVaR}(x) \leq 0 \quad (\text{C.135})$$

Inequality (C.135) is a sufficient condition for (C.130) and (C.128). This conservative replacement is apposite to the spirit of robust optimization. In what follows, we will reformulate (C.135) in a solver-compatible form.

According to [45], the left-hand side of (C.135) is equal to the optimum of the following minimization problem

$$\min_{\gamma} \left\{ \gamma + \frac{1}{\alpha'_+} \int_{\xi \in \mathbb{R}^K} \max\{L(x, \xi) - \gamma, 0\} f(\xi) d\xi \right\} \quad (\text{C.136})$$

By performing SAA, the integral in (C.136) renders a summation over discrete sampled scenarios $\xi_1, \xi_2, \dots, \xi_q$, resulting in

$$\min_{\gamma} \left\{ \gamma + \frac{1}{q\alpha'_+} \sum_{k=1}^q \max\{L(x, \xi_k) - \gamma, 0\} \right\} \quad (\text{C.137})$$

By introducing auxiliary variable s_k , the feasible region defined by (C.135) can be expressed via

$$\begin{aligned} \exists \gamma \in \mathbb{R}, s_k \in \mathbb{R}^+, \sigma_k \in \mathbb{R}, \quad k = 1, \dots, q \\ \sigma_k - \gamma \leq s_k, \quad k = 1, \dots, q \\ A(\xi_k)x + B(\xi_k)y_k \leq b(\xi_k) + \sigma_k \mathbf{1}, \quad k = 1, \dots, q \\ \gamma + \frac{1}{q\alpha'_+} \sum_{k=1}^q s_k \leq 0 \end{aligned}$$

Now we can write out the the conservative LP reformulation for robust chance constrained program (C.120) based on CVaR and SAA

$$\begin{aligned} \min_{x, y, s, \gamma} \quad & c^T x \\ \text{s.t.} \quad & x \in X, \quad \gamma + \frac{1}{q\alpha'_+} \sum_{k=1}^q s_k \leq 0, \quad s_k \geq 0, \quad k = 1, \dots, q \\ & A(\xi_k)x + B(\xi_k)y_k - b(\xi_k) \leq (\gamma + s_k) \mathbf{1}, \quad k = 1, \dots, q \end{aligned} \quad (\text{C.138})$$

where σ_k is eliminated.

According to (C.134), condition (C.135) guarantees (C.130) as well as (C.128), so chance constraint in (C.125) holds with a probability no less (usually higher) than $1 - \alpha$, regardless of the true distributions in confidence set D_ϕ . Since (C.134) is usually a strict inequality, this fact will introduce some extent of conservatism in the CVaR based LP model (C.138).

Relations among different mathematical models discussed in this section are summarized in Fig. C.1.

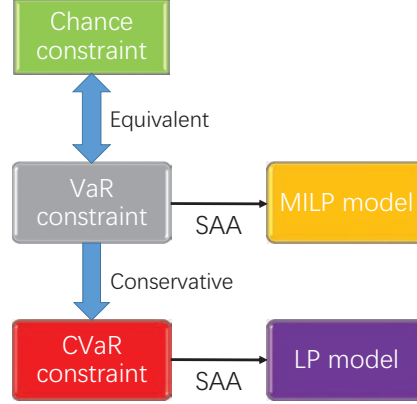


Fig. C.1 Relations of the models discussed in this section.

4. Considering second-stage cost

Finally, we elaborate how to solve problem (C.120) with a second-stage cost in the sense of worst-case expectation, i.e.

$$\begin{aligned}
 \min_x \quad & \left\{ c^T x + \max_{P(\xi) \in D_{KL}} \mathbb{E}_P[Q(x, \xi)] \right\} \\
 \text{s.t.} \quad & x \in X \\
 & \sup_{P(\xi) \in D'} \Pr[C(x, \xi)] \geq 1 - \alpha
 \end{aligned} \tag{C.139}$$

where $Q(x, \xi)$ is the optimal value function of the second-stage problem

$$\begin{aligned}
 Q(x, \xi) = \min \quad & q^T y \\
 \text{s.t.} \quad & B(\xi)y \leq b(\xi) - A(\xi)x
 \end{aligned}$$

which is an LP for a fixed first-stage decision x and a given parameter ξ ;

$$D_{KL} = \{P(\xi) \mid D_\phi^{KL}(f \| f_0) \leq d_{KL}(\alpha^*), f = dP/d\xi\}$$

is the KL-divergence based ambiguity set, and d_{KL} is an α -dependent threshold which determines the size of the ambiguity set, and α^* reflects the confidence level: the real distribution is contained in D_{KL} with a probability no less than α^* . For discrete distributions, the KL-divergence measure has the form of

$$D_{\phi}^{KL}(f \parallel f_0) = \sum_s \rho_s \log \frac{\rho_s}{\rho_s^0}$$

In either case, there are infinitely many PDFs satisfying the inequality in the ambiguity set D_{KL} when $d_{KL} > 0$. Otherwise, when $d_{KL} = 0$, the ambiguity set D_{KL} becomes a singleton, and the model (C.139) degenerates to a traditional SO problem. In practice, the user can specify the value of d_{KL} according to the attitude towards risks. Nevertheless, the proper value of d_{KL} can be obtained from probability theory. Intuitively, the more historical data we possess, the closer the reference PDF f_0 leaves from the true one, and the smaller d_{KL} should be set.

Suppose we have totally M samples with equal probabilities to fit in N bins, and there are M_1, M_2, \dots, M_N samples fall into each bin, then the discrete reference PDF for the histogram is $\{\pi_1, \dots, \pi_N\}$, where $\pi_i = M_i/M$, $i = 1, \dots, N$. Let π_1^r, \dots, π_N^r be the real probability of each bin, according to the discussions in [58], random variable $2M \sum_{i=1}^N \pi_i^r \log(\pi_i^r/\pi_i)$ follows χ^2 distribution with $N - 1$ degrees of freedom. Therefore, the confidence threshold can be calculated from

$$d_{KL}(\alpha^*) = \frac{1}{2M} \chi_{N-1, \alpha^*}^2$$

where χ_{N-1, α^*}^2 stands for the α^* upper quantile of χ^2 distribution with $N - 1$ degrees of freedom. For other divergence based ambiguity sets, please see more discussions in [58]. Robust chance constraints in (C.139) are tackled using the method presented previously, and the objective function will be treated independently. The ambiguity sets in the objective function and chance constraints could be the same one or different ones, and thus are distinguished by D_{KL} and D' .

Sometimes, it is imperative to coordinately optimize the costs in both stages. For example, in the facility planning problem, the first stage represents the investment decision and the second stage describes the operation management. If we only optimize the first-stage cost, then the facilities with lower investment costs will be preferred, but they may suffer from higher operating costs, and not be the optimal choice from the long-term aspect.

To solve (C.139), we need a tractable reformulation for the worst-case expectation problem under KL-divergence ambiguity set

$$\max_{P(\xi) \in D_{KL}} \mathbb{E}_P[Q(x, \xi)] \quad (\text{C.140})$$

under fixed x . It is proved in [50, 58] that problem (C.140) is equivalent to

$$\min_{\alpha \geq 0} \alpha \log \mathbb{E}_{P_0}[e^{Q(x, \xi)/\alpha}] + \alpha d_{KL} \quad (\text{C.141})$$

where α is the dual variable. Formulation (C.141) has two advantages: first, the expectation is evaluated associated with the reference distribution P_0 , which is much easier than optimizing over the ambiguity set D_{KL} ; second, the maximum operator switches to a minimum operator, which is consistent with the objective function of the decision making problem. We will use SAA to express the expectation, giving rise to a discrete version of problem (C.141). In fact, in discrete cases, (C.141) can be derived from (C.140) using Lagrange duality. The following interpretation is given in [51].

Denote by ξ_1, \dots, ξ_s the representative scenarios in the discrete distribution; their corresponding probabilities in the reference PDF and the actual PDF are given by $P_0 = \{p_1^0, \dots, p_s^0\}$ and $P = \{p_1, \dots, p_s\}$, respectively. Then problem (C.140) can be written in a discrete form as

$$\begin{aligned} \max_p \quad & \sum_{i=1}^s p_i Q(x, \xi_i) \\ \text{s.t.} \quad & \sum_{i=1}^s p_i \log \left(\frac{p_i}{p_i^0} \right) \leq d_{KL} \\ & p \geq 0, \quad 1^T p = 1 \end{aligned} \quad (\text{C.142})$$

where vector $p = [p_1, \dots, p_s]^T$ is the decision variable. According to Lagrange duality theory, the objective function of the dual problem is

$$g(\alpha, \mu) = \alpha d_{KL} + \mu + \sum_{i=1}^s \max_{p_i \geq 0} p_i \left(Q(x, \xi_i) - \mu - \alpha \log \left(\frac{p_i}{p_i^0} \right) \right) \quad (\text{C.143})$$

where μ is the dual variable associated with equality constraint $1^T p = 1$, and α with the KL-divergence inequality. Substituting $t_i = p_i/p_i^0$ into (C.143) and eliminating p_i , we get

$$g(\alpha, \mu) = \alpha d_{KL} + \mu + \sum_{i=1}^s \max_{t_i \geq 0} p_i^0 t_i (Q(x, \xi_i) - \mu - \alpha \log t_i)$$

Calculating the first-order derivative of $t_i(Q(x, \xi_i) - \mu - \alpha \log t_i)$ with respect to t_i , the optimal solution is

$$t_i = e^{\frac{Q(x, \xi_i) - \mu - \alpha}{\alpha}} > 0$$

and the maximum is

$$\alpha e^{\frac{Q(x, \xi_i) - \mu - \alpha}{\alpha}}$$

As a result, the dual objective reduces to

$$g(\alpha, \mu) = \alpha d_{KL} + \mu + \alpha \sum_{i=1}^s p_i^0 e^{\frac{Q(x, \xi_i) - \mu - \alpha}{\alpha}} \quad (\text{C.144})$$

and the dual problem of (C.142) can be rewritten as

$$\min_{\alpha \geq 0, \mu} g(\alpha, \mu) \quad (\text{C.145})$$

The optimal solution μ^* must satisfy $\partial g / \partial \mu = 0$, yielding

$$\sum_{i=1}^s p_i^0 e^{\frac{Q(x, \xi_i) - \mu^* - \alpha}{\alpha}} = 1$$

or

$$\mu^* = \alpha \log \sum_{i=1}^s p_i^0 e^{Q(x, \xi_i)/\alpha} - \alpha$$

Substituting above relations into $g(\alpha, \mu)$ results in the following dual problem

$$\min_{\alpha \geq 0} \left\{ \alpha d_{KL} + \alpha \log \sum_{i=1}^s p_i^0 e^{Q(x, \xi_i)/\alpha} \right\} \quad (\text{C.146})$$

which is a discrete form of (C.141).

In (C.139), replacing the inner problem (C.140) with its Lagrangian dual form (C.146), we can obtain an equivalent mathematical program

$$\begin{aligned} \min \quad & \left\{ c^T x + \alpha d_{KL} + \alpha \log \sum_{i=1}^s p_i^0 e^{\theta_i/\alpha} \right\} \\ \text{s.t.} \quad & x \in X, \quad \alpha \geq 0, \quad \theta_i = q^T y_i, \quad \forall i \\ & A(\xi_i)x + B(\xi_i)y_i \leq b(\xi_i), \quad \forall i \\ & \text{Cons-RCC} \end{aligned} \quad (\text{C.147})$$

where Cons-RCC stands for the LP based formulation of robust chance constraints, so the constraints in problem (C.147) are all linear, and the only nonlinearity rests in the last term of the objective function. In what follows, we will show it is actually a convex function in θ_i and α .

In the first step, we claim that the following function is convex ([8], page 87, in Example 3.14)

$$h_1(\theta) = \log \left(\sum_{i=1}^s e^{\theta_i} \right)$$

Since the composition with an affine mapping preserves convexity ([8], Sect. 3.2.2), a new function

$$h_2(\theta) = h_1(A\theta + b)$$

remains convex under linear mapping $\theta \rightarrow A\theta + b$. Let A be an identity matrix, and

$$b = \begin{bmatrix} \log p_1^0 \\ \vdots \\ \log p_s^0 \end{bmatrix}$$

then we have

$$h_2(\theta) = \log \left(\sum_{i=1}^s p_i^0 e^{\theta_i} \right)$$

is a convex function; at last, function

$$h_3(\alpha, \theta) = \alpha h_2(\theta/\alpha)$$

is the perspective of $h_2(\theta)$, so is also convex ([8], page 89, Sect. 3.2.6).

In view of this convex structure, (C.147) essentially gives rise to a convex program, and the local minimum is also the global one. However, according to our experiments, general purpose NLP solvers still have difficulty to solve (C.147). Therefore, we employ the outer approximation method [53, 54]. The motivation is to solve the epigraph form of (C.147), in which nonlinearity is moved into the constraints; then linearize the feasible region with an increasing number of cutting planes generated in an iteration algorithm, until certain convergence criterion is met. In this way, the hard problem (C.147) can be solved via a sequence of LPs. The outer approximation algorithm is outlined in Algorithm C.6. Because (C.147) is a convex program, the cutting planes will not remove any feasible point, and Algorithm C.6 finds the global optimal solution in finite steps, regardless of the initial point. But for sure, the number of iterations is affected by the quality of initial guess. A proper initiation could be obtained by solving a traditional SO problem without considering distribution uncertainty.

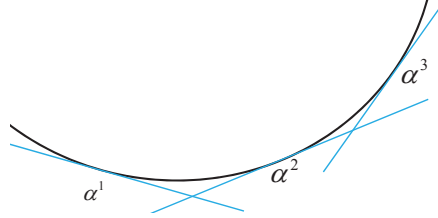


Fig. C.2 Illustration of the outer approximation algorithm.

The motivation of Algorithm C.6 is illustrated in C.2. The original objective function is nonlinear but convex. In the epigraph form (C.148), we generate a set of linear cuts (C.149) dynamically according to the optimal solution found in step 2, then the convex region can be approximated with

Algorithm C.6 Outer Approximation

- 1: Choose an initial point (θ^1, α^1) and convergence tolerance $\epsilon > 0$, the initial objective value is $R^1 = 0$, and iteration index $k = 1$.
- 2: Solve the following master problem which is an LP

$$\begin{aligned}
& \min_{\alpha, \theta, \gamma, x} \quad c^T x + \alpha d_{KL} + \gamma \\
& \text{s.t.} \quad h_3(\alpha^j, \theta^j) + \nabla h_3(\alpha^j, \theta^j) \begin{bmatrix} \alpha - \alpha^j \\ \theta - \theta^j \end{bmatrix} \leq \gamma, \quad j = 1, \dots, k \\
& \quad x \in X, \quad \alpha \geq 0, \quad \theta_i = q^T y_i, \quad \forall i \\
& \quad A(\xi_i)x + B(\xi_i)y_i \leq b(\xi_i), \quad \forall i \\
& \quad \text{Cons-RCC}
\end{aligned} \tag{C.148}$$

- The optimal value is R^{k+1} , and the optimal solution is $(x^{k+1}, \theta^{k+1}, \alpha^{k+1})$.
- 3: If $R^{k+1} - R^k \leq \epsilon$, terminate and report the optimal solution $(x^{k+1}, \theta^{k+1}, \alpha^{k+1})$; otherwise, update $k \leftarrow k + 1$, calculate the gradient ∇h_3 at the obtained solution (α^k, θ^k) , add the following cut to problem (C.148), and go to step 2.

$$h_3(\alpha^k, \theta^k) + \nabla h_3(\alpha^k, \theta^k) \begin{bmatrix} \alpha - \alpha^k \\ \theta - \theta^k \end{bmatrix} \leq \gamma \tag{C.149}$$

arbitrarily high accuracy around the optimal solution. The convergence of the very basic version of outer approximation method has been analyzed in [55, 56]. In fact, Algorithm C.6 is very efficient to solve problem (C.147), because problem (C.148) is an LP, the objective function is smooth, and the algorithm often converges in a few number of iterations.

C.4.2 Stochastic Program with Discrete Distributions

In ARO discussed in Appendix C.2, the uncertain parameter is assumed to reside in the so-called uncertainty set. Every element in this set is treated equally, so the scenario in the worst case must be one of the extreme points of the uncertainty set, which is the main source of conservatism in the traditional RO paradigm. In contrast, in the classic two-stage SO, uncertain parameter ξ is modeled through a certain probability distribution P , and the expected cost is minimized, giving rise to

$$\begin{aligned}
& \min \quad c^T x + \mathbb{E}_P[Q(x, \xi)] \\
& \text{s.t.} \quad x \in X
\end{aligned} \tag{C.150}$$

where the bounded polyhedron X is the feasible region of first-stage decision x , ξ is the uncertain parameter, and $Q(x, \xi)$ is the optimal value function of the second-stage problem, which is an LP for fixed x and ξ

$$\begin{aligned} Q(x, \xi) = \min \quad & q^T y \\ \text{s.t.} \quad & B(\xi)y \leq b(\xi) - A(\xi)x \end{aligned} \quad (\text{C.151})$$

where q is the cost coefficients, $A(\xi)$, $B(\xi)$, and $b(\xi)$ are constant matrices affected by uncertain data, $y(\xi)$ is the second-stage decision, which is the reaction to the realization of uncertainty.

Since the true PDF of ξ is difficult to obtain in some circumstances, in this section, we do not require perfect knowledge on the probability distribution \mathbb{P} of random variable ξ , and let it be ambiguous around a reference distribution and reside in an ambiguity set D , which can be constructed from limited historical data. We take all possible distributions in the ambiguity set into consideration, so as to minimize the expected cost in the worst-case distribution, resulting in the following model

$$\begin{aligned} \min \quad & c^T x + \max_{P(\xi) \in D} \mathbb{E}_P[Q(x, \xi)] \\ \text{s.t.} \quad & x \in X \end{aligned} \quad (\text{C.152})$$

Compared with (C.120), constraint violation is not allowed in problem (C.152), and the second-stage expected cost in the worst-case distribution is considered. It is a particular case of (C.139) without chance constraints. Specifically, we will utilize discrete distributions in this section. This formulation enjoys several benefits. One is the easy exposition of the density function. In previous sections, the candidate in the moment or divergence based ambiguity sets is not given in an analytical form, and vanishes during the dual transformation. As a result, we don't have clear knowledge on the worst-case distribution. For discrete distributions, the density function is a vector of real entries associated with the probability of each representative scenario. We can easily construct the ambiguity set and optimize an expectation over discrete distributions. The other originates from the computational perspective, which can be seen later. Main results in this section come from [61, 62].

1. Modeling the confidence set

For a given set of historical data with M elements, which can be regarded as M samples of the random variable, we can draw a histogram with K bins as an estimation of the reference distribution. Suppose that the numbers of samples fall in each bin is M_1, M_2, \dots, M_K , where $\sum_{i=1}^K M_i = M$, then the reference (empirical) distribution of the uncertain data is given by $\mathbb{P}_0 = [p_1^0, \dots, p_K^0]$, where $p_i^0 = M_i/M$, $i = 1, \dots, K$. Since the data may not be enough to fit a PDF with high accuracy, the actual distribution should be close to but might be different from its reference. It is proposed in [61] to

construct the ambiguity set using statistical inference corresponding to a given tolerance. Two types of ambiguity sets are suggested based on L_1 norm and L_∞ norm

$$D_1 = \{\mathbb{P} \in \mathbb{R}_+^K \mid \|\mathbb{P} - \mathbb{P}_0\|_1 \leq \theta\} = \left\{ p \in \Delta_K \mid \sum_{i=1}^K |p_i - p_i^0| \leq \theta \right\} \quad (\text{C.153})$$

$$D_\infty = \{\mathbb{P} \in \mathbb{R}_+^K \mid \|\mathbb{P} - \mathbb{P}_0\|_\infty \leq \theta\} = \left\{ p \in \Delta_K \mid \max_{1 \leq i \leq K} |p_i - p_i^0| \leq \theta \right\} \quad (\text{C.154})$$

where $\Delta_K = \{p \in [0, 1]^K : \mathbf{1}^T p = 1\}$. These two ambiguity sets can be easily expressed by polyhedral sets as follows

$$D_1 = \left\{ p \in \Delta_K \mid \begin{array}{l} \exists t \in \mathbb{R}_+^K : \sum_{k=1}^K t_k \leq \theta \\ t_k \geq p_k - p_k^0, \quad k = 1, \dots, K \\ t_k \geq p_k^0 - p_k, \quad k = 1, \dots, K \end{array} \right\} \quad (\text{C.155})$$

$$D_\infty = \left\{ p \in \Delta_K \mid \begin{array}{l} \theta \geq p_k - p_k^0, \quad k = 1, \dots, K \\ \theta \geq p_k^0 - p_k, \quad k = 1, \dots, K \end{array} \right\} \quad (\text{C.156})$$

where $p = [p_1, \dots, p_K]^T$ is the variable in the ambiguity set; $t = [t_1, \dots, t_K]^T$ is the lifting (auxiliary) variable in D_1 ; parameter θ reflects decision maker's confidence level on the distance between the reference distribution and the true one. Apparently, the more historical data we utilize, the smaller their distance will be. Provided with M observations and K bins, the quantitative relation between the value of θ and the number of samples are given by [61]

$$\Pr\{\|\mathbb{P} - \mathbb{P}_0\|_1 \leq \theta\} \geq 1 - 2Ke^{-2M\theta/K} \quad (\text{C.157})$$

$$\Pr\{\|\mathbb{P} - \mathbb{P}_0\|_\infty \leq \theta\} \geq 1 - 2Ke^{-2M\theta} \quad (\text{C.158})$$

According to (C.157) and (C.158), if we want to maintain (C.153) and (C.154) with a confidence level of β , parameter θ should be selected as

$$\text{For } D_1 : \theta_1 = \frac{K}{2M} \ln \frac{2K}{1-\beta} \quad (\text{C.159})$$

$$\text{For } D_\infty : \theta_\infty = \frac{1}{2M} \ln \frac{2K}{1-\beta} \quad (\text{C.160})$$

As the size of sampled data approaches infinity, θ_1 and θ_∞ decrease to 0, and the reference distribution converges to the true one. Accordingly, problem (C.152) becomes a traditional two-stage SO.

2. CCG based decomposition algorithm

Let ξ^k denote the representative scenario of the k -th bin, p_k be the corresponding probability, and $P = [p_1, \dots, p_K]$ belongs to the ambiguity set in form of (C.153) or (C.154), then problem (C.152) can be written as

$$\begin{aligned} \min \quad & c^T x + \max_{\mathbb{P}} \sum_{k=1}^K p_k \min q^T y^k \\ \text{s.t.} \quad & x \in X, \mathbb{P} \in D \\ & A(\xi^k)x + B(\xi^k)y^k \leq b(\xi^k), \forall k \end{aligned} \quad (\text{C.161})$$

Problem (C.161) has a min-max-min structure and can be solved by the Benders decomposition method [61] or the CCG method [62]. The latter one will be introduced in the rest of this section. It decomposes problem (C.161) into a lower bounding master problem and an upper bounding subproblem, which are solved iteratively until the gap between the upper bound and lower bound gets smaller than a convergence tolerance. The basic idea has been explained in Appendix C.2.3. As we can see in [62], the second-stage problem can be a broader class of convex programs, such as an SOCP.

1) Subproblem

For a given first-stage decision x , the subproblem aims to find the worst-case distribution, which comes down to a max-min program shown below

$$\max_{\mathbb{P} \in D} \sum_{k=1}^K p_k \min_{y^k \in Y_k(x)} q^T y^k \quad (\text{C.162})$$

where

$$Y_k = \{y^k \mid B(\xi^k)y^k \leq b(\xi^k) - A(\xi^k)x\}, \forall k \quad (\text{C.163})$$

Problem (C.162) has some unique features that facilitate the computation:

- (1) Feasible sets Y_k are decoupled.
- (2) The probability variables p_k do not affect feasible sets Y_k .
- (3) The ambiguity set D and feasible sets Y_k are decoupled.

Although (C.162) seems nonlinear due to the production of scalar variable p_k and vector variable y^k in the objective function, as we can see in the following discussion, it is equivalent to an LP or can be decomposed into several LPs, and thus can be solved efficiently.

An equivalent LP

Because $p_k \geq 0$, we can exchange the summation operator and the minimization operator, and problem (C.162) can be written as

$$\max_{\mathbb{P} \in D} \min_{y^k \in Y_k(x)} \sum_{k=1}^K p_k q^T y^k \quad (\text{C.164})$$

For the inner minimization problem, p_k is constant, so it is an LP, whose dual problem is

$$\begin{aligned} \max_{\mu^k} \quad & \sum_{k=1}^K (b(\xi^k) - A(\xi^k)x)^T \mu^k \\ \text{s.t.} \quad & \mu_k \leq 0, \quad B^T(\xi^k)\mu^k = p_k q, \quad \forall k \end{aligned}$$

where μ^k are dual variables. Substituting it into (C.164), and combining two maximization operators, we obtain

$$\begin{aligned} \max_{p_k, \mu^k} \quad & \sum_{k=1}^K (b(\xi^k) - A(\xi^k)x)^T \mu^k \\ \text{s.t.} \quad & \mu_k \leq 0, \quad B^T(\xi^k)\mu^k = p_k q, \quad \forall k \\ & (p_1, \dots, p_k) \in D \end{aligned} \tag{C.165}$$

Since D is polyhedral, problem (C.165) is in fact an LP. The optimal solution offers the worst-case distribution $[p_1^*, \dots, p_K^*]$, which will be used to generate cuts in the master problem. The recourse actions y^k in each scenario will be provided by the optimal solution of the master problem.

Despite of the fact that LP is acknowledged as the most tractable mathematical programming problem, however, when K is extremely large, it is still challenging to solve (C.165) or even store it in a computer. Nevertheless, the separability of feasible regions allows solving (C.162) in a decomposition manner.

A decomposition method

As mentioned above, p_k has no impact on Y_k , which are decoupled; moreover, because p_k is a scalar in the objective function of each inner minimization problem, it does not affect the optimal solution y^k . In view of this convenience, problem (C.162) can be decomposed into $K + 1$ smaller LPs, and can be solved in parallel. To this end, for each ξ^k , solve the following LP:

$$h_k^* = \min_{y^k \in Y_k(x)} q^T y^k, \quad k = 1, \dots, K$$

The optimal value is h_k^* ; after obtaining optimal values (h_1^*, \dots, h_K^*) of the K LPs, we can retrieve the worst-case distribution through solving an additional LP

$$\max_{p \in D} \sum_{k=1}^K p_k h_k^*$$

In fact, if the second-stage problem is a conic program (in [62], it is an SOCP), above discussions are still valid, as long as the strong duality holds.

It is interesting to notice that in the ARO problem in Sect. C.2.3, the subproblem comes down to a non-convex bilinear program after dualizing the inner minimization problem, and is generally NP-hard; in this section,

the subproblem actually gives rise to LPs, whose complexity is polynomial in problem sizes. The reason accounting for this difference is that the uncertain parameter in (C.161) is expressed by sampled scenarios and thus is constant; the distributional uncertainty appearing in the objective function does not influence the constraints of the second stage problem, and thus the linear max-min problem (C.164) reduces to an LP after a dual transformation.

2) The CCG algorithm

The motivation of CCG algorithm has been thoroughly discussed in Appendix C.2.3. In this section, for a fixed x , the optimal value of subproblem (C.162) is denoted by $Q(x)$, and $c^T x + Q(x)$ gives an upper bound of the optimal solution of (C.161), because the first-stage variable is un-optimized. Then a set of new variables and optimality cuts are generated and added into master problem. If the subproblem is infeasible in some scenario, then a set of feasibility cuts are assigned to the master problem. The master problem starts from a subset of D , which is updated by including the worst-case distribution identified by the subproblem. Forasmuch, the master problem is a relax version of the original problem (C.161), and provides a lower bound on the optimal value. The flowchart of the CCG procedure for problem (C.161) is given in Algorithm C.7. This algorithm will terminate in a finite number of iterations, as the confidence set D has finite extreme points.

C.4.3 Formulations based on Wasserstein Metric

Up to now, the KL-divergence based ambiguity set based formulations have received plenty of research, because it enjoys some convenience when deriving the robust counterpart. For example, it has already known in Sect. C.4.1 that robust chance constraints under KL-divergence ambiguity set can reduce to a traditional chance constraints under the empirical distribution with a rescaled confidence level, and the worst-case expectation problem under KL-divergence ambiguity set is equivalent to a convex program. However, according to its definition, KL-divergence ambiguity set may encounter theoretical difficulty to represent confidence sets for continuous distribution [72], because the empirical distribution calibrated from finite data must be discrete, and any distribution in the KL-divergence ambiguity set must assign positive probability mass to each sampled scenario. As a continuous distribution has a density function, it must reside outside the KL-divergence ambiguity set regardless of the sampled scenarios. In contrast, Wasserstein metric based ambiguity sets contain both discrete and continuous distributions. It offers an explicit confidence level for the unknown distribution belonging to the set, and enables the decision maker more informative guidance to control the model conservativeness. This section introduces state-of-the-art results in robust SO with Wasserstein metric based ambiguity sets. The most critical

Algorithm C.7

-
- 1: Choose a convergence tolerance $\varepsilon > 0$, and an initial probability vector $p^0 \in D$; Set $\text{LB} = -\infty$, $\text{UB} = +\infty$, and iteration index $s = 0$.
 - 2: Solve the master problem

$$\begin{aligned}
& \min_{x, \eta, y^{k,m}} c^T x + \eta \\
& \text{s.t. } x \in X, \quad \eta \geq \sum_{k=1}^K p_k^m q^T y^{k,m}, \quad m \in \text{Opt}\{0, 1, \dots, s\} \\
& \quad A(\xi^k)x + B(\xi^k)y^{k,m} \leq b(\xi^k), \quad m \in \text{Opt}\{0, 1, \dots, s\}, \quad \forall k \\
& \quad A(\xi^k)x + B(\xi^k)y^{k,m} \leq b(\xi^k), \quad m \in \text{Fea}\{0, 1, \dots, s\}, \quad k \in I(s)
\end{aligned} \tag{C.166}$$

where $\text{Opt}\{*\}/\text{Fea}\{*\}$ selects the iterations in which an optimality (feasibility) cut is generated; $I(s)$ depicts the index of scenarios in which the second-stage problem is infeasible in iteration s . The optimal solution is (x^*, η^*) ; update $\text{LB} = c^T x^* + \eta^*$;

- 3: Solve subproblem (C.162) with current x^* . If there exists some ξ^k such that $Y_k(x^*) = \emptyset$, then generate new variable $y^{k,s}$, update $I(s)$, and add the following feasibility cut to the master problem

$$A(\xi^k)x + B(\xi^k)y^{k,s} \leq b(\xi^k), \quad k \in I(s) \tag{C.167}$$

Otherwise, if $Y_k(x^*) \neq \emptyset, \forall k$, subproblem (C.162) can be solved. The optimal solution is p^{s+1} , and the optimal value is $Q(x^*)$; update $\text{UB} = \min\{\text{UB}, c^T x^* + Q(x^*)\}$, create new variables $(y^{1,s+1}, \dots, y^{k,s+1})$, and add the following optimality cut to the master problem

$$\begin{aligned}
& \eta \geq \sum_{k=1}^K p_k^{s+1} q^T y^{k,s+1} \\
& A(\xi^k)x + B(\xi^k)y^{k,s+1} \leq b(\xi^k), \quad \forall k
\end{aligned} \tag{C.168}$$

- 4: If $\text{UB} - \text{LB} < \varepsilon$, terminate and report the optimal first-stage solution x^* as well as the worst-case distribution p^{s+1} ; otherwise, update $s \leftarrow s + 1$, and go to step 2.
-

problem is the robust counterparts of the worst-case expectation problem and robust chance constraints, which will be discussed respectively. They can be embedded in single- and two-stage robust SO problems without substantial barriers. The materials in this section mainly come from [72].

1. Wasserstein metric based ambiguity set

Let Ξ be the support set of multi-dimensional random variable $\xi \in \mathbb{R}^m$. $M(\Xi)$ represent all probability distributions \mathbb{Q} supported on Ξ , and $\mathbb{E}_{\mathbb{Q}}[\|\xi\|] = \int_{\Xi} \|\xi\| \mathbb{Q}(d\xi) < \infty$, where $\|\cdot\|$ stands for an arbitrary norm on \mathbb{R}^m .

Definition C.8. Wasserstein metric $d_W : M(\Xi) \times M(\Xi) \rightarrow \mathbb{R}_+$ is defined as

$$d_W(\mathbb{Q}, \mathbb{Q}_0) = \inf \left(\int_{\Xi^2} \|\xi - \xi^0\| \Pi(d\xi, d\xi^0) \mid \begin{array}{l} \Pi \text{ is a joint distribution of } \xi \text{ and } \\ \xi^0 \text{ with marginals } \mathbb{Q} \text{ and } \mathbb{Q}_0 \end{array} \right)$$

for two probability distributions $\mathbb{Q}, \mathbb{Q}_0 \in M(\Xi)$.

As a special case, for two discrete distributions, Wasserstein metric is given by

$$d_W(\mathbb{Q}, \mathbb{Q}_0) = \inf_{\pi \geq 0} \left(\sum_i \sum_j \pi_{ij} \|\xi_j - \xi_i^0\| \mid \begin{array}{l} \sum_j \pi_{ij} = p_i^0, \forall i \\ \sum_i \pi_{ij} = p_j, \forall j \end{array} \right) \quad (\text{C.169})$$

where p_i^0 and p_j denote the probability of representative scenario ξ_i^0 and ξ_j .

In either case, the decision variable Π (or π_{ij}) represents the probability mass transported from ξ_i^0 to ξ_j , therefore, the Wasserstein metric can be viewed as the minimal cost of a transportation plan, where the distance $\|\xi_j - \xi_i^0\|$ encodes the transportation cost of unit mass.

Sometimes, the Wasserstein metric can be represented in the dual form

$$d_W(\mathbb{Q}, \mathbb{Q}_0) = \sup_{f \in L} \left(\int_{\Xi} f(\xi) \mathbb{Q}(d\xi) - \int_{\Xi} f(\xi) \mathbb{Q}_0(d\xi) \right) \quad (\text{C.170})$$

where $L = \{f : |f(\xi) - f(\xi^0)| \leq \|\xi - \xi^0\|, \forall \xi, \xi^0 \in \Xi\}$ (Theorem 3.2, [72], which was firstly discovered by Kantorovich and Rubinstein [73] for distributions with a bounded support).

With above definition, the Wasserstein ambiguity set is the ball of radius ϵ centered at the empirical distribution \mathbb{Q}_0

$$D_W = \{\mathbb{Q} \in M(\Xi) : d_W(\mathbb{Q}, \mathbb{Q}_0) \leq \epsilon\} \quad (\text{C.171})$$

where \mathbb{Q}_0 is constructed with N independent data samples

$$\mathbb{Q}_0 = \frac{1}{N} \sum_{i=1}^N \delta_{\xi_i^0}$$

where $\delta_{\xi_i^0}$ stands for Dirac distribution concentrating unit mass at ξ_i^0 .

Particularly, we require the unknown distribution \mathbb{Q} follow a light tail assumption, i.e., there exists $a > 1$ such that

$$\int_{\Xi} e^{\|\xi\|^a} \mathbb{Q}(d\xi) < \infty$$

This assumption indicates that the tail of distribution \mathbb{Q} decays at an exponential rate. If Ξ is bounded and compact, this assumption trivially holds. Under this assumption, modern measure concentration theory provides the following finite sample guarantee for the unknown distribution belonging to Wasserstein ambiguity set

$$\Pr [d_W(\mathbb{Q}, \mathbb{Q}_0) \geq \epsilon] \leq \begin{cases} c_1 e^{-c_2 N \epsilon^{\max\{m, 2\}}} & \text{if } \epsilon \leq 1 \\ c_1 e^{-c_2 N \epsilon^a} & \text{if } \epsilon > 1 \end{cases} \quad (\text{C.172})$$

where c_1, c_2 are positive constants depending on a, A , and m and $m \neq 2$.

Equation (C.172) provides a priori estimate of the confidence level for $\mathbb{Q} \notin D_W$. On the other hand, we can utilize (C.172) to select parameter ϵ of the Wasserstein ambiguity set such that D_W contains the uncertain distribution \mathbb{Q} with probability $1 - \beta$ for some prescribed β . This requires solving ϵ from the right-hand side of (C.172) with a given left-hand side β , resulting in

$$\epsilon = \begin{cases} \left(\frac{\ln(c_1 \beta^{-1})}{c_2 N} \right)^{1/\max\{m, 2\}} & \text{if } N \geq \frac{\ln(c_1 \beta^{-1})}{c_2} \\ \left(\frac{\ln(c_1 \beta^{-1})}{c_2 N} \right)^{1/a} & \text{if } N < \frac{\ln(c_1 \beta^{-1})}{c_2} \end{cases} \quad (\text{C.173})$$

Wasserstein ambiguity set with above radius can be regarded as a confidence set for the unknown distribution \mathbb{Q} as in statistical testing.

2. Worst-case expectation problem

A robust SO problem under Wasserstein metric naturally requests to minimize the worst-case expected cost:

$$\inf_{x \in X} \sup_{\mathbb{Q} \in D_W} \mathbb{E}_{\mathbb{Q}}[h(x, \xi)] \quad (\text{C.174})$$

We demonstrate how to solve the core problem: the worst-case expectation

$$\sup_{\mathbb{Q} \in D_W} \mathbb{E}_{\mathbb{Q}}[l(\xi)] \quad (\text{C.175})$$

where $l(\xi) = \max_{1 \leq k \leq K} l_k(\xi)$ is the payoff function, consisting of the point-wise maximum of K elementary functions. For notation brevity, the dependence on x is suppressed and will be recovered later on when necessary. We further assume that the support set Ξ is closed and convex, and specific $l(\xi)$ will be discussed.

Problem (C.175) renders an infinite-dimensional optimization problem for continuous distribution. Nonetheless, the inspiring work in [72] show that (C.175) can be reformulated as a finite-dimensional convex program for various payoff functions. To see this, expand the worst-case expectation as

$$\sup_{\mathbb{Q} \in D_W} \mathbb{E}_{\mathbb{Q}}[l(\xi)] = \begin{cases} \sup_{\Pi} \int_{\Xi} l(\xi) \mathbb{Q}(\mathrm{d}\xi) \\ \text{s.t.} \int_{\Xi^2} \|\xi - \xi^0\| \Pi(\mathrm{d}\xi, \mathrm{d}\xi^0) \leq \epsilon \\ \Pi \text{ is a joint distribution of } \xi \\ \xi^0 \text{ with marginals } \mathbb{Q} \text{ and } \mathbb{Q}_0 \end{cases}$$

According to the law of total probability, Π can be decomposed as the marginal distribution \mathbb{Q}_0 of ξ^0 and the conditional distributions \mathbb{Q}_i of ξ given $\xi^0 = \xi_i^0$:

$$\Pi = \frac{1}{N} \sum_{i=1}^N \delta_{\xi_i^0} \otimes \mathbb{Q}_i$$

and the worst-case expectation evolves into a generalized moment problem in conditional distributions \mathbb{Q}_i , $i \leq N$

$$\sup_{\mathbb{Q} \in D_W} \mathbb{E}_{\mathbb{Q}}[l(\xi)] = \begin{cases} \sup_{\mathbb{Q}_i \in M(\Xi)} \frac{1}{N} \sum_{i=1}^N \int_{\Xi} l(\xi) \mathbb{Q}_i(\mathrm{d}\xi) \\ \text{s.t.} \quad \frac{1}{N} \sum_{i=1}^N \int_{\Xi} \|\xi - \xi_i^0\| \mathbb{Q}_i(\mathrm{d}\xi) \leq \epsilon \end{cases}$$

Using standard Lagrangian duality, we obtain

$$\begin{aligned} \sup_{\mathbb{Q} \in D_W} \mathbb{E}_{\mathbb{Q}}[l(\xi)] &= \sup_{\mathbb{Q}_i \in M(\Xi)} \inf_{\lambda \geq 0} \frac{1}{N} \sum_{i=1}^N \int_{\Xi} l(\xi) \mathbb{Q}_i(\mathrm{d}\xi) \\ &\quad + \lambda \left(\epsilon - \frac{1}{N} \sum_{i=1}^N \int_{\Xi} \|\xi - \xi_i^0\| \mathbb{Q}_i(\mathrm{d}\xi) \right) \\ &\leq \inf_{\lambda \geq 0} \sup_{\mathbb{Q}_i \in M(\Xi)} \lambda \epsilon + \frac{1}{N} \sum_{i=1}^N \int_{\Xi} (l(\xi) - \lambda \|\xi - \xi_i^0\|) \mathbb{Q}_i(\mathrm{d}\xi) \\ &= \inf_{\lambda \geq 0} \lambda \epsilon + \frac{1}{N} \sum_{i=1}^N \sup_{\xi \in \Xi} (l(\xi) - \lambda \|\xi - \xi_i^0\|) \end{aligned}$$

Decision variables λ and ξ have finite dimensions. The last problem can be reformulated as

$$\begin{aligned}
& \inf_{\lambda, s_i} \lambda \epsilon + \frac{1}{N} \sum_{i=1}^N s_i \\
& \text{s.t. } \sup_{\xi \in \Xi} (l_k(\xi) - \lambda \|\xi - \xi_i^0\|) \leq s_i \\
& \quad i = 1, \dots, N, \quad k = 1, \dots, K \\
& \quad \lambda \geq 0
\end{aligned} \tag{C.176}$$

From the definition of dual norm, we know $\lambda \|\xi - \xi_i^0\| = \max_{\|z_{ik}\|_* \leq \lambda} \langle z_{ik}, \xi - \xi_i^0 \rangle$, so the constraints give rise to

$$\begin{aligned}
\sup_{\xi \in \Xi} (l_k(\xi) - \lambda \|\xi - \xi_i^0\|) &= \sup_{\xi \in \Xi} \left(l_k(\xi) - \max_{\|z_{ik}\|_* \leq \lambda} \langle z_{ik}, \xi - \xi_i^0 \rangle \right) \\
&= \sup_{\xi \in \Xi} \min_{\|z_{ik}\|_* \leq \lambda} l_k(\xi) - \langle z_{ik}, \xi - \xi_i^0 \rangle \\
&\leq \min_{\|z_{ik}\|_* \leq \lambda} \sup_{\xi \in \Xi} l_k(\xi) - \langle z_{ik}, \xi - \xi_i^0 \rangle
\end{aligned}$$

Substituting it into problem (C.176) leads to a more restricted feasible set and a larger objective value, yielding

$$\begin{aligned}
& \inf_{\lambda, s_i} \lambda \epsilon + \frac{1}{N} \sum_{i=1}^N s_i \\
& \text{s.t. } \min_{\|z_{ik}\|_* \leq \lambda} \sup_{\xi \in \Xi} l_k(\xi) - \langle z_{ik}, \xi - \xi_i^0 \rangle \leq s_i \\
& \quad i = 1, \dots, N, \quad k = 1, \dots, K \\
& \quad \lambda \geq 0
\end{aligned} \tag{C.177}$$

The constraints of (C.177) trivially suggests the feasible set of λ is $\lambda \geq \|z_{ik}\|_*$, and the min operator in constraints can be omitted because it is in compliance with the objective function. Therefore, we arrive at

$$\begin{aligned}
& \inf_{\lambda, s_i} \lambda \epsilon + \frac{1}{N} \sum_{i=1}^N s_i \\
& \text{s.t. } \sup_{\xi \in \Xi} (l_k(\xi) - \langle z_{ik}, \xi \rangle) + \langle z_{ik}, \xi_i^0 \rangle \leq s_i, \quad \lambda \geq \|z_{ik}\|_* \\
& \quad i = 1, \dots, N, \quad k = 1, \dots, K
\end{aligned} \tag{C.178}$$

It is proved in [72] that problems (C.175) and (C.178) are actually equivalent. Next, we will derive the concrete forms of (C.178) under specific payoff function $l(\xi)$ and uncertainty set Ξ . Unlike [72] which relies on conjugate functions in convex analysis, we mainly exploit LP duality theory, which is more friendly to readers with engineering background.

Case 1: Convex PWL payoff function $l(\xi) = \max_{1 \leq k \leq K} \{a_k^T \xi + b_k\}$ and bounded polyhedral uncertainty set $\Xi = \{\xi \in \mathbb{R}^m : C\xi \leq d\}$. The key point is the supremum regarding ξ in the following constraint

$$\sup_{\xi \in \Xi} (a_k^T \xi - \langle z_{ik}, \xi \rangle) + b_k + \langle z_{ik}, \xi_i^0 \rangle \leq s_i$$

For each k , the supremum is an LP

$$\begin{aligned} \max \quad & (a_k - z_{ik})^T \xi \\ \text{s.t.} \quad & C\xi \leq d \end{aligned}$$

Its dual LP reads

$$\begin{aligned} \min \quad & d^T \gamma_{ik} \\ \text{s.t.} \quad & C^T \gamma_{ik} = a_k - z_{ik} \\ & \gamma_{ik} \geq 0 \end{aligned}$$

Therefore, $z_{ik} = a_k - C^T \gamma_{ik}$. Because of strong duality, we can replace the supremum by the objective of the dual LP, which gives rise to:

$$d^T \gamma_{ik} + b_k + \langle a_k - C^T \gamma_{ik}, \xi_i^0 \rangle \leq s_i, \quad i = 1, \dots, N, \quad k = 1, \dots, K$$

Arrange all constraints together, we obtain a convex program which is equivalent to problem (C.178) in Case 1:

$$\begin{aligned} \inf_{\lambda, s_i} \quad & \lambda \epsilon + \frac{1}{N} \sum_{i=1}^N s_i \\ \text{s.t.} \quad & b_k + a_k^T \xi_i^0 + \gamma_{ik}^T (d - C^T \xi_i^0) \leq s_i, \quad i = 1, \dots, N, \quad k = 1, \dots, K \\ & \lambda \geq \|a_k - C^T \gamma_{ik}\|_*, \quad \gamma_{ik} \geq 0, \quad i = 1, \dots, N, \quad k = 1, \dots, K \end{aligned} \quad (\text{C.179})$$

In the absence of distributional uncertainty, or $\epsilon = 0$ which implies that Wasserstein ambiguity set D_W is a singleton, λ can take any non-negative value without changing the objective function. Because all sampled scenarios must belong to the support set, i.e. $d - C^T \xi_i^0 \geq 0, \forall i$ holds, so there must be $\gamma_{ik} = 0$ at the optimal solution, leading to an optimal value of $\sum_{i=1}^N s_i / N$, where $s_i = \max_{1 \leq k \leq K} \{a_k^T \xi_i^0 + b_k\}$, which represents the sample average of the payoff function under the empirical distribution.

Case 2: Concave PWL payoff function $l(\xi) = \min_{1 \leq k \leq K} \{a_k^T \xi + b_k\}$ and bounded polyhedral uncertainty set $\Xi = \{\xi \in \mathbb{R}^m : C\xi \leq d\}$. In such circumstance, the supremum regarding ξ in the constraint becomes

$$\max_{\xi \in \Xi} \left\{ -z_i^T \xi + \min_{1 \leq k \leq L} \{a_k^T \xi + b_k\} \right\}$$

which is equivalent to an LP

$$\begin{aligned}
\max \quad & -z_i^T \xi + \tau_i \\
\text{s.t.} \quad & A\xi + b \geq \tau_i \mathbf{1} \\
& C\xi \leq d
\end{aligned}$$

where the k -th row of A is a_k^T ; the k -th entry of b is b_k ; $\mathbf{1}$ is all-one vector with a compatible dimension. Its dual LP reads

$$\begin{aligned}
\min \quad & b^T \theta_i + d^T \gamma_i \\
\text{s.t.} \quad & -A^T \theta_i + C^T \gamma_i = -z_i \\
& \mathbf{1}^T \theta_i = 1, \theta_i \geq 0, \gamma_i \geq 0
\end{aligned}$$

Therefore, $z_i = A^T \theta_i - C^T \gamma_i$. Because of strong duality, we can replace the supremum by the objective of the dual LP, which gives rise to:

$$b^T \theta_i + d^T \gamma_i + \langle A^T \theta_i - C^T \gamma_i, \xi_i^0 \rangle \leq s_i, \quad i = 1, \dots, N$$

Arrange all constraints together, we obtain a convex program which is equivalent to problem (C.178) in Case 2:

$$\begin{aligned}
\inf_{\lambda, s_i} \quad & \lambda \epsilon + \frac{1}{N} \sum_{i=1}^N s_i \\
\text{s.t.} \quad & \theta_i^T (b + A\xi_i^0) + \gamma_i^T (d - C\xi_i^0) \leq s_i, \quad i = 1, \dots, N \\
& \lambda \geq \|A^T \theta_i - C^T \gamma_i\|_*, \quad i = 1, \dots, N \\
& \gamma_i \geq 0, \theta_i \geq 0, \mathbf{1}^T \theta_i = 1, \quad i = 1, \dots, N
\end{aligned} \tag{C.180}$$

There will be no k index for the constraints, because it is packaged in A and b .

An analogous analysis shows that if $\epsilon = 0$, there must be $\gamma_i = 0$ and

$$s_i = \min\{\theta_i^T (b + A\xi_i^0) : \theta_i \geq 0, \mathbf{1}^T \theta_i = 1\} = \min_{1 \leq k \leq K} \{a_k^T \xi + b_k\}$$

implying $\sum_{i=1}^N s_i/N$ is the sample average of the payoff function under the empirical distribution.

Now we focus our attention on the min-max problem (C.174) which frequently arises in two-stage robust SO, which entails evaluation of the expected recourse cost from an LP parameterized in ξ . We investigate two cases depending on where ξ appears.

Case 3: Uncertain cost coefficients: $l(\xi) = \min_y \{y^T Q\xi : Wy \geq h - Ax\}$ where x is the first-stage decision variable, y represents the recourse action, and the feasible region is always non-empty. In this case, the supremum regarding ξ in the constraint becomes

$$\begin{aligned}
& \max_{\xi \in \Xi} \left\{ -z_i^T \xi + \min_y \{ y^T Q \xi : Wy \geq h - Ax \} \right\} \\
& = \min_y \left\{ \max_{\xi \in \Xi} \left\{ (Q^T y - z_i)^T \xi \right\} : Wy \geq h - Ax \right\}
\end{aligned}$$

Replace the inner LP with its dual, we get an equivalent LP

$$\begin{aligned}
& \min_{\gamma_i, y_i} d^T \gamma_i \\
& \text{s.t. } C^T \gamma_i = Q^T y_i - z_i, \quad \gamma_i \geq 0 \\
& \quad Wy_i \geq h - Ax
\end{aligned}$$

Here we associated variable y with a subscript i to highlight its dependence on the value of ξ . Therefore, $z_i = Q^T y_i - C^T \gamma_i$, and we can replace the supremum by the objective of the dual LP, which gives rise to:

$$d^T \gamma_i + \langle Q^T y_i - C^T \gamma_i, \xi_i^0 \rangle \leq s_i, \quad i = 1, \dots, N$$

Arrange all constraints together, we obtain a convex program which is equivalent to problem (C.178) in Case 3:

$$\begin{aligned}
& \inf_{\lambda, s_i} \lambda \epsilon + \frac{1}{N} \sum_{i=1}^N s_i \\
& \text{s.t. } y_i^T Q \xi_i^0 + \gamma_i^T (d - C^T \xi_i^0) \leq s_i, \quad i = 1, \dots, N \\
& \quad \lambda \geq \|Q^T y_i - C^T \gamma_i\|_*, \quad \gamma_i \geq 0, \quad i = 1, \dots, N \\
& \quad Wy_i \geq h - Ax, \quad i = 1, \dots, N
\end{aligned} \tag{C.181}$$

Without distributional uncertainty, $\epsilon = 0$, λ can be arbitrary nonnegative value; for similar reason, we have $\gamma_i = 0$ and $s_i = y_i^T Q \xi_i^0$ at optimum. So problem (C.181) is equivalent to the SAA problem under the empirical distribution

$$\min_{y_i} \left\{ \frac{1}{N} \sum_{i=1}^N y_i^T Q \xi_i^0 : Wy_i \geq h - Ax \right\}$$

Case 4: Uncertain constraint right-hand side:

$$\begin{aligned}
l(\xi) &= \min_y \{ q^T y : Wy \geq H\xi + h - Ax \} \\
&= \max_{\theta} \{ \theta^T (H\xi + h - Ax) : W^T \theta = q, \theta \geq 0 \} \\
&= \max_k v_k^T (H\xi + h - Ax) = \max_k \{ v_k^T H\xi + v_k^T (h - Ax) \}
\end{aligned}$$

where v_k is the vertices of polyhedron $\{\theta : W^T \theta = q, \theta \geq 0\}$. In this way, $l(\xi)$ is expressed as a convex PWL function. Applying the result in Case 1, we obtain a convex program which is equivalent to problem (C.178) in Case 4:

$$\begin{aligned}
& \inf_{\lambda, s_i} \lambda \epsilon + \frac{1}{N} \sum_{i=1}^N s_i \\
& \text{s.t. } v_k^T(h - Ax) + v_k^T H \xi_i^0 + \gamma_{ik}^T(d - C^T \xi_i^0) \leq s_i, \quad i = 1, \dots, N, \quad \forall k \\
& \lambda \geq \|H^T v_k - C^T \gamma_{ik}\|_*, \quad \gamma_{ik} \geq 0, \quad i = 1, \dots, N, \quad \forall k
\end{aligned} \tag{C.182}$$

For similar reason, without distributional uncertainty, we have $\gamma_{ik} = 0$ and $s_i = v_k^T(h - Ax) + v_k^T H \xi_i^0 = q^T y_i$ at optimum, where the last equality is because of strong duality. So problem (C.181) is equivalent to the SAA problem under the empirical distribution

$$\min_{y_i} \left\{ \frac{1}{N} \sum_{i=1}^N q^T y_i : W y_i \geq H \xi + h - Ax \right\}$$

The following discussions are devoted to the computational tractability.

- If the 1-norm or ∞ -norm is used to define Wasserstein metric, their dual norms are ∞ -norm and 1-norm respectively, then problems (C.179)-(C.182) reduce to LPs whose sizes grow with the number N of sampled data. If the Euclidean norm is used, the resulting problems will be SOCP.
- For Case 1, Case 2 and Case 3, the remaining equivalent LPs scale polynomially and can be therefore readily solved. As for Case 4, the number of vertices may grow exponential in the problem size. However, one can adopt a decomposition algorithm similar to CCG which iteratively identifies critical vertices without enumerating all of them.
- The computational complexity of all equivalent convex programs is independent of the size of the Wasserstein ambiguity set.
- It is shown in [72] that the worst-case expectation can also be computed from the following problem

$$\begin{aligned}
& \sup_{\alpha_{ik}, q_{ik}} \frac{1}{N} \sum_{i=1}^N \sum_{k=1}^K \alpha_{ik} l_k \left(\xi_i^0 - \frac{q_{ik}}{\alpha_{ik}} \right) \\
& \text{s.t. } \frac{1}{N} \sum_{i=1}^N \sum_{k=1}^K \|q_{ik}\| \leq \epsilon \\
& \alpha_{ik} \geq 0, \forall i, \forall k, \quad \sum_{k=1}^K \alpha_{ik} = 1, \forall i \\
& \xi_i^0 - \frac{q_{ik}}{\alpha_{ik}} \in \Xi, \quad \forall i, \forall k
\end{aligned} \tag{C.183}$$

Non-convex term arise from the fraction q_{ik}/α_{ik} . In fact, problem (C.183) is convex following the definition of extended perspective function [72]. Moreover, if $[\alpha_{ik}(r), q_{ik}(r)]_{r \in \mathbb{N}}$ is a sequence of feasible solutions and the corresponding objective values converge to the supremum of (C.183), then

the discrete distribution

$$\mathbb{Q}_r = \frac{1}{N} \sum_{i=1}^N \sum_{k=1}^K \alpha_{ik}(r) \delta_{\xi_{ik}(r)}, \quad \xi_{ik}(r) = \xi_i^0 - \frac{q_{ik}(r)}{\alpha_{ik}(r)}$$

approaches the worst-case distribution in D_W [72].

3. Static robust chance constraints

Another important issue in SO is chance constraint. Here we discuss robust joint chance constraints in the following form

$$\inf_{\mathbb{Q} \in D_W} \Pr[a(x)^T \xi_i \leq b_i(x), i = 1, \dots, I] \geq 1 - \beta \quad (\text{C.184})$$

where x is the decision variable; the chance constraint involves I inequalities with uncertain parameter ξ_i supported on set $\Xi_i \subseteq \mathbb{R}^n$ for each i . The joint probability distribution \mathbb{Q} belongs to the Wasserstein ambiguity set. $a(x) \in \mathbb{R}^n$ and $b_i(x) \in \mathbb{R}$ are affine mappings of x , where $a(x) = \eta x + (1 - \eta)\mathbf{1}$, $\eta \in \{0, 1\}$, and $b_i(x) = B_i^T x + b_i^0$. When $\eta = 1$ ($\eta = 0$), (C.184) involves left-hand (right-hand) uncertainty. $\Xi = \prod_i \Xi_i$ is the support set of $\xi = [\xi_1^T, \dots, \xi_I^T]^T$. The robust chance constraint (C.184) requires that all inequalities be met for all possible distributions in Wasserstein ambiguity set D_W with a probability of at least $1 - \beta$, where $\beta \in (0, 1)$ denotes a prescribed risk tolerance. The feasible region stipulated by (C.184) is X . We will introduce main results from [74] while avoiding rigorous mathematical proofs.

Assumption C.6 *The support set Ξ is an $n \times I$ -dimensional vector space, and the distance metric in Wasserstein ambiguity set is $d(\xi, \zeta) = \|\xi - \zeta\|$.*

Theorem C.2. [74] *Under Assumption (C.6), $X = Z_1 \cup Z_2$, where*

$$Z_1 = \left\{ x \in \mathbb{R}^n \left| \begin{array}{l} \epsilon v - \beta \gamma \leq \frac{1}{N} \sum_{j=1}^N z_j \\ z_j + \gamma \leq \max \left\{ b_i(x) - a(x)^T \zeta_i^j, 0 \right\} \\ i = 1, \dots, I, \quad j = 1, \dots, N \\ z_j \leq 0, \quad j = 1, \dots, N \\ \|a(x)\|_* \leq v, \quad \gamma \geq 0 \end{array} \right. \right\} \quad (\text{C.185})$$

where ϵ is the radius of the Wasserstein ambiguity set, N is the number of sampled scenarios in the empirical distribution, and

$$Z_2 = \{x \in \mathbb{R}^n \mid a(x) = 0, \quad b_i(x) \geq 0, \quad i = 1, \dots, I\} \quad (\text{C.186})$$

In Theorem C.2, Z_2 is trivial: If $\eta = 1$, then $Z_2 = \{x \in \mathbb{R}^n \mid x = 0, b_i \geq 0, \forall i\}$; If $\eta = 0$, then $Z_2 = \emptyset$. Z_1 can be reformulated as an MILP compatible form if it is bounded. By linearizing the second constraint, we have

$$Z_1 = \left\{ x \in \mathbb{R}^n \mid \begin{array}{l} \epsilon v - \beta \gamma \leq \frac{1}{N} \sum_{j=1}^N z_j \\ z_j + \gamma \leq s_{ij}, \forall i, \forall j \\ b_i(x) - a(x)^T \zeta_i^j \leq s_{ij} \leq M_{ij} y_{ij}, \forall i, \forall j \\ s_{ij} \leq b_i(x) - a(x)^T \zeta_i^j + M_{ij}(1 - y_{ij}), \forall i, \forall j \\ \|a(x)\|_* \leq v, \gamma \geq 0, z_j \leq 0, \forall j \\ s_{ij} \geq 0, y_{ij} \in \{0, 1\}, \forall i, \forall j \end{array} \right\} \quad (\text{C.187})$$

where $\forall i$ and $\forall j$ are short for $i = 1, \dots, I$ and $j = 1, \dots, N$, respectively;

$$M_{ij} \geq \max_{x \in Z_1} |b_i(x) - a(x)^T \zeta_i^j|$$

It is easy to see that if $b_i(x) - a(x)^T \zeta_i^j < 0$, then $y_{ij} = 0$ (otherwise $s_{ij} \leq b_i(x) - a(x)^T \zeta_i^j < 0$), hence $s_{ij} = 0 = \max\{b_i(x) - a(x)^T \zeta_i^j, 0\}$. If $b_i(x) - a(x)^T \zeta_i^j > 0$, then $y_{ij} = 1$ (otherwise $b_i(x) - a(x)^T \zeta_i^j \leq M_{ij} y_{ij} = 0$), hence $s_{ij} = b_i(x) - a(x)^T \zeta_i^j = \max\{b_i(x) - a(x)^T \zeta_i^j, 0\}$. If $b_i(x) - a(x)^T \zeta_i^j = 0$, then we have $s_{ij} = 0$ regardless of the value of y_{ij} . In conclusion, (C.185) and (C.187) are equivalent.

For right-hand uncertainty in which $\eta = 0$, $a(x) = \mathbf{1}$, $X = Z_1$ because $Z_2 = \emptyset$. Moreover, variable v in (C.187) is equal to 1 if 1-norm is used in Wasserstein ambiguity set D_W , indicating $v \geq \|\mathbf{1}\|_\infty = 1$ in Z_1 .

In (C.187), a total number of $I \times N$ binary variables are introduced to linearize the $\max\{a, b\}$ function, making the problem challenging to solve. An inner approximation of Z is to simply replace $\max\{b_i(x) - a(x)^T \zeta_i^j, 0\}$ with its first input, yielding a parameter-free approximation

$$Z = \left\{ x \in \mathbb{R}^n \mid \begin{array}{l} \epsilon v - \beta \gamma \leq \frac{1}{N} \sum_{j=1}^N z_j \\ z_j + \gamma \leq b_i(x) - a(x)^T \zeta_i^j, \forall i, \forall j \\ z_j \leq 0, \forall j, \|a(x)\|_* \leq v, \gamma \geq 0 \end{array} \right\} \quad (\text{C.188})$$

This formulation can be derived from CVaR model, and enjoys better computational tractability.

4. Adaptive robust chance constraints

Robust chance constraint program with Wasserstein metric is studied in [82] in a different but more general form. The problem is as follows

$$\begin{aligned}
& \min_{x \in X} c^T x \\
& \text{s.t.} \quad \inf_{\mathbb{Q} \in D_W} \Pr[F(x, \xi) \leq 0] \geq 1 - \beta
\end{aligned} \tag{C.189}$$

where X is a bounded polyhedron, $F : \mathbb{R}^n \times \Xi \rightarrow \mathbb{R}$ is a scalar function that is convex in x for every ξ . This formulation is general enough to capture joint chance constraints. To see this, suppose F contains K individual constraints, then F can be defined as the component-wise maximum as in (C.91).

Here we develop a technique to solve two-stage problems where $F(x, \xi)$ is the optimal value of another LP parameterized in x and ξ . More precisely, we consider

$$\begin{aligned}
& \min_{x \in X} c_1^T x \\
& \text{s.t.} \quad \sup_{\mathbb{Q} \in D_W} \Pr[f(x, \xi) \geq c_2^T \xi] \leq \beta
\end{aligned} \tag{C.190a}$$

$$\begin{aligned}
& f(x, \xi) = \min_y c_3^T y \\
& \text{s.t.} \quad Ax + By + C\xi \leq d
\end{aligned} \tag{C.190b}$$

where in (C.190a), the robust chance constraint can be regarded a risk limiting requirement, and the threshold value depends on uncertain parameter ξ . We assume LP (C.190b) is always feasible (relatively complete recourse) and has finite optimum. Second-stage cost can be considered in the objective function of (C.190a) in form of worst-case expectation which has been discussed in previous sections and is omitted here for the sake of brevity. Here we focus on coping with second-stage LP in robust chance constraint.

Define loss function

$$g(x, \xi) = f(x, \xi) - c_2^T \xi \tag{C.191}$$

Recall the relation between chance constraint and CVaR discussed in Sect. C.3.1, a sufficient condition of robust chance constraint in (C.190a) is $\text{CVaR}(g(x, \xi), \beta) \leq 0$, $\forall \mathbb{Q} \in D_W$, or equivalently

$$\sup_{\mathbb{Q} \in D_W} \inf_{\gamma \in \mathbb{R}} \beta\gamma + \mathbb{E}_{\mathbb{Q}}(\max\{g(x, \xi) - \gamma, 0\}) \leq 0 \tag{C.192}$$

According to [82], constraint (C.192) can be conservatively approximated by

$$\epsilon L + \inf_{\gamma \in \mathbb{R}} \left\{ \beta\gamma + \frac{1}{N} \sum_{i=1}^N \max\{g(x, \xi^i) - \gamma, 0\} \right\} \leq 0 \tag{C.193}$$

where ϵ is the parameter in Wasserstein ambiguity set D_W , L is a constant satisfying $g(x, \xi) \leq L\|\xi\|_1$, and ξ^i , $i = 1, \dots, N$ are samples of uncertain data. Substituting (C.190b) and (C.191) into (C.193), we obtain an LP that is equivalent to problem (C.190)

$$\begin{aligned}
& \min \quad c_1^T x \\
& \text{s.t.} \quad x \in X, \quad \epsilon L + \beta \gamma + \frac{1}{N} \sum_{i=1}^N s_i \leq 0 \\
& \quad \quad s_i \geq 0, \quad s_i \geq c_3^T y^i - c_2^T \xi^i - \gamma, \quad i = 1, \dots, N \\
& \quad \quad Ax + By^i + C\xi^i \leq d, \quad i = 1, \dots, N
\end{aligned} \tag{C.194}$$

where y^i is the second-stage decision associated with ξ^i . This formulation could be very conservative due to three reasons. First, worst-case distribution is considered; second, CVaR constraint (C.192) is a pessimistic approximation of chance constraints; finally, sampling constraint (C.193) is a pessimistic approximation of (C.192).

More discussions on robust chance constraints with Wasserstein metric under various settings can be found in [82].

5. Use of forecast data

Wasserstein metric enjoys many advantages, such as finite-sample performance guarantee and existence of tractable reformulation. However, moment information is not used, especially the first-order moment reflecting the prediction, which can be updated with time rolling on, so the worst-case distribution generally has a mean value different from the forecast (if available). To incorporate forecast data, we propose the following Wasserstein ambiguity set with fixed-mean

$$D_W^M = \left\{ \mathbb{Q} \in D_W \mid \mathbb{E}_{\mathbb{Q}}[\xi] = \hat{\xi} \right\} \tag{C.195}$$

and the worst-case expectation problem can be expressed as

$$\sup_{\mathbb{Q} \in D_W^M} \mathbb{E}_{\mathbb{Q}}[l(\xi)] \tag{C.196a}$$

$$= \sup_{f^n(\xi)} \frac{1}{N} \sum_{n=1}^N \int_{\Xi} l(\xi) f^n(\xi) d\xi \tag{C.196b}$$

$$\text{s.t.} \quad \frac{1}{N} \sum_{n=1}^N \int_{\Xi} \|\xi - \xi^n\|_p f^n(\xi) d\xi \leq \epsilon : \lambda \tag{C.196c}$$

$$\int_{\Xi} f^n(\xi) d\xi = 1 : \theta_n, \quad n = 1, \dots, N \tag{C.196d}$$

$$\frac{1}{N} \sum_{n=1}^N \int_{\Xi} \xi f^n(\xi) d\xi = \hat{\xi} : \rho \tag{C.196e}$$

where $l(\xi)$ is a loss function similar to that in (C.175), $f^n(\xi)$ is the conditional density function under historical data sample ξ^n , dual variables λ , θ_n , and ρ are listed following a colon. Similar to the discussions for problem (C.79),

the dual problem of (C.196) is

$$\min_{\lambda \geq 0, \theta_n, \rho} (\lambda \epsilon + \rho^T \hat{\xi})N + \sum_{n=1}^N \theta_n \quad (\text{C.197a})$$

$$\text{s.t. } \theta_n + \lambda \|\xi - \xi^n\|_p + \rho^T \xi \geq l(\xi), \forall \xi \in \Xi, \forall n \quad (\text{C.197b})$$

For $p = 2$, polyhedral Ξ and PWL $l(\xi)$, constraint (C.197b) can be transformed into the intersection of PSD cones, and problem (C.197) gives rise to an SDP; some examples can be found in Sect. C.3.1. If Ξ is described by a single quadratic constraint, constraint (C.197b) can be reformulated by using the well-known S-Lemma, which has been discussed in Sect. A.2.4, and problem (C.197) still comes down to an SDP. For $p = 1$ or $p = +\infty$, polyhedral Ξ and PWL $l(\xi)$, constraint (C.197b) can be transformed into a polyhedron using duality theory, and problem (C.197) gives rise to an LP. Because the ambiguity set is more restrictive, problem (C.197) would be less conservative than problem (C.178) in which the mean value of uncertain data is free.

A Wasserstein-moment metric with variance is exploited in [83] and applied to wind power dispatch. Nevertheless, the ambiguity set neglects first-order moment and considers second-order moment. This formulation is useful when little historical data is available at hand.

As a short conclusion, distributionally robust optimization and data-driven robust stochastic optimization leverage statistical information on the uncertain data and overcome the conservatism of traditional robust optimization approaches which are built upon the worst-case scenario. The core issue is the equivalent convex reformulation of the worst-case expectation problem or the robust chance constraint over the uncertain probability distribution restricted in the ambiguity set. Optimization over a moment based ambiguity set can be formulated as a semi-infinite LP, whose dual problem gives rise to SDPs, and hence can be readily solved. When additional structure property is taken into account, such as unimodality, more sophisticated treatment is need. As for the robust stochastic programming, tractable reformulation of the worst-case expectation and robust chance constraints is the central issue. Robust chance constraint under a ϕ -divergence based ambiguity set are equivalent to traditional chance constraint under the empirical distribution but with a modified confidence level, and it can be transformed into an MILP or approximated by LP based on risk theory under the help of sampling average approximation technique, so does a robust chance constraint under a Wasserstein metric based ambiguity set, following somewhat different expressions. The worst-case expectation under ϕ -divergence based ambiguity set boils down to a convex program with linear constraints and a nonlinear objective function, which can be efficiently solved via outer approximation algorithm. The worst-case expectation under Wasserstein ambiguity set comes down to a conic program which is convex and readily solvable. Unlike the max-min

problem in traditional robust optimization method identifying the worst-case scenario which the decision maker wishes to avoid, the worst-case expectation problem in distributionally robust optimization and robust stochastic programming is solved in its dual form, whose solution is less intuitive to the decision maker; moreover, it may not be easy to recover the primal optimal solution, i.e., the worst-case probability. The worst-case distribution in the robust chance constrained stochastic programming is discussed in [49, 72]; the worst-case discrete distribution in a two-stage stochastic program with min-max expectation can be computed via a polynomial complexity algorithm. Nonetheless, from a practical perspective, what the human decision makers actually need to deploy is merely the here-and-now decision, and the worst probability distribution is usually not very important, since corrective actions can be postponed to a later stage when the uncertain data have been observed or can be predicted with high accuracy.

C.5 Further Reading

Uncertainty is ubiquitous in real-life decision-making problems, and the decision maker usually has limited information and statistic data on the uncertain factors, which makes robust optimization very attractive in practice, as it is tailored to the available information at hand, and often gives rise to computationally tractable reformulations. Although the original idea can date back to [63] in 1970s, it is during the past two decades that the fundamental theory of robust optimization has been systematically developed. This research field is even more active during the past five years. This chapter aims to help beginners get an overview on this method and understand how to apply robust optimization in practice. We provide basic models and tractable reformulations, called the robust counterparts, for various robust optimization models under different assumptions on the uncertainty and decision-making manner. Basic theory of robust optimization is provided in [1, 3]. Comprehensive surveys can be found in [4, 6]. Here we shed more light on several important topics in robust optimization.

Uncertainty sets play a decisive role on the performance of a robust solution. A larger set could protect the system against a higher level of uncertainty, and increase the cost as well. However, the probability that uncertain data take their worst-case values is usually small. The decision-maker needs to make a trade-off between reliability and economy. Ambiguous chance constraints and their approximations are discussed in Chapter 2 of [1], based on which the parameter in the uncertainty set can be selected. It is proposed in [64] to construct uncertainty sets from historical data and statistical tests. The connection of uncertainty sets and coherent risk measures are revealed in [65]. It is shown that the distortion risk measure leads to a polyhedral uncertainty set. Specifically, the connection of CVaR and uncertainty sets is

discussed in [66]. A reverse correspondence is reported in [67], demonstrating that robust optimization could generalize the concepts of risk measures. A data-driven approach is proposed in [64] to construct uncertainty sets for robust optimization based on statistical hypothesis tests. The counterpart problems are shown to be tractable, and optimal solutions satisfy constraints with finite-sample probabilistic guarantee.

Distributionally robust optimization integrates statistic information, worst-case expectation, and robust probability guarantee in a holistic optimization framework, in which the uncertainty is modeled via an ambiguous probability distribution. The choice of ambiguity sets for candidate distributions affects not only the model conservatism, but also the existence of tractable reformulations. Various ambiguity sets have been proposed in the literature, which can be roughly classified into two categories:

1) Moment ambiguity sets. All PDFs share the same moment data, usually the first- and second-order moments, and structured properties, such as symmetry and unimodality. For example, Markov ambiguity set contains all distributions with the same mean and support, and the worst-case expectation is shown to be equivalent to LPs [68]. Chebyshev ambiguity set is composed of all distributions with known expectation and covariance matrix, and usually leads to SDP counterparts [38, 69, 70]; the Gauss ambiguity set contains all unimodal distributions in the Chebyshev ambiguity set, and also gives rise to SDP reformulations [71].

2) Divergence ambiguity sets. All PDFs are close to a reference distribution in term of a specified measure. For example, the Wasserstein ambiguity quantifies the divergence via Wasserstein metric [72, 74, 75]; the ϕ -divergence ambiguity [58, 76] characterizes the divergence of two probability density functions through the distance of special non-negative weights (for discrete distributions) or integrals (for continuous distributions).

More information on the types of ambiguity sets and reformulations of their distributionally robust counterparts can be found in [57]. According to the latest research progress, the moment based distributionally robust optimization is relatively mature and has been widely adopted in engineering, because the semi-infinite LP formulation and its dual for the worst-case expectation problem offer a systematic approach to analyze the impact of uncertain distributions. However, when more complicated ambiguity sets are involved, such as the Gauss ambiguity set, deriving a tractable reformulation needs more sophisticated approaches. The study on the latter category, which directly imposes uncertainty on the distributions is attracting growing attentions in the past two or three years, because it makes full use of historical data, which can better capture the unique feature of uncertain factors under investigation.

Data-driven robust stochastic programming, conceptually the same as distributionally robust optimization but preferred by some researchers, has been studied using ϕ -divergence in [77, 78], and Wasserstein metric in

[72, 74, 75, 79, 80, 81, 82], because a tractable counterpart problem can be derived under such ambiguity sets.

Many decision-making problems in engineering and finance often require that a certain risk measure associated with random variables should be limited below a threshold. However, the probability distribution of random variables is not exactly known; therefore, the risk limiting constraint must be able to withstand perturbations of distribution in a reasonable range. This entails a tractable reformulation of a risk measure under distributional uncertainty. This problem has been comprehensively discussed in [75]. In more recent publications, CVaR under moment ambiguity set with unimodality is studied in [84]; VaR and CVaR under moment ambiguity set are discussed in [85]; distortion risk measure under Wasserstein ambiguity set is considered in [86].

In multi-stage decision making, causality is a pivotal issue for practical implementation, which means that the wait-and-see decisions in the current stage cannot depend on the information of uncertainty in future stages. For example, in a unit commitment problem with 24 periods, the wind power output is observed period-by-period. It is shown in [87] that the two-stage robust model in [29] offers non-causal dispatch strategies, which are in fact not robust. A multi-stage causal unit commitment model is suggested in [87, 88] based on affine policy. Causality is put to effect by imposing block diagonal constraints on the gain matrix of affine policy. Causality is also called non-anticipativity in some literature, such as [89], which is attracting attention from practitioners [90, 91].

For some other interesting topics on robust optimization, such as the connection with stochastic optimization, connection with risk theory, and applications in engineering problems other than those in power systems, readers can refer to [6]. Nonlinear issues have been addressed in [11, 92]. Optimization models with uncertain SOC and SDP constraints are discussed in [93, 94]. The connection among robust optimization, data utilization, and machine learning has been reviewed in [95].

References

1. Ben-Tal, A., El Ghaoui, L., Nemirovski, A.: Robust Optimization. Princeton University Press (2009)
2. Bertsimas, D., Sim, M.: The price of robustness. *Oper. Res.* 52(1): 35-53 (2004)
3. Bertsimas, D., Brown, D.B., Caramanis, C.: Theory and applications of robust optimization. *SIAM Rev.* 53(3):464-501 (2011)
4. Gorissen, B.L., Yanıkoğlu, İ., den Hertog, D.: A practical guide to robust optimization. *Omega* 53: 124-137 (2015)
5. Ben-Tal, A., Nemirovski, A.: Selected topics in robust convex optimization. *Math. Program.* 112(1):125-158 (2008)
6. Gabrel, V., Murat, C., Thiele, A.: Recent advances in robust optimization: An overview. *Eur. J. Oper. Res.* 235(3): 471-483 (2014)

7. Yanıkoğlu, İ., Gorissen, B. L., den Hertog, D.: A survey of adjustable robust optimization. *Eur. J. Oper. Res.* (2018)
8. Boyd, S., Vandenberghe, L.: *Convex Optimization*. Cambridge University Press, New York, NY (2004)
9. Alizadeh, F., Goldfarb, D.: Second-order cone programming. *Math. Program.* 95(1): 3-51 (2003)
10. Ben-Tal, A., Nemirovski, A.: *Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering applications*. SIAM, Philadelphia, PA, USA (2001) available online: http://www2.isye.gatech.edu/~nemirovs/Lect_ModConvOpt.pdf
11. Ben-Tal, A., den Hertog, D., Vial, J-Ph. Deriving robust counterparts of nonlinear uncertain inequalities. *Math. Program.* 149(1): 265-299 (2015)
12. Bandi, C., Bertsimas, D.: Tractable stochastic analysis in high dimensions via robust optimization. *Math. Program.* 134(1): 23-70 (2012)
13. Bertsimas, Dimitris, Iain Dunning, and Miles Lubin. Reformulation versus cutting-planes for robust optimization. *Comput. Manag. Sci.* 13(2): 195-217 (2016)
14. Iancu, D.A., Trichakis, N.: Pareto efficiency in robust optimization. *Manage. Sci.* 60(1): 130-147 (2013)
15. Wei, W., Liu, F., Mei, S.: Offering non-dominated strategies under uncertain market prices. *IEEE Trans. Power Syst.* 30(5): 2820-2821 (2015)
16. Wei, W., Liu, F., Mei, S.: Energy pricing and dispatch for smart grid retailers under demand response and market price uncertainty. *IEEE Trans. Smart Grid* 6(3): 1364-1374 (2015)
17. Bertsimas, D., Goyal, V.: On the power of robust solutions in two-stage stochastic and adaptive optimization problems. *Math. Oper. Res.* 35(2): 284-305 (2010)
18. Chen, X., Sim, M., Sun, P.: A robust optimization perspective on stochastic programming. *Oper. Res.* 55(6): 1058-1071 (2007)
19. Ben-Tal, A., Goryashko, A., Guslitzer, E., Nemirovski, A.: Adjustable robust solutions of uncertain linear programs. *Math. Program.* 99(2): 351-376 (2004)
20. Bertsimas, D., Iancu, D.A., Parrilo, P.A.: Optimality of affine policies in multi-stage robust optimization. *Math. Oper. Res.* 35(2): 363-394 (2010)
21. Bertsimas, D., Goyal, V.: On the power and limitations of affine policies in two-stage adaptive optimization. *Math. Program.* 134(2): 491-531 (2012)
22. Bertsimas, D., Bidkhori, H.: On the performance of affine policies for two-stage adaptive optimization: a geometric perspective. *Math. Program.* 153(2): 577-594 (2015)
23. Chen, X., Zhang, Y.: Uncertain linear programs: Extended affinely adjustable robust counterparts. *Oper. Res.* 57(6): 1469-1482 (2009)
24. Bertsimas, D., Caramanis, C.: Finite adaptability in multistage linear optimization. *IEEE Trans. Automat. Contr.* 55(12): 2751-2766 (2010)
25. Bertsimas, D., Goyal, V., Sun, X.A.: A geometric characterization of the power of finite adaptability in multistage stochastic and adaptive optimization. *Math. Oper. Res.* 36(1): 24-54 (2011)
26. Chen, X., Sim, M., Sun, P., Zhang, J.: A linear decision-based approximation approach to stochastic programming. *Oper. Res.* 56(2): 344-357 (2008)
27. Bertsimas, D., Georghiou, A.: Design of near optimal decision rules in multistage adaptive mixed-integer optimization. *Oper. Res.* 63(3): 610-627 (2015)
28. Georghiou, A., Wiesemann, W., Kuhn, D.: Generalized decision rule approximations for stochastic programming via liftings. *Math. Program.* 152(1): 301-338 (2015)
29. Bertsimas, D., Litvinov, E., Sun, X.A., Zhao, J., Zheng, T.: Adaptive robust optimization for the security constrained unit commitment problem. *IEEE Trans. Power Syst.* 28(1): 52-63 (2013)

30. Falk, J.E.: A linear max-min problem. *Math. Program.* 5(1): 169-188 (1973)
31. Konno, H.: A cutting plane algorithm for solving bilinear programs. *Math. Program.*, 11(1): 14-27 (1976)
32. Korsi, J., Pfeuffer, F., Klamroth, K.: Biconvex sets and optimization with biconvex functions: a survey and extensions. *Math. Methods Oper. Res.* 66(3), 373-407 (2007)
33. Zeng, B., Zhao, L.: Solving two-stage robust optimization problems using a column-and-constraint generation method. *Oper. Res. Lett.* 41(5): 457-461 (2013)
34. Bertsimas, D., de Ruiter, F.: Duality in two-stage adaptive linear optimization: Faster computation and stronger bounds. *INFORMS J. Comput.* 28(3): 500-511 (2016)
35. Bertsimas, D., Shtern, S.: A scalable algorithm for two-stage adaptive linear optimization. *arXiv preprint arXiv:1807.02812* (2018)
36. Zhao, L., Zeng, B.: An exact algorithm for two-stage robust optimization with mixed integer recourse problems. Working paper (2012), available online: http://www.optimization-online.org/DB_FILE/2012/01/3310.pdf
37. Bertsimas, D., Thiele, A.: A robust optimization approach to inventory theory. *Oper. Res.* 54(1) 150-168 (2006)
38. Zymler, S., Kuhn, D., Rustem, B.: Distributionally robust joint chance constraints with second-order moment information. *Math. Program.* 137(1): 167-198 (2013)
39. Alizadeh, F., Goldfarb, D.: Second-order cone programming. *Math. Program.* 95(1), 3-51 (2003)
40. Prékopa, A.: On probabilistic constrained programming. In: *Proceedings of the Princeton Symposium on Mathematical Programming*, pp. 113-138. Princeton University Press, Princeton (1970)
41. Shapiro, A.: On duality theory of conic linear problems. In: Goberna, M.A., Lopez, M.A. (eds.) *Semi-Infinite Programming: Recent Advances*, Kluwer, Dordrecht (2001)
42. Isii, K.: The extrema of probability determined by generalized moments (I) bounded random variables. *Ann. Inst. Stat. Math.* 12(2), 119-134 (1960)
43. Bertsimas, D., Popescu, I.: Optimal inequalities in probability theory: A convex optimization approach. *SIAM J. Optimiz.* 15(3): 780-804 (2005)
44. Lasserre, J.B.: *Moments, Positive Polynomials and Their Applications*. Imperial College Press, London (2009)
45. Rockafellar, R.T., Uryasev, S.: Optimization of conditional value-at-risk. *J. Risk* 2: 21-24 (2002)
46. Shapiro, A., Kleywegt, A.J.: Minimax analysis of stochastic problems. *Optim. Method. Softw.* 17(3), 523-542 (2002)
47. Lipp, T., Boyd, S.: Variations and extension of the convex-concave procedure. *Optim. Eng.* 17(2): 263-287 (2016)
48. Bertsimas, D., Doan, X.V., Natarajan, K., Teo, C.P.: Models for minimax stochastic linear optimization problems with risk aversion. *Math. Oper. Res.* 35(3): 580-602 (2010)
49. Jiang, R., Guan, Y.: Data-driven chance constrained stochastic program. *Math. Program.* 158(1): 291-327 (2016)
50. Hu, Z., Hong, L.J.: Kullback-Leibler divergence constrained distributionally robust optimization. Available at http://www.optimization-online.org/DB_FILE/2012/11/3677.pdf
51. Chen, Y., Guo, Q., Sun, H., Li, Z., Wu, W., Li, Z.: A distributionally robust optimization model for unit commitment based on Kullback-Leibler divergence. *IEEE Trans. Power Syst.* 33(5): 5147-5160 (2018)
52. Ahmed, S., Xie, W.: Relaxations and approximations of chance constraints under finite distributions. *Math. Program.* 170(1): 43-65 (2018)

53. Duran, M.A., Grossmann, I.E.: An outer-approximation algorithm for a class of mixed-integer nonlinear programs. *Math. Program.* 36(3): 307-339 (1986)
54. Fletcher, R., Leyffer, S.: Solving mixed integer nonlinear programs by outer approximation. *Math. Program.* 66(1-3): 327-349 (1994)
55. Kelley, J.E.: The cutting-plane method for solving convex programs. *Journal of the society for Industrial and Applied Mathematics*, 8(4): 703-712 (1960)
56. Hogan, W.W.: Applications of a general convergence theory for outer approximation algorithms. *Math. Program.* 5(1): 151-168 (1973)
57. Hanasusanto, G.A., Roitch, V., Kuhn, D., Wiesemann, W.: A distributionally robust perspective on uncertainty quantification and chance constrained programming. *Math. Program.* 151(1): 35-62 (2015)
58. Ben-Tal, A., Den Hertog, D., De Waegenaere, A., Melenberg, B., Rennen, G.: Robust solutions of optimization problems affected by uncertain probabilities. *Manag. Sci.* 59(2), 341-357 (2013)
59. Liese, F., Vajda, I.: On divergences and informations in statistics and information theory. *IEEE Trans. Inform. Theory* 52(10):4394-4412 (2006)
60. Calafiore, G.C., El Ghaoui, L.: On distributionally robust chance-constrained linear programs. *J. Optimiz. Theory App.* 130(1): 1-22 (2006)
61. Zhao, C., Guan, Y.: Data-driven stochastic unit commitment for integrating wind generation. *IEEE Trans. Power Syst.* 31(4): 2587-2596 (2016)
62. Ding, T., Yang, Q., Yang, Y., Li, C., Bie, Z.: A data-driven stochastic reactive power optimization considering uncertainties in active distribution networks and decomposition method. *IEEE Trans. Smart Grid* 9(5): 4994-5004 (2018)
63. Soyster, A.L.: Convex programming with set-inclusive constraints and applications to inexact linear programming. *Oper. Res.* 21(5): 1154-1157 (1973)
64. Bertsimas, D., Gupta, V., Kallus, N.: Data-driven robust optimization. *Math. Program.* 167(2): 235-292 (2018)
65. Bertsimas, D., David, B. Constructing uncertainty sets for robust linear optimization. *Oper. Res.* 57(6) : 1483-1495 (2009)
66. Chen, W., Sim, M., Sun, J., Teo, C.P.: From CVaR to uncertainty set: Implications in joint chance-constrained optimization. *Oper. Res.* 58(2): 470-485 (2010)
67. Natarajan, K., Pachamanova, D., and Sim, M.: Constructing risk measures from uncertainty sets. *Oper. Res.* 57(5) : 1129-1141 (2009)
68. Wiesemann, W., Kuhn, D., Sim, M.: Distributionally robust convex optimization. *Oper. Res.* 62(6): 1358-1376 (2014)
69. Delage, E., Ye, Y.: Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Oper. Res.* 58(3), 596-612 (2010)
70. Vandenberghe, L., Boyd, S., Comanor, K.: Generalized Chebyshev bounds via semidefinite programming. *SIAM Rev.* 49(1), 52-64 (2007)
71. Van Parys, B.P.G., Goulart, P.J., Kuhn, D.: Generalized Gauss inequalities via semidefinite programming. *Math. Program.* 156(1): 271-302, (2016)
72. Esfahani, P.M., Kuhn, D.: Data-driven distributionally robust optimization using the Wasserstein metric: Performance guarantees and tractable reformulations. *Math. Program.* 171(1-2): 115-166 (2018)
73. Kantorovich, L.V., Rubinshtein, G.S.: On a space of totally additive functions. *Vestn. Leningr. Univ.* 13: 52-59 (1958)
74. Xie, W.: On distributionally robust chance constrained program with Wasserstein distance. *arXiv:1806.07418* (2018)
75. Postek, K., den Hertog, D., Melenberg, B.: Computationally tractable counterparts of distributionally robust constraints on risk measures. *SIAM Rev.* 58(4): 603-650 (2016)
76. Shapiro, A.: Distributionally robust stochastic programming. *SIAM J. Optimiz.* 27(4): 2258-2275 (2017)

77. Bayraksan, G., Love, D.K.: Data-driven stochastic programming using phi-divergences. *The Operations Research Revolution*, 1-19, INFORMS, Catonsville, MD (2015)
78. Love, D., Bayraksan, G.: Phi-divergence constrained ambiguous stochastic programs for data-driven optimization. Working paper, Available online at: http://www.optimization-online.org/DB_FILE/2016/03/5350.pdf (2015)
79. Zhao, C., Guan, Y.: Data-driven risk-averse stochastic optimization with Wasserstein metric. *Oper. Res. Lett.* 46(2): 262-267 (2018)
80. Hanasusanto, G.A., Kuhn, D.: Conic programming reformulations of two-stage distributionally robust linear programs over Wasserstein balls. *Oper. Res.* 66(3): 849-869 (2018)
81. Luo, F., Mehrotra, S.: Decomposition algorithm for distributionally robust optimization using Wasserstein metric. *arXiv preprint arXiv:1704.03920* (2017)
82. Hota, A. R., Cherukuri, A., Lygeros, J.: Data-driven chance constrained optimization under Wasserstein ambiguity sets. *arXiv preprint arXiv:1805.06729* (2018)
83. Yao, L., Wang, X., Duan, C., Wu, X., Zhang, W.: Risk-based distributionally robust energy and reserve dispatch with Wasserstein-moment metric. *IEEE Power & Energy Society General Meeting* (2018)
84. Li, B., Jiang, R., Mathieu, J.L.: Ambiguous risk constraints with moment and unimodality information. *Math. Program.* (2019)
85. Lotfi, S., Zenios, S.A.: Equivalence of robust VaR and CVaR optimization. Working paper 16-03, The Wharton Financial Institutions Center (2016)
86. Pichler, A., Xu, H.: Quantitative stability analysis for minimax distributionally robust risk optimization. *Math. Program.* (2019)
87. Lorca, A., Sun, X.A., Litvinov, E., Zheng, T.: Multistage adaptive robust optimization for the unit commitment problem. *Oper. Res.* 64(1): 32-51 (2016)
88. Lorca, A., Sun, X.A.: Multistage robust unit commitment with dynamic uncertainty sets and energy storage. *IEEE Trans. Power Syst.* 32(3): 1678-1688 (2017)
89. Gupta, V., Grossmann, I.E.: Solution strategies for multistage stochastic programming with endogenous uncertainties. *Comput. Chem. Eng.* 35(11): 2235-2247 (2011)
90. Ding, T., Hu, Y., Bie, Z.: Multi-stage stochastic programming with nonanticipativity constraints for expansion of combined power and natural gas systems. *IEEE Trans. Power Syst.* 33(1): 317-328 (2018)
91. Zhai, Q., Li, X., Lei, X., Guan, X.: Transmission constrained UC with wind power: An all-scenario-feasible MILP formulation with strong nonanticipativity. *IEEE Trans. Power Syst.* 32(3): 1805-1817 (2017)
92. Ben-Tal, A., Brekelmans, R., Den Hertog, D., Vial, J.P.: Globalized robust optimization for nonlinear uncertain inequalities. *INFORMS J. Comput.* 29(2): 350-366 (2017)
93. El Ghaoui, L., Oustry, F., Lebret, H.: Robust solutions to uncertain semidefinite programs. *SIAM J. Optimiz.* 9(1): 33-52 (1998)
94. Zhen, J., de Ruiter, F., den Hertog, D.: Robust optimization for models with uncertain SOC and SDP constraints. Available at http://www.optimization-online.org/DB_FILE/2017/12/6371.pdf (2017)
95. Ning, C., You, F.: Optimization under uncertainty in the era of big data and deep learning: When machine learning meets mathematical programming. *Comput. Chem. Eng.* 125(9): 434-448, (2019)

Appendix D

Equilibrium Problems

Life is not a game. Still, in this life, we choose the games we live to play.

The concept of an equilibrium describes a state that the system has no incentive to change. These incentives can be profit-driven in the case of competitive markets or a reflection of physical laws such as energy flow equations. In this sense, equilibrium encompasses broader concepts than the solution of a game. Equilibrium is a fundamental notation appearing in various disciplines in economics and engineering. Identifying the equilibria allows eligible authorities to predict the system state at a future time or design reasonable policies for regulating a system or a market. This is not saying that an equilibrium state must appear sooner or later, partly because decision makers in reality have only limited rationality and information. Nevertheless, the awareness of such an equilibrium could be helpful for system design and operation. In this chapter, we restrict our attention in the field of game theory, which entails simultaneously solving multiple interactive optimization problems. We review the notions of some quintessential equilibrium problems and show how they can be solved via traditional optimization methods. These problems can be roughly categorized into two classes: the first one contains only one level: all players must make a decision simultaneously, which is referred to as a Nash-type game; the second one has two levels: decisions are made sequentially by two groups of players, called the leaders and the followers. This category is widely known as Stackelberg-type games, or multi-leader-follower games, or equilibrium programs with equilibrium constraints (EPEC). Unlike a traditional mathematical programming problem where the decision maker is unique, in an equilibrium problem or a game, multiple decision makers seek optimums of individual optimization problems parameterized in the optimal solutions of others.

General notations used throughout this chapter are defined as follows. Specific symbols are explained in the individual sections. In the game theoretic language, a decision maker is called a player. Vector $x = (x_1, \dots, x_n)$ refers to the joint decisions of all upper-level players or the so-called leaders in a bilevel setting, where x_i stands for the decisions of leader i ; $x_{-i} = (x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n)$ refers to the rivals' actions for leader i .

Similarly, $y = (y_1, \dots, y_m)$ refers to the joint decisions of all lower-level players or the so-called followers, where y_j stands for the decisions of follower j ; $y_{-j} = (y_1, \dots, y_{j-1}, y_{j+1}, \dots, y_m)$ refers to the rivals' actions for follower j . λ and μ are Lagrangian dual multipliers associated with inequality and equality constraints.

D.1 Standard Nash Equilibrium Problem

After J. F. Nash published his work on the equilibrium of n -person non-cooperative games in early 1950s [1, 2], game theory quickly became a new branch of operational research. Nash equilibrium problem (NEP) captures the interactive behaviors of strategic players, in which each player's utility depends on the actions of other players. During decades of wonderful research, a variety of new concepts and algorithms of Nash equilibriums have been proposed and applied to almost every area of knowledge. This section just reviews some basic concepts and the most prevalent best-response algorithms.

D.1.1 Formulation and Optimality Condition

In a standard n -person non-cooperative game, each player minimizes his payoff function $f_i(x_i, x_{-i})$ which depends on all players' actions. The strategy set $X_i = \{x_i \in \mathbb{R}^{k_i} \mid g_i(x_i) \leq 0\}$ of player i is independent of x_{-i} . The joint strategy set of the game is the Cartesian product of X_i , i.e., $X = \prod_{i=1}^n X_i$, and $X_{-i} = \prod_{j \neq i} X_j$. Roughly speaking, the non-cooperative game is a collection of coupled optimization problems, where player i chooses $x_i \in X_i$ that minimizes his payoff $f_i(x_i, x_{-i})$ given his rivals' strategies x_{-i} , or mathematically

$$\left. \begin{array}{ll} \min_{x_i} & f_i(x_i, x_{-i}) \\ \text{s.t.} & g_i(x_i) \leq 0 : \lambda_i \end{array} \right\}, \quad i = 1, \dots, n \quad (\text{D.1})$$

In the problem of player i , the decision variable is x_i , and x_{-i} is regarded as parameters; λ_i is the dual variable.

The Nash equilibrium consists of a strategy profile such that every player's strategy constitutes the best response to all other players' strategies, or in other words, no player can further reduce his payoff by changing his action unilaterally. Therefore, the Nash equilibrium is a stable state which can sustain spontaneously. The mathematical definition is formally given below.

Definition D.1. A strategy vector $x^* \in X$ is a Nash equilibrium if the condition

$$f_i(x_i^*, x_{-i}^*) \leq f_i(x_i, x_{-i}^*), \quad \forall x_i \in X_i \quad (\text{D.2})$$

holds for all players.

Condition (D.2) naturally interprets the fact that at a Nash equilibrium, if any player choose an alternative strategy, his payoff may grow, which is undesired. To depict a Nash equilibrium, a usual approach is the fixed-point of best-response mapping. Let $B_i(x_{-i})$ be the set of optimal strategies of player i given the strategies x_{-i} of others, then set $B(x) = \prod_{i=1}^n B_i(x_{-i})$ is the best-response mapping of the game. It is clear that x^* is a Nash equilibrium if and only if $x^* \in B(x^*)$, i.e., x^* is a fixed point of $B(x)$. This fact establishes the foundation for analyzing Nash equilibria using the well-developed fixed-point theory. However, conducting the fixed-point analysis usually requires the best-response mapping $B(x)$ in a closed form. Moreover, to declare the existence and uniqueness of a Nash equilibrium, the mapping should be contractive [3]. These strong assumptions inevitably limit the applicability of fixed-point method. For example, in many instances, the best-response mapping $B(x)$ is neither contractive nor continuous, but Nash equilibria may still exist.

Another way to characterize the Nash equilibrium is the KKT system approach. Generally speaking, in a standard Nash game, each player is facing an NLP parameterized in the rivals' strategies. If we consolidate the KKT optimality conditions of all these NLPs in (D.1), we get the following KKT system

$$\left. \begin{aligned} \nabla_{x_i} f_i(x_i, x_{-i}) + \lambda_i^T \nabla_{x_i} g_i(x_i) &= 0 \\ \lambda_i &\geq 0, \quad g_i(x_i) \leq 0, \quad \lambda_i^T g_i(x_i) &= 0 \end{aligned} \right\} \quad i = 1, \dots, n \quad (\text{D.3})$$

If x^* is a Nash equilibrium that satisfies (D.2), and any standard constraint qualification holds for every player's problem in (D.1), then x^* must be a stationary point of the concentrated KKT system (D.3) [4]; and vice versa: if all problems in (D.1) meet a standard constraint qualification, and a point x^* together with a proper vector of dual multipliers $\lambda = (\lambda_1, \dots, \lambda_n)$ solves KKT system (D.3), then x^* is also a Nash equilibrium that satisfies (D.2).

Problem (D.3) is an NCP and is the optimality condition of Nash equilibrium. It is a natural attempt to retrieve an equilibrium by solving NCP (D.3) without deploying an iterative algorithm, which may suffer from divergence. To obviate the computational challenges brought by the complementarity and slackness constraints in KKT system (D.3), a merit function approach and an interior-point method are comprehensively discussed in [4].

D.1.2 Variational Inequality Formulation

An alternative perspective to study the NEP is to formulate it as a variational inequality (VI) problem. This approach is pursued in [5]. The advantage of variational inequality approach is that it permits an easy access to existence and uniqueness results without the best-response mapping. From a compu-

tational point of view, it naturally leads to easily implementable algorithms along with provable convergence performances.

Given a closed and convex set $X \in \mathbb{R}^n$ and a mapping $F : X \rightarrow \mathbb{R}^n$, a variational inequality problem, denoted by $\text{VI}(X, F)$, is to determine a point $x^* \in X$ satisfying [6]

$$(x - x^*)^T F(x^*) \geq 0, \forall x \in X \quad (\text{D.4})$$

To see the connection between a VI problem and a traditional convex optimization problem that seeks a minimum of a convex function $f(x)$ over a convex set X , let us assume that the optimal solution is x^* , then the feasible region must not lie in the half space where $f(x)$ decreases; geometrically, the line segment connecting any $x \in X$ with x^* must form an acute angle with the gradient of f at x^* , which can be mathematically described as $(x - x^*)^T \nabla f(x^*) \geq 0, \forall x \in X$. This condition can be concisely expressed by $\text{VI}(X, \nabla f)$ [7].

However, when the Jacobian matrix of F is not symmetric, F cannot be written as the gradient of another scalar function, and hence the variational inequality problem encompasses broader classes of problems than traditional mathematical programs. For example, when $X = \mathbb{R}^n$, problem (D.4) degenerates into a system of equations $F(x^*) = 0$; when $X = \mathbb{R}_+^n$, problem (D.4) comes down to an NCP $0 \leq x^* \perp F(x^*) \geq 0$.

To see the later case, $x^* \geq 0$ because it belongs to X ; if any element of $F(x^*)$ is negative, say the first element $[F(x^*)]_1 < 0$, we let $x_1 = x_1^* + 1$, and $x_i = x_i^*, i = 2, \dots$, then $(x - x^*)^T F(x^*) = [F(x^*)]_1 < 0$, which is contradictive to (D.4). Hence $F(x^*) \geq 0$ must hold. Let $x = 0$ in (D.4), we have $(x^*)^T F(x^*) \leq 0$. Because $x^* \geq 0$ and $F(x^*) \geq 0$, there must be $(x^*)^T F(x^*) = 0$, resulting in the target NCP.

The monotonicity of F plays a central role in the theoretical analysis of VI problems, just like the role of convexity in mathematical programming. It has a close relationship with the Jacobian matrix ∇F [5, 8]:

$F(x)$ is monotone on X	\Leftrightarrow	$\nabla F(x) \succeq 0, \forall x \in X$
$F(x)$ is strictly monotone on X	\Leftarrow	$\nabla F(x) \succ 0, \forall x \in X$
$F(x)$ is strongly monotone on X	\Leftrightarrow	$\nabla F(x) - c_m I \succeq 0, \forall x \in X$

where c_m is a strictly positive constant. As a correspondence to convexity, a differentiable function f is convex (strictly convex, strongly convex) on X if and only if ∇f is monotone (strictly monotone, strongly monotone) on X .

Conceptually, monotonicity (convexity) is the weakest, since the matrix $\nabla F(x)$ can have zero eigenvalues; strict monotonicity (strict convexity) is stronger, as all eigenvalues of matrix $\nabla F(x)$ are strictly positive; strong

monotonicity (strong convexity) is the strongest, because the smallest eigenvalue of matrix $\nabla F(x)$ should be greater than a given positive number. Intuitively, a strong convex function must be more convex than a given convex quadratic function; for example, $f(x) = x^2$ is strongly convex on \mathbb{R} ; $f(x) = 1/x$ is convex on \mathbb{R}_{++} and strongly convex on $(0, 1]$.

To formulate an NEP as a VI problem and establish the existence and uniqueness result of Nash equilibria, we list some assumptions on the convexity and smoothness of each player's problem.

Assumption D.1 [5]

- 1) The strategy set X_i is non-empty, closed and convex;
- 2) Function $f_i(x_i, x_{-i})$ is convex in $x_i \in X_i$ for fixed $x_{-i} \in X_{-i}$;
- 3) Function $f_i(x_i, x_{-i})$ is continuously differentiable in $x_i \in X_i$ for fixed $x_{-i} \in X_{-i}$;
- 4) Function $f_i(x_i, x_{-i})$ is twice continuously differentiable in $x \in X$ with bounded second derivatives.

Proposition D.1. [5] *In a standard NEP $NE(X, f)$, where $f = (f_1, \dots, f_n)$, if conditions 1)-3) in Assumption D.1 are met, then the game is equivalent to a variational inequality problem $VI(X, F)$ with*

$$X = X_1 \times \dots \times X_n$$

and

$$F(x) = (\nabla_{x_1} f_1(x), \dots, \nabla_{x_n} f_n(x))$$

In the VI problem corresponding to a traditional mathematical program, the Jacobian matrix ∇F is symmetric, because it is the Hessian matrix of a scalar function. However, in Proposition D.1, the Jacobian matrix ∇F for an NEP is generally non-symmetric. Building upon the VI reformulation, the standard results on solution properties of VI problems [6] can be extended to standard NEPs.

Proposition D.2. *Given an NEP $NE(X, f)$, all conditions in Assumption D.1 are met, then we have the following statements:*

- 1) *If $F(x)$ is strictly monotone, then the game has at most one Nash equilibrium.*
- 2) *If $F(x)$ is strongly monotone, then the game has a unique Nash equilibrium.*

Some sufficient guarantees for $F(x)$ to be (strictly, strongly) monotone are given in [5]. It should be pointed out that the equilibrium concept in the sense of Proposition D.2 is termed the pure-strategy Nash equilibrium, so as to distinguish it from the mixed-strategy Nash equilibrium which will appear later on.

D.1.3 Best Response Algorithms

A major benefit of the VI reformulation is that it leads to easily implementable solution algorithms. Here we list two of them. Readers who are interested in the proofs on their performances can consult [5].

1. Algorithms for strongly convex cases

The first algorithm is a totally asynchronous-iterative one, in which players may update their strategies with different frequencies. Let $T = \{0, 1, 2, \dots\}$ be the indices of iteration steps, and $T_i \subseteq T$ be the set of steps in which player i updates his own strategy x_i . The notation x_i^k implies that at step $k \notin T_i$, x_i^k remains unchanged. Let $t_j^i(k)$ be the latest step at which the strategy of player j is received by player i at step k . Therefore, if player i updates his strategy at step k , he uses the following strategy profile offered by other players:

$$x_{-i}^{t^i(k)} = \left(x_1^{t_1^i(k)}, \dots, x_{i-1}^{t_{i-1}^i(k)}, x_{i+1}^{t_{i+1}^i(k)}, \dots, x_n^{t_n^i(k)} \right) \quad (\text{D.5})$$

Using above definitions, the totally asynchronous-iterative algorithm is summarized in Algorithm D.1. Some technique conditions for which the schedules T_i and $t_j^i(k)$ should satisfy in order to be implementable in practice are discussed in [9, 10], which are assumed to be satisfied without particular mention.

Algorithm D.1 : Asynchronous best-response algorithm

- 1: Choose a convergence tolerance $\varepsilon > 0$ and a feasible initial point $x^0 \in X$; the iteration index is $k = 0$;
- 2: For player $i = 1, \dots, n$, update the strategy x_i^{k+1} as

$$x_i^{k+1} = \begin{cases} x_i^* \in \arg \min_{x_i} \left\{ f_i \left(x_i, x_{-i}^{t^i(k)} \right) \mid x_i \in X_i \right\} & \text{if } k \in T_i \\ x_i^n & \text{otherwise} \end{cases} \quad (\text{D.6})$$

- 3: If $\|x^{k+1} - x^k\|_2 \leq \varepsilon$, terminate and report x^{k+1} as the Nash equilibrium; otherwise, update $k \leftarrow k + 1$, and go to step 2.
-

A sufficient condition which guarantees the convergence of Algorithm D.1 is provided in [5]. Roughly speaking, Algorithm D.1 would converge if $f_i(x)$ is strongly convex in x_i . However, this is a strong assumption, which cannot be satisfied even if there is only one point where the partial Hessian matrix $\nabla_{x_i}^2 f_i(x)$ of player i is singular.

Algorithm D.1 reduces to some classic algorithms by enforcing a special updating procedure, i.e., a particular selection of T_i and $t_j^i(k)$. For example, if

players update their strategies simultaneously (sequentially), Algorithm D.1 becomes the Jacobi (Gauss-Seidel) type iterative scheme. Interestingly, the asynchronous best-response algorithm is robust against data missing or delay, and is guaranteed to find the unique Nash equilibrium. This feature greatly relaxes the requirement on data synchronization and simplifies the design of communication systems, and makes this class of algorithm very appealing in distributed system operations.

2. Algorithms for convex cases

To relax the strong monotonicity assumption on $F(x)$, the second algorithm has been proposed in [5], which only uses the monotonicity property and is summarized below. Algorithm D.2 converges to a Nash equilibrium, if each player's optimization problem is convex (or $F(x)$ is monotone), which significantly improves its applicability.

Algorithm D.2 : Proximal Decomposition Algorithm

- 1: Given $\{\rho_n\}_{n=0}^\infty$, $\varepsilon > 0$, and $\tau > 0$, choose a feasible initial point $x^0 \in X$;
- 2: Find an equilibrium z^0 of the following NEP using Algorithm D.1

$$\left. \begin{array}{ll} \min_{x_i} & f_i(x_i, x_{-i}) + \tau \|x_i - x_i^0\|_2^2 \\ \text{s.t.} & x_i \in X_i \end{array} \right\}, i = 1, \dots, n \quad (\text{D.7})$$

- 3: If $\|z^0 - x^0\|_2 \leq \varepsilon$, terminate and report x^0 as the Nash equilibrium; otherwise, update $x^0 \leftarrow (1 - \rho_n)x^0 + \rho_n z^0$, and go to step 2.
-

Algorithm D.2 is a double-loop method: the inner loop identifies a Nash equilibrium of the regularized game (D.7) with x^0 being a parameter, which is updated in each iteration, and the outer loop updates x^0 by selecting a new point along the line connecting x^0 and z^0 . Notice that in step 2, as long as τ is large enough, the Hessian matrix $\nabla_{x_i}^2 f_i(x) + 2\tau I$ must be positive definite, and thus the best-response algorithm applied to (D.7) is guaranteed to converge to the unique Nash equilibrium. See [5] for more details about parameter selection.

The penalty term $\tau \|x_i - x_i^0\|_2^2$ limits the change of optimal strategies in two consecutive iterations, and can be interpreted as a damping factor that attenuates possible oscillations during the computation. It is worth mentioning that the penalty parameter τ significantly impacts the convergence rate of Algorithm D.2 and should be carefully selected. If it is too small, the damping effect of the penalty term is limited, and the oscillation may still take place; if it is too large, the increment of x in each step is very small, and Algorithm D.2 may suffer from a slow convergence rate. The optimal value of τ is problem-dependent. There is not a universal way to determine its best value.

Recently, single-loop distributed algorithms for monotone Nash games are proposed in [11], which authors believe to be promising in practical applications. In these two schemes, the regularization parameter is updated at once after each iteration is completed, rather than when the regularized problem is approximately solved, and players can select their parameter independently.

D.1.4 Nash Equilibrium of Matrix Games

As explained before, not all Nash games have an equilibrium, especially when the strategy set and the payoff function are non-convex or discrete. To widen the equilibrium notion and reveal deeper insights on the behaviors of players in such instances, it is instructive to revisit some simple games, called the matrix game, which is the primary research object of game theorists.

The bimatrix game refers to a matrix game involving two players P1 and P2. The numbers of possible strategies of P1 and P2 are m and n , respectively. $A = \{a_{ij}\} \in \mathbb{M}^{m \times n}$ is the payoff matrix of P1: when P1 chooses strategy i and P2 selects strategy j , the payoff of P1 is a_{ij} . The payoff matrix $B \in \mathbb{M}^{m \times n}$ of P2 can be defined in the same way. In a matrix game, each player is interested to determine a probability distribution of his actions, such that his expected payoff is minimized. Let x_i (y_j) be the probability that P1 (P2) will use strategy i (j), vectors $x = [x_1, \dots, x_m]^T$ and $y = [y_1, \dots, y_n]^T$ are called mixed strategies, clearly,

$$\begin{aligned} x &\geq 0, \quad \sum_{i=1}^m x_i = 1 \quad \text{or} \quad x \in \Delta_m \\ y &\geq 0, \quad \sum_{j=1}^n y_j = 1 \quad \text{or} \quad y \in \Delta_n \end{aligned} \tag{D.8}$$

where Δ_m and Δ_n are simplex slices in \mathbb{R}^m and \mathbb{R}^n .

1. Two-person zero-sum games

The zero-sum game represents a totally competitive situation: P1's gain is P2's loss, so the sum of their payoff matrices is $A + B = 0$, as its name suggests. Such type of game has been well studied in vast literature since von Neumann found the famous Minimax Theorem in 1928. The game is revisited from a mathematical programming perspective in [12]. The proposed linear programming method is especially powerful for instances with a high-dimensional payoff matrix. Next, we briefly introduce this method.

Let us begin with a payoff matrix $A = \{a_{ij}\}$, $a_{ij} > 0$, $\forall i, j$ with strictly positive entries (otherwise, we can add a constant to every entry, such that the smallest entry becomes positive, and the equilibrium strategy remains the same). The expected payoff of P1 is given by

$$V_A = \sum_{i=1}^m \sum_{j=1}^n x_i a_{ij} y_j = x^T A y \quad (\text{D.9})$$

which must be positive because of the element-wise positivity assumption on A .

Since $B = -A$ and minimizing $-x^T A y$ is equivalent to maximizing $x^T A y$, the two-person zero-sum game has a min-max form as

$$\min_{x \in \Delta_m} \max_{y \in \Delta_n} x^T A y \quad (\text{D.10})$$

or

$$\max_{y \in \Delta_n} \min_{x \in \Delta_m} x^T A y \quad (\text{D.11})$$

The solution to the two-person zero-sum matrix game (D.10) or (D.11) is called a mixed-strategy Nash equilibrium, or the saddle point of a min-max problem. It satisfies

$$\begin{aligned} (x^*)^T A y^* &\leq x^T A y^*, \quad \forall x \in \Delta_m \\ (x^*)^T A y^* &\geq y^T A x^*, \quad \forall y \in \Delta_n \end{aligned}$$

To solve this game, consider (D.10) in the following format

$$\min_x \{v_1(x) \mid x \in \Delta_m\} \quad (\text{D.12})$$

where $v_1(x)$ is the optimal value function of the problem faced by P2 with the fixed strategy x of P1

$$v_1(x) = \max_y \{x^T A y \mid y \in \Delta_n\}$$

In view of the feasible region defined in (D.8), $v_1(x)$ is equal to the maximal element of vector $x^T A$, which is strictly positive, and the inequality

$$A^T x \leq \mathbf{1}^n v_1(x)$$

holds. Furthermore, introducing a normalized vector $\bar{x} = x/v_1(x)$, we have

$$\begin{aligned} \bar{x} &\geq 0, \quad A^T \bar{x} \leq \mathbf{1}^n \\ v_1(x) &= (\bar{x}^T \mathbf{1}^m)^{-1} \end{aligned}$$

Taking these relations into account, problem (D.12) becomes

$$\begin{aligned} \min_{\bar{x}} \quad & (\bar{x}^T \mathbf{1}^m)^{-1} \\ \text{s.t.} \quad & A^T \bar{x} \leq \mathbf{1}^n \\ & \bar{x} \geq 0 \end{aligned} \quad (\text{D.13})$$

Because the objective is strictly positive and monotonic, the optimal solution of (D.13) keeps unchanged if we choose to maximize $\bar{x}^T \mathbf{1}^m$ under the same constraints, giving rise to the following LP

$$\begin{aligned} \max_{\bar{x}} \quad & \bar{x}^T \mathbf{1}^m \\ \text{s.t.} \quad & A^T \bar{x} \leq \mathbf{1}^n \\ & \bar{x} \geq 0 \end{aligned} \quad (\text{D.14})$$

Let \bar{x}^* and \bar{v}_1^* be the optimal solution and optimal value of LP (D.14). According to the analysis of variable transformation, the optimal expected payoff v_1^* and the optimal mixed strategy x^* of P1 in this game are given by

$$v_1^* = 1/\bar{v}_1^*, \quad x^* = \bar{x}^*/\bar{v}_1^* \quad (\text{D.15})$$

Consider (D.11) in the same way, we obtain the following LP for P2:

$$\begin{aligned} \min_{\bar{y}} \quad & \bar{y}^T \mathbf{1}^n \\ \text{s.t.} \quad & A\bar{y} \geq \mathbf{1}^m \\ & \bar{y} \geq 0 \end{aligned} \quad (\text{D.16})$$

Denote by \bar{y}^* and \bar{v}_2^* the optimal solution and optimal value of LP (D.16), and then the optimal expected payoff v_2^* and the optimal mixed strategy y^* of P2 in this game can be posed as

$$v_2^* = 1/\bar{v}_2^*, \quad y^* = \bar{y}^*/\bar{v}_2^* \quad (\text{D.17})$$

In summary, the mixed-strategy Nash equilibrium of two-person zero-sum matrix game (D.10) is (x^*, y^*) , and the payoff of P1 is v_1^* . Interestingly, we notice that problems (D.14) and (D.16) constitute a pair of dual LPs, implying that their optimal values are equal, and the optimal solution y^* in (D.17) also solves the inner LP of (D.10). This observation leads to two important conclusions:

1) The Nash equilibrium of a two-person zero-sum matrix game, or the saddle point, can be computed by solving a pair of dual LPs. In fact, if one player's strategy, say x^* , have been obtained from (D.14) and (D.15), the rival's strategy can be retrieved by solving (D.10) with $x = x^*$.

2) The decision sequence of a two-person zero-sum game is interchangeable without influencing the saddle point.

2. General bimatrix games

In more general two-person matrix games, the sum of payoff matrices is not equal to zero, and each player wishes to minimize its own expected payoff taking the other player's strategy as given. In the setting of mixed strategies, players are selecting the probability distribution among available

strategies rather than a single action (the pure strategy), and the respective optimization problems are as follows

$$\begin{aligned} \min_x \quad & x^T A y \\ \text{s.t.} \quad & x^T \mathbf{1}^m = 1 : \lambda \\ & x \geq 0 \end{aligned} \tag{D.18}$$

$$\begin{aligned} \min_y \quad & x^T B y \\ \text{s.t.} \quad & y^T \mathbf{1}^n = 1 : \gamma \\ & y \geq 0 \end{aligned} \tag{D.19}$$

The pair of probability distributions (x^*, y^*) is called a mixed-strategy Nash equilibrium if

$$\begin{aligned} (x^*)^T A y^* &\leq x^T A y^*, \quad \forall x \in \Delta_m \\ (x^*)^T B y^* &\leq (x^*)^T B y, \quad \forall y \in \Delta_n \end{aligned}$$

Unlike the zero-sum case, there is not an equivalent LP that can extract the Nash equilibrium. Performing the KKT system approach, we write out the KKT condition for (D.18)

$$\begin{aligned} A y - \lambda \mathbf{1}^m - \mu &= 0 \\ 0 &\leq \mu \perp x \geq 0 \\ x^T \mathbf{1}^m &= 1 \end{aligned}$$

where μ is the dual variable associated with the non-negative constraint, and can be eliminated from the first equality. Concentrating the KKT conditions of LPs (D.18) and (D.19) gives

$$\begin{aligned} 0 &\leq A y - \lambda \mathbf{1}^m \perp x \geq 0, \quad x^T \mathbf{1}^m = 1 \\ 0 &\leq B^T x - \gamma \mathbf{1}^n \perp y \geq 0, \quad y^T \mathbf{1}^n = 1 \end{aligned} \tag{D.20}$$

Complementarity condition (D.20) can be solved by setting $\lambda = \gamma = 1$ and omitting equality constraints, and recovering them at a later normalization step, i.e., we first solve

$$\begin{aligned} 0 &\leq A y - \mathbf{1}^m \perp x \geq 0 \\ 0 &\leq B^T x - \mathbf{1}^n \perp y \geq 0 \end{aligned} \tag{D.21}$$

Suppose that the solution is (\bar{x}, \bar{y}) , then the Nash equilibrium is

$$\begin{aligned} x^* &= \bar{x} / \bar{x}^T \mathbf{1}^m \\ y^* &= \bar{y} / \bar{y}^T \mathbf{1}^n \end{aligned} \tag{D.22}$$

and the corresponding multipliers are derived from (D.20) as

$$\begin{aligned}\lambda^* &= (x^*)^T A y^* \\ \gamma^* &= (x^*)^T B y^*\end{aligned}\tag{D.23}$$

On the other hand, if (x^*, y^*) is a Nash equilibrium and solves (D.20) with multipliers (λ^*, γ^*) , we can observe that $(x^*/\gamma^*, y^*/\lambda^*)$ solves (D.21), therefore

$$\begin{aligned}\bar{x} &= \frac{x^*}{(x^*)^T B y^*} \\ \bar{y} &= \frac{y^*}{(x^*)^T A y^*}\end{aligned}\tag{D.24}$$

Now we can see that identifying the mixed-strategy Nash equilibrium of a bimatrix game entails solving KKT system (D.20) or (D.21), which is called a linear complementarity problem (LCP). A classical algorithm for LCP is the Lemke's method [13, 14]. Another systematic way to solve an LCP is to reformulate it as an MILP using the method described in Appendix B.3.5. Nonetheless, there are more tailored MILP models for LCPs, which will be detailed in Sect. D.4.2.

Unlike the pure-strategy Nash equilibrium, whose existence relies on some assumptions on convexity, the mixed-strategy Nash equilibrium for matrix games, which is the discrete probability distribution among available actions, always exists [2]. If a game with two players has no pure-strategy Nash equilibrium, and each player can choose actions from a finite strategy set, we can then calculate the payoff matrices as well as the mixed-strategy Nash equilibrium, which informs the likelihood that the player will adopt each corresponding pure strategy.

D.1.5 Potential Games

Despite that a direct certification of the existence and uniqueness of a pure-strategy Nash equilibrium for a general game model is non-trivial, when the game possesses some special structures, such a certification becomes axiomatic. One of these guarantees is the existence of an exact potential function, and the associated problem is known as the potential game [15]. Four types of potential games are listed in [15], categorized by the type of the potential function. Other extensions of the potential game have been studied as well. For a complete introduction, we recommend [16].

Definition D.2. (Exact potential game) A game is an exact potential game if there is a potential function $U(x)$ such that:

$$\begin{aligned} f_i(x_i, x_{-i}) - f_i(y_i, x_{-i}) &= U(x_i, x_{-i}) - U(y_i, x_{-i}) \\ \forall x_i, y_i \in X_i, \forall x_{-i} \in X_{-i}, i &= 1, \dots, n \end{aligned} \quad (\text{D.25})$$

In an exact potential game, the change in the utility/payoff of any single player due to the unilateral strategy deviation leads to the same amount of change in the potential function. Among various variations of potential games which are defined by relaxing the strict equality (D.25), the exact potential game is the most fundamental one and has attracted the majority of research interests. Throughout this section, the term potential game means the exact one without particular mention.

The condition for a game being a potential game and the method for constructing the potential function are given in the following proposition.

Proposition D.3. [15] *Suppose the payoff functions f_i , $i = 1, \dots, n$ in a game are twice continuously differentiable, then a potential function exists if and only if*

$$\frac{\partial^2 f_i}{\partial x_i \partial x_j} = \frac{\partial^2 f_j}{\partial x_i \partial x_j}, \quad \forall i, j = 1, \dots, n \quad (\text{D.26})$$

and the potential function can be constructed as

$$U(v) - U(z) = \sum_{i=1}^n \int_0^1 (x'_i(t))^T \frac{\partial f_i}{\partial x_i}(x(t)) dt \quad (\text{D.27})$$

where $x(t) : [0, 1] \rightarrow X$ is a continuously differentiable path in X connecting strategy profile v and a fixed strategy profile z , such that $x(0) = z$, $x(1) = v$.

To obtain (D.27), first, a direct consequence of (D.25) is

$$\frac{\partial f_i}{\partial x_i} = \frac{\partial U}{\partial x_i}, \quad i = 1, \dots, n \quad (\text{D.28})$$

For any smooth curve $C(t) : [0, 1] \rightarrow X$ and any function U with a continuous gradient ∇U , the gradient theorem in calculus tells us

$$U(C_{end}) - U(C_{start}) = \int_C \nabla U(s) ds$$

where vector s represents points along the integral trajectory C parameterized in a scalar variable. Introducing $s = x(t)$: when $t = 0$, $s = C_{start} = z$; when $t = 1$, $s = C_{end} = v$. By the chain rule, $ds = x'(t)dt$, and hence we get

$$\begin{aligned} U(v) - U(z) &= \int_0^1 (x'(t))^T \nabla U(x(t)) dt \\ &= \sum_{i=1}^n \int_0^1 (x'_i(t))^T \frac{\partial U}{\partial x_i}(x(t)) dt \end{aligned}$$

Then, if U is a potential function, substituting (D.28) into above equation gives equation (D.27).

In summary, for a standard NEP with continuous payoff functions, we can check whether it is a potential game, and further construct its potential function, if the right-hand side of (D.27) has a closed form expression. Nevertheless, in some particular cases, the potential function can be observed without calculating an integral.

1. The payoff functions of the game can be decomposed as

$$f_i(x_i, x_{-i}) = p_i(x_i) + Q(x), \quad \forall i = 1, \dots, n$$

where the first term only depends on x_i , and the second term that couples all players' strategies and appears in every utility function is identical. In such circumstance, the potential function is instantly posed as

$$U(x) = Q(x) + \sum_{i=1}^n p_i(x_i)$$

which can be verified through its definition in (D.25).

2. The payoff functions of the game can be decomposed as

$$f_i(x_i, x_{-i}) = p_i(x_{-i}) + Q(x), \quad \forall i = 1, \dots, n$$

where the first term only depends on the joint actions of opponents x_{-i} , and the second term is common and identical to all players. In such circumstance, the potential function is $Q(x)$. This is easy to understand because x_{-i} is constant in the decision-making problem of player i and thus the first term $p_i(x_{-i})$ can be omitted from the objective function.

3. The payoff function of each player has a form of

$$f(x_i, x_{-i}) = \left(a + b \sum_{j=1}^n x_j \right) x_i + c_i(x_i), \quad i = 1, \dots, n$$

Obviously,

$$\frac{\partial^2 f_i}{\partial x_i \partial x_j} = \frac{\partial^2 f_j}{\partial x_i \partial x_j} = b$$

therefore, a potential function exists and is given by

$$U(x) = a \sum_{i=1}^n x_i + \frac{b}{2} \sum_{i=1}^n \sum_{j \neq i}^n x_i x_j + b \sum_{i=1}^n x_i^2 + \sum_{i=1}^n c_i(x_i)$$

The potential function provides a convenient way to analyze the Nash equilibria of potential games, since the function coincides with incentives of all players.

Proposition D.4. [16] *If game G_1 is a potential game with potential function $U(x)$; G_2 is another game with the same number of players and their payoff functions are $F_1(x_1, x_{-1}) = \dots = F_n(x_n, x_{-n}) = U(x)$. Then G_1 and G_2 have the same set of Nash equilibria.*

This is easy to understand because an equilibrium of G_1 satisfies

$$f_i(x_i^*, x_{-i}^*) \leq f_i(x_i, x_{-i}^*), \forall x_i \in X_i, i = 1, \dots, n$$

By the definition of potential function (D.25), this gives

$$U(x_i^*, x_{-i}^*) \leq U(x_i, x_{-i}^*), \forall x_i \in X_i, i = 1, \dots, n \quad (\text{D.29})$$

So any equilibrium of G_1 is an equilibrium of G_2 . Similarly, the reverse holds, too.

In Proposition D.4, the identical interest game G_2 is actually an optimization problem. The potential function builds a bridge between an NEP and a mathematical programming problem. Let $X = X_1 \times \dots \times X_n$, (D.29) can be written as

$$U(x^*) \leq U(x), \forall x \in X \quad (\text{D.30})$$

On this account, we have

Proposition D.5. [15] *Every minimizer of the potential function $U(x)$ in X is a (pure-strategy) Nash equilibrium of the potential game.*

Proposition D.5 is very useful. It reveals the fact that computing a Nash equilibrium of a potential game is equivalent to solving a traditional mathematical program. Meanwhile, the existence and uniqueness results of Nash equilibrium for potential games can be understood from the solution property of NLPs.

Proposition D.6. [16] *Every potential game with a continuous potential function $U(x)$ and a compact strategy space X has at least one (pure-strategy) Nash equilibrium. If $U(x)$ is strictly convex, then the Nash equilibrium is unique.*

Propositions D.5-D.6 make no reference on the convexity of individual payoff functions of players. Moreover, if the potential function $U(x)$ is non-convex and has multiple local minimums, then each local optimizer corresponds to a local Nash equilibrium where X_i in (D.29) is replaced with the intersection of X_i with a neighborhood region of x_i^* .

D.2 Generalized Nash Equilibrium Problem

In above developments for standard NEPs, we have assumed that the strategy sets are decoupled: the available strategies of each player do not depend

on other players' choices. However, there are indeed many practical cases where the strategy sets are interactive. For example, when players consume a common resource, the total consumption should not exceed the inventory quantity. The generalized Nash equilibrium problem (GNEP), invented in [17], relaxes the strategy independence assumption in classic NEPs and allows the feasible set of each player's actions to depend on the rivals' strategies. For a comprehensive review, we recommend [18].

D.2.1 Formulation and Optimality Condition

Denote by $X_i(x_{-i})$ the strategy set of player i when others select x_{-i} . In a GNEP, given the value of x_{-i} , each player i determines a strategy $x_i \in X_i(x_{-i})$ which minimizes a payoff function $f_i(x_i, x_{-i})$. In this regard, a GNEP with n players is the joint solution of n coupled optimization problems

$$\left. \begin{array}{ll} \min_{x_i} & f_i(x_i, x_{-i}) \\ \text{s.t.} & x_i \in X_i(x_{-i}) \end{array} \right\}, \quad i = 1, \dots, n \quad (\text{D.31})$$

In (D.31), correlation takes place not only in the objective function, but also in the constraints.

Definition D.3. A generalized Nash equilibrium (GNE), or the solution of a GNEP, is a feasible point x^* such that

$$f(x_i^*, x_{-i}^*) \leq f(x_i, x_{-i}^*), \quad \forall x_i \in X_i(x_{-i}^*) \quad (\text{D.32})$$

holds for all players.

In its full generality, the GNEP is much more difficult than an NEP due to the variability of strategy sets. In this section, we restrict our attention to a particular class of GNEP: the so-called GNEP with shared convex constraints. In such a problem, the strategy sets can be expressed as

$$X_i(x_{-i}) = \{x_i \mid x_i \in Q_i, g(x_i, x_{-i}) \leq 0\}, \quad i = 1, \dots, n \quad (\text{D.33})$$

where Q_i is a closed and convex set which involves only x_i ; $g(x_i, x_{-i}) \leq 0$ represents the shared constraints. They consist of a set of convex inequalities coupling all players' strategies and are identical in $X_i(x_{-i})$, $i = 1, \dots, n$. Sometimes, Q_i and $g(x_i, x_{-i}) \leq 0$ are also mentioned as local and global constraints, respectively.

In the absence of shared constraints, the GNEP reduces to a standard NEP. Define the feasible set of strategy profile $x = (x_1, \dots, x_n)$ in a GNEP

$$X = \left\{ x \mid x \in \prod_{i=1}^n Q_i, g(x) \leq 0 \right\} \quad (\text{D.34})$$

It is easy to see that $X_i(x_{-i})$ is a slice of X . A geometric interpretation of (D.33) is illustrated in Fig. D.1. It is seen that the choice of x_1 influences the feasible interval $X_2(x_1)$ of Player 2.

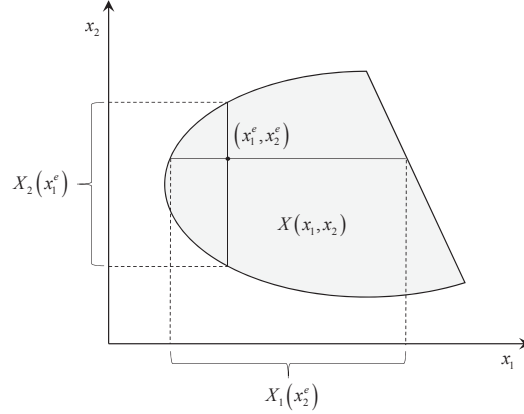


Fig. D.1 Relations of X and the individual strategy sets.

We make some assumptions on the smoothness and convexity for a GNEP with shared constraints.

Assumption D.2

- 1) Strategy set Q_i of each player is nonempty, closed, and convex.
- 2) Payoff function $f_i(x_i, x_{-i})$ of each player is twice continuously differentiable in x and convex in x_i for every fixed x_{-i} .
- 3) Functions $g(x) = (g_1(x), \dots, g_m(x))$ are differentiable and convex in x .

In analogy with the NEP, concatenating the KKT optimality condition of each optimization problem in (D.31) gives us what is called the KKT condition of the GNEP. For notation brevity, we omit local constraints ($Q_i = \mathbb{R}^{n_i}$) and assume that $X_i(x_{-i})$ contains only global constraints. Write out the KKT condition of GNEP (D.31)

$$\left. \begin{aligned} \nabla_{x_i} f_i(x_i, x_{-i}) + \lambda_i^T \nabla_{x_i} g(x) &= 0 \\ \lambda_i &\geq 0, g(x) \leq 0, \lambda_i^T g(x) &= 0 \end{aligned} \right\} i = 1, \dots, n \quad (\text{D.35})$$

where λ_i is the Lagrange multiplier vector associated with the global constraints in the i -th player's problem.

Proposition D.7. [18]

1) Let $\bar{x} = (\bar{x}_1, \dots, \bar{x}_n)$ be the equilibrium of a GNEP, then a multiplier vector $\bar{\lambda} = (\bar{\lambda}_1, \dots, \bar{\lambda}_n)$ exists, such that the pair $(\bar{x}, \bar{\lambda})$ solves KKT system (D.35).

2) If $(\bar{x}, \bar{\lambda})$ solves KKT system (D.35), and Assumption D.2 holds, then \bar{x} is an equilibrium of GNEP (D.31) with shared convex constraints.

However, in contrast to an NEP, the solutions of an GNEP may be non-isolated and constitute a low dimensional manifold, because $g(x)$ is a common constraint shared by all, and the Jacobian of the KKT system may appear to be singular. A meticulous explanation is provided in [18]. We give a graphic interpretation for this phenomenon.

Consider a GNEP with two players:

$$\begin{aligned} \text{Player 1: } & \begin{cases} \max_{x_1} & x_1 \\ \text{s.t.} & x_1 \in X_1(x_2) \end{cases} \\ \text{Player 2: } & \begin{cases} \max_{x_2} & x_2 \\ \text{s.t.} & x_2 \in X_2(x_1) \end{cases} \end{aligned}$$

where

$$\begin{aligned} X_1(x_2) &= \{x_1 \mid x_1 \geq 0, g(x_1, x_2) \leq 0\} \\ X_2(x_1) &= \{x_2 \mid x_2 \geq 0, g(x_1, x_2) \leq 0\} \end{aligned}$$

and the global constraint set is

$$\left\{ x \mid \begin{array}{l} g_1 = 2x_1 + x_2 \leq 0 \\ g_2 = x_1 + 2x_2 \leq 0 \end{array} \right\}$$

The feasible set X of the strategy profile is plotted in Fig. D.2. It can be verified that any point on the line segments

$$L_1 = \left\{ (x_1, x_2) \mid 0 \leq x_1 \leq \frac{2}{3}, \frac{2}{3} \leq x_2 \leq 1, x_1 + 2x_2 = 2 \right\}$$

and

$$L_2 = \left\{ (x_1, x_2) \mid \frac{2}{3} \leq x_1 \leq 1, 0 \leq x_2 \leq \frac{2}{3}, 2x_1 + x_2 = 2 \right\}$$

is an equilibrium point that satisfies Definition D.3.

To refine a meaningful equilibrium from the infinitely many candidates, it is proposed to impose additional conditions on the Lagrange multipliers associated with shared constraints [19]. The outcome is called a restricted Nash equilibrium. Two special cases are discussed here.

1. Normalized Nash equilibrium

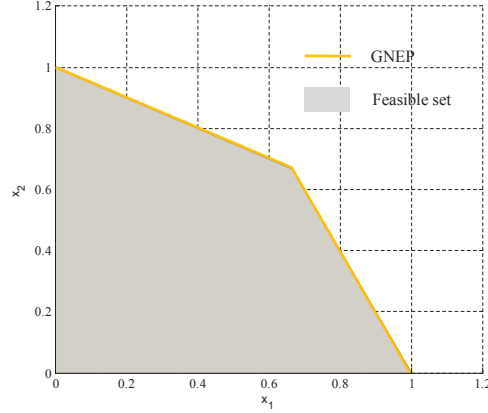


Fig. D.2 Illustration of the equilibria of a simple GNEP.

The normalized Nash equilibrium is firstly introduced in [20]. It incorporates a cone constraint on the dual multipliers

$$\lambda_i = \beta_i \lambda_0, \quad \beta_i > 0, \quad i = 1, \dots, n \quad (\text{D.36})$$

where $\lambda_0 \in \mathbb{R}_+^m$. Solving KKT system (D.35) with constraint (D.36) gives an equilibrium solution. It is shown that for any given $\beta \in \mathbb{R}_{++}^n$, a normalized Nash equilibrium exists as long as the game is feasible. Moreover, if the mapping

$$F(\beta) : \mathbb{R}^n \rightarrow \mathbb{R}^n = \begin{pmatrix} \frac{1}{\beta_1} \nabla_{x_1} f_1(x_1, x_{-1}) \\ \vdots \\ \frac{1}{\beta_n} \nabla_{x_n} f_n(x_n, x_{-n}) \end{pmatrix}$$

parameterized in β is strictly monotone (by assuming convexity of payoff functions), then the normalized Nash equilibrium is unique.

The relation given in (D.36) indicates that the dual variables λ_i associated with the shared constraints are a constant vector scaled by different scalars. From an economic perspective, this means that the shadow prices of common resources at any normalized Nash equilibrium are proportional among each player.

2. Variational equilibrium

Recall the variational inequality formulation for the NEP in Proposition D.1, a GNEP with shared convex constraints can be treated in the same way: Let $F(x) = (\nabla_{x_1} f_1(x), \dots, \nabla_{x_n} f_n(x))$ be a mapping, and the feasible region X is defined in (D.34), then every solution of variational inequality

problem $\text{VI}(X, F)$ gives an equilibrium solution of the GNEP, which is called the variational equilibrium (VE).

However, unlike an NEP and its associated VI problem which have the same solutions, not all equilibria of the GNEP are preserved when it is passed to a corresponding VI problem; see [21, 22] for examples and further details. In fact, a solution x^* of a GNEP is a VE if and only if it solves KKT system (D.35) with the following constraints on the Lagrange dual multipliers [5, 18, 21]:

$$\lambda_1 = \cdots = \lambda_n = \lambda_0 \in \mathbb{R}_+^m \quad (\text{D.37})$$

implying that all players perceive the same shadow prices of common resource at a VE. The VI approach has two important implications. First, it allows us analyze a GNEP using well-developed VI theory, such as conditions which could guarantee the existence and uniqueness of the equilibrium point; second, condition (D.37) gives an interesting economic interpretation of the VE, and inspires pricing-based distributed algorithms to compute an equilibrium solution, which will be discussed in the next section.

The concept of potential game for NEPs directly applies to GNEPs. If a GNEP with shared convex constraints possesses a potential function $U(x)$ which satisfies (D.25), an equilibrium can be retrieved from a mathematical program which minimizes the potential function over the feasible set X defined in (D.34). To reveal the connection of the optimal solution and the VE, we omit constraints in the local strategy sets Q_i , $i = 1, \dots, n$ for notation simplicity, and write out the mathematical program as follows

$$\begin{aligned} \min_x \quad & U(x) \\ \text{s.t.} \quad & g(x) \leq 0 \end{aligned} \quad (\text{D.38})$$

whose KKT optimality condition is given by

$$\begin{aligned} \nabla_x U(x) + \lambda^T \nabla_x g(x) &= 0 \\ \lambda \geq 0, \quad g(x) &\leq 0, \quad \lambda^T g(x) = 0 \end{aligned} \quad (\text{D.39})$$

The first equality can be decomposed into n sub-equations

$$\nabla_{x_i} U(x) + \lambda^T \nabla_{x_i} g(x) = 0, \quad i = 1, \dots, n \quad (\text{D.40})$$

Recall (D.28), $\nabla_{x_i} U(x) = \nabla_{x_i} f_i(x_i, x_{-i})$, substituting it into (D.39) we have

$$\begin{aligned} \nabla_{x_i} f_i(x_i, x_{-i}) + \lambda^T \nabla_{x_i} g(x) &= 0, \quad i = 1, \dots, n \\ \lambda \geq 0, \quad g(x) &\leq 0, \quad \lambda^T g(x) = 0 \end{aligned}$$

which is exactly KKT system (D.35) with identical shadow price constraint (D.37). In this regard, we can see

Proposition D.8. *Optimizing the potential function of a GNEP with shared convex constraints gives a variational equilibrium.*

Consider the example shown in Fig. D.2 again, $(2/3, 2/3)$ is the unique VE of the GNEP, which is plotted in Fig. D.3. The corresponding dual variables of global constraints $g_1 \leq 0$ and $g_2 \leq 0$ are $(1/3, 1/3)$.

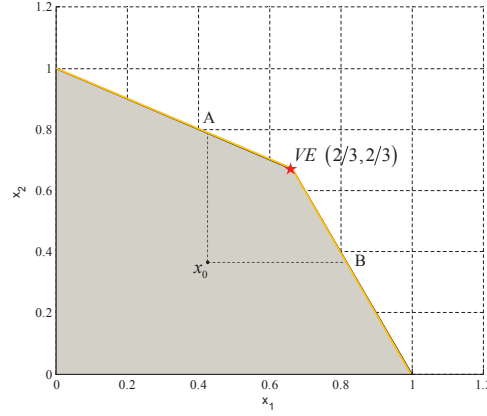


Fig. D.3 Illustration of the variational equilibrium.

D.2.2 Best-Response Algorithm

The presence of shared constraints wrecks the Cartesian structure of $\prod_{i=1}^n Q_i$ in a standard Nash game, and prevents a direct application of the best response methods presented in Appendix D.1.3 to solve an NEP. Moreover, even if an equilibrium can be found, it may depend on the initial point as well as the optimization sequence, because solutions of a GNEP are non-isolated. To illustrate this pitfall, take Fig. D.3 for an example. Suppose we pick up an arbitrary point $x_0 \in X$ as the initial value. If we first maximize x_1 (x_2), the point moves to B (A), and then in the second step, x_2 (x_1) does not change, because it is already an equilibrium solution in the sense of Definition D.3. In view of this, fixed-point iteration may give any outcome on the line segments connecting $(2/3, 2/3)$ and $(0, 1)/(1, 0)$, depending on the initiation.

This section introduces the distributed algorithms proposed in [5] which identify a VE of GNEP (D.31) with shared convex constraints. Motivated by the Lagrange decomposition framework, we can rewrite problem (D.31) in a more convenient form. Consider finding a pair (x, λ) , where x is the equilibrium of the following standard NEP $\mathcal{G}(\lambda)$ with a given vector λ of Lagrange multipliers

$$\mathcal{G}(\lambda) : \left\{ \begin{array}{l} \min_{x_i} f_i(x_i, x_{-i}) + \lambda^T g(x) \\ \text{s.t. } x_i \in Q_i \end{array} \right\}, \quad i = 1, \dots, n \quad (\text{D.41})$$

and furthermore, a complementarity constraint

$$0 \leq \lambda \perp -g(x) \geq 0 \quad (\text{D.42})$$

Problem (D.41)-(D.42) has a clear economic interpretation: suppose the shared constraints represent the availability of some common resources, vector λ can be viewed as the prices paid by players for consuming these resources. Actually, when a resource is adequate, the inequality constraint is not binding and the Lagrange dual multiplier is zero; the dual multiplier or shadow price is positive only if a resource becomes scarce, indicated by a binding inequality constraint. This relation has been imposed in constraint (D.42).

The KKT conditions of $\mathcal{G}(\lambda)$ (D.41) in conjunction with condition (D.42) turn out to be the VE condition of GNEP (D.31). In view of this connection, a VE can be found by solving (D.41)-(D.42) in a distributed manner based on previous algorithms developed for NEPs. Likewise, we discuss strongly convex cases and convex cases separately, due to their different convergence guarantees.

1. Algorithms for strongly convex cases

Suppose that the game $\mathcal{G}(\lambda)$ in (D.41) is strongly convex and has a unique Nash equilibrium $x(\lambda)$ for any given $\lambda \geq 0$. This uniqueness condition allows defining the map

$$\Phi(\lambda) : \lambda \rightarrow -g(x(\lambda)) \quad (\text{D.43})$$

which quantifies the negative violation of the shared constraints at $x(\lambda)$. Based on (D.41)-(D.42), the distributed algorithm is provided as follows.

Algorithm D.3

- 1: Choose an initial price vector $\lambda^0 \geq 0$. The iteration index is $k = 0$.
- 2: Given λ^k , find the unique equilibrium $x(\lambda^k)$ of $\mathcal{G}(\lambda^k)$ using Algorithm D.1.
- 3: If $0 \leq \lambda^k \perp \Phi(\lambda^k) \geq 0$ is satisfied, terminate and report $x(\lambda^k)$ as the VE; otherwise, choose $\tau_k > 0$, and update the price vector according to

$$\lambda^{k+1} = [\lambda^k - \tau_k \Phi(\lambda^k)]^+$$

set $k \leftarrow k + 1$, and go to step 2.

Algorithm D.3 is a double-loop method. The range of parameter τ_n and convergence proof have been thoroughly discussed in [5] based on the mono-

tonicity of the mapping $F + \nabla g(x)\lambda$, where $F = (\nabla_{x_i} f_i)_{i=1}^n$, and $\nabla g(x)$ is a matrix whose i -th column is equal to ∇g_i .

2. Algorithms for convex cases

Now we consider the case in which the VI associated with problem (D.41)-(D.42) is merely monotone (at least one problem in (D.41) is not strongly convex). In such circumstance, the convergence of Algorithm D.3 is no longer guaranteed. This is not only because Algorithm D.1 for the inner loop game $\mathcal{G}(\lambda)$ may not converge, but also because the outer loop has to be complicated. To circumvent this difficulty, we try to convexify the game using regularization terms as what has been done in Algorithm D.2. To this end, we have to explore an optimization reformulation for the complementarity constraint (D.42), which is given by

$$\lambda \in \arg \min_{\bar{\lambda}} \{ -\bar{\lambda}^T g(x) \mid \bar{\lambda} \geq 0 \}$$

Then, consider the following ordinary NEP with $n + 1$ players in which the last player controls the price vector λ :

$$\begin{aligned} \min_{x_i} \quad & \{ f_i(x_i, x_{-i}) + \lambda^T g(x) \mid x_i \in Q_i \}, \quad i = 1, \dots, n \\ \min_{\lambda} \quad & \{ -\lambda^T g(x) \mid \lambda \geq 0 \} \end{aligned} \quad (\text{D.44})$$

where the last player solves an LP in variable λ parameterized in x . At the equilibrium, $g(x) \leq 0$ is implicitly satisfied. To see this, because Q_i is bounded, problems of the first n players must have a finite optimum for arbitrary λ . If $g(x) \not\leq 0$, the last problem has an infinite optimum, imposing a large penalty on the constraint that is violated, and thus the first n players will alter their strategies accordingly. Whenever $g(x) \leq 0$ is met, the last LP must have a zero minimum, which satisfies (D.42). In summary, this extended game (D.44) has the same equilibria as problem (D.41)-(D.42). Since the strategy sets of (D.44) have a Cartesian structure, Algorithm D.2 can be applied to find an equilibrium.

The convergence of Algorithm D.4 is guaranteed under a sufficiently large τ . More quantitative discussions on parameter selection and convergence conditions can be found in [5]. In practice, the value of τ should be carefully chosen to achieve satisfactory computational performances.

In NEPs and GNEPs, players make simultaneous decisions. In real-life decision making problems, there are many situations in which players can move sequentially. In the rest of this chapter, we consider three kinds of bilevel games, in which the upper-level (lower-level) players are called leaders (followers), and leaders make decisions prior to followers. The simplest one is the Stackelberg game, or the single-leader-single-follower game, or just the bilevel program; Stackelberg game can be generalized by incorporating mul-

Algorithm D.4

-
- 1: Given $\{\rho_n\}_{n=0}^\infty$, $\varepsilon > 0$, and $\tau > 0$, choose a feasible initial point $x^0 \in X$ and an initial price vector λ^0 ; the iteration index is $k = 0$.
 - 2: Given $z^k = (x^k, \lambda^k)$, find a Nash equilibrium $z^{k+1} = (x^{k+1}, \lambda^{k+1})$ of the following regularized NEP using Algorithm D.1

$$\begin{aligned} & \min_{x_i} \left\{ f_i(x_i, x_{-i}) + \lambda^T g(x) + \tau \|x_i - x_i^k\|_2^2 \mid x_i \in Q_i \right\}, \quad i = 1, \dots, n \\ & \min_{\lambda} \left\{ -\lambda^T g(x) + \tau \|\lambda - \lambda^k\|_2^2 \mid \lambda \geq 0 \right\} \end{aligned}$$

- 3: If $\|z^{k+1} - z^k\|_2 \leq \varepsilon$, terminate and report x^{k+1} as the variational equilibrium; otherwise, update $k \leftarrow k + 1$, $z^k \leftarrow (1 - \rho_k)z^{k-1} + \rho_k z^k$, and go to step 2.
-

multiple players in the upper and lower levels. Players at the same level make decisions simultaneously, whereas followers' actions are subject to leaders' movements, forming an NEP parameterized in the leaders' decisions. When there is only one leader, the problem is called a mathematical program with equilibrium constraints (MPEC); when there are multiple leaders, the problem is referred to as an equilibrium program with equilibrium constraints (EPEC). It is essentially a bilevel GNEP among the leaders.

D.3 Bilevel Programs

Bilevel program is a special mathematical program with another optimization problem nested in the constraints. The main problem is called the upper-level problem, and the decision maker is the leader; the one nested in constraints is called the lower-level problem, and the decision maker is the follower. In game theory, a bilevel program is usually referred to as the Stackelberg game, which arises in many economic and engineering design problems.

D.3.1 Bilevel Programs with a Convex Lower Level

1. Mathematic model and single-level equivalence

A bilevel program is the most basic instance of bilevel games. The leader moves first and chooses a decision x ; then the follower selects its strategy y solving the lower-level problem parameterized in x

$$\begin{aligned}
& \min_y f(x, y) \\
& \text{s.t. } g(x, y) \leq 0 : \lambda \\
& \quad h(x, y) = 0 : \mu
\end{aligned} \tag{D.45}$$

where λ and μ following the colon are dual variables associated with inequality and equality constraints, respectively. We assume that problem (D.45) is convex and the KKT condition is necessary and sufficient for a global optimum

$$\text{Cons-KKT} = \left\{ (x, y, \lambda, \mu) \left| \begin{array}{l} \nabla_y f(x, y) + \lambda^T \nabla_y g(x, y) + \mu^T \nabla_y h(x, y) = 0 \\ 0 \leq \lambda \perp -g(x, y) \geq 0, \quad h(x, y) = 0 \end{array} \right. \right\} \tag{D.46}$$

The set of optimal solutions of problem (D.45) is denoted by $S(x)$. If (D.45) is strictly convex, the optimal solution is unique, and $S(x)$ reduces to a singleton.

When the leader minimizes its payoff function $F(x, y)$, the best response $y(x) \in S(x)$ is taken into account. The leader's problem is formally described as

$$\begin{aligned}
& \min_{x, \bar{y}} F(x, \bar{y}) \\
& \text{s.t. } x \in X \\
& \quad \bar{y} \in S(x)
\end{aligned} \tag{D.47}$$

Notice that although \bar{y} acts as a decision variable of the leader, it is actually controlled by the follower through the best response mapping $S(x)$. When the leader makes decisions, it will take the response from the follower into account. When $S(x)$ is a singleton, qualifier \in reduces to $=$; otherwise, if $S(x)$ contains more than one elements, (D.47) assumes that the follower will choose the one which is preferred by the leader. Therefore, (D.47) is called an optimistic equivalence. On the contrary, the pessimistic equivalence assumes that the follower will choose the one which is unfavorable for the leader, which is more difficult to solve. As for the optimistic case, replacing $\bar{y} \in S(x)$ with KKT condition (D.46) leads to the NLP formulation of the bilevel program, or more exactly, a mathematical program with complementarity constraints (MPCC)

$$\begin{aligned}
& \min_{x, \bar{y}, \lambda, \mu} F(x, \bar{y}) \\
& \text{s.t. } x \in X, (x, \bar{y}, \lambda, \mu) \in \text{Cons-KKT}
\end{aligned} \tag{D.48}$$

Although the lower-level problem (D.45) is convex, the best reaction map of the follower characterized by Cons-KKT is non-convex, so a bilevel program is intrinsically non-convex and generally difficult to solve.

2. Why bilevel programs are difficult to solve?

Two difficulties prevent an MPCC from being solved reliably and efficiently.

1) The feasible region of (D.47) is non-convex: even if objective functions and constraints of the leader and the follower are linear, the complementarity and slackness condition in (D.46) is still non-convex. An NLP solver only finds a local solution for non-convex problems, if succeeds, and global optimality can hardly be guaranteed.

2) Despite of its non-convexity, the failure to meet ordinary constraint qualifications creates another barrier for solving an MPCC. NLP algorithms generally stop when a stationary point of the KKT conditions is found; however, due to the presence of the complementarity and slackness condition, the dual multipliers may not be well-defined because of the violation of standard constraint qualifications. Therefore, NLP solvers may fail to find a local optimum without particular treatment on the complementarity constraints. To see how constraint qualifications are violated, consider the following simplest linear complementarity constraint

$$x \geq 0, y \geq 0, x^T y = 0, x \in \mathbb{R}^5, y \in \mathbb{R}^5$$

The Jacobian matrix of the active constraints at point (\bar{x}, \bar{y}) is

$$J = \begin{bmatrix} e_{\bar{x}} & 0 \\ 0 & e_{\bar{y}} \\ \bar{y} & \bar{x} \end{bmatrix}$$

where $e_{\bar{x}}$ and $e_{\bar{y}}$ are zero-one matrices corresponding to the active constraints $x_i = 0, i \in I, y_j = 0, j \in J$, where $I \cup J = \{1, 2, 3, 4, 5\}$, and $I \cap J$ is not necessarily empty. Suppose that $I = \{1, 2, 4\}$ and $J = \{3, 5\}$, then

$$J = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ \bar{y}_1 & \bar{y}_2 & \bar{y}_3 & \bar{y}_4 & \bar{y}_5 & \bar{x}_1 & \bar{x}_2 & \bar{x}_3 & \bar{x}_4 & \bar{x}_5 \end{bmatrix}$$

Since $\bar{x}_1 = \bar{x}_2 = \bar{x}_4 = 0$ and $\bar{y}_3 = \bar{y}_5 = 0$, it is apparent that the row vectors of J are linearly dependent at point (\bar{x}, \bar{y}) . The same applies to any (\bar{x}, \bar{y}) regardless of the indices I and J of active constraints, because whenever $y_j > 0$, complementarity will enforce $x_i = 0, i = j$, creating a binding inequality in $x \geq 0$ and a row in matrix J whose i -th element is 1; whenever $x_i > 0$, complementarity will enforce $y_j = 0, j = i$, creating a binding inequality in $y \geq 0$ and a row in matrix J whose $(i + 5)$ -th element is 1. Therefore, the last row of J can be represented by a linear combination of the other rows.

Above discussion and conclusion on linear complementarity constraints also apply to the nonlinear case, because the Jacobian matrix J has the same structure. In this regard, above difficulty is an intrinsic phenomenon in MPCCs.

Proposition D.9. *Complementarity and slackness conditions violate the linear independent constraint qualification at any feasible solution.*

From a geometric perspective, the feasible region of complementarity constraints consists of slices like $x_i = 0, y_j = 0$; there is no strictly feasible point and the Slater's condition does not hold. In conclusion, general purpose NLP solvers are not numerically reliable for solving MPCCs, although they were once used to carry out such tasks.

3. Methods for solving MPCCs

In view of the limitations of standard NLP algorithms, new constraint qualifications are proposed to define stationary solutions so as to solve MPCCs through conventional NLP methods, such as the Bouligand-, Clarke-, Mordukhovich-, weakly-, and Strongly-stationary constraint qualifications. See [23, 24, 25, 26] for further information. Through some proper transformation, MPCCs can be solved via standard NLP algorithms as well. Several approaches are available for this task.

a. Regularization method [28, 29, 30]

In this approach, the non-negativity and complementarity requirements

$$x \geq 0, y \geq 0, xy = 0 \quad (\text{D.49})$$

are approximated by

$$x \geq 0, y \geq 0, xy \leq \varepsilon \quad (\text{D.50})$$

Please note that $xy \geq 0$ is a natural result of non-negativity requirements on x and y . When $\varepsilon = 0$, (D.50) is equivalent to (D.49); when $\varepsilon > 0$, (D.50) defines a larger feasible region than (D.49), so this approach is sometimes called a relaxation method. The smaller ε is, the closer any feasible point (x, y) is to achieve complementarity. if x and y are vectors with non-negative elements, $x^T y = 0$ is the same as $x_i y_i = 0, i = 1, 2, \dots$. The same procedure can be applied if x and y are replaced by nonlinear functions.

Since Slater's condition holds for the feasible set defined by (D.50) with $\varepsilon > 0$, NLP solvers can be used to solve related optimization problem. In a regularization procedure for solving an MPCC, the relaxation (D.50) is applied with gradually decreased value of ε for implementation issues. If the initial value of ε is too small, the solver may be numerically unstable and fail to find a feasible solution.

b. Penalization method [31, 32, 33]

In this approach, the complementarity condition $xy = 0$ is removed from the set of constraints; instead, an associated penalty term xy/ε is added to the objective function to create an extra cost whenever complementarity is not satisfied. Since x and y are non-negative, as indicated by (D.49), the

penalty term would never take a negative value. In this way, the feasible region becomes much simpler.

In a penalization procedure for solving an MPCC, a sequence of NLPs are solved iteratively with gradually decreased value of ε , and the violation of complementarity condition gradually approaches to 0 as iterations proceed. If ε is initiated too small, the penalty coefficient $1/\varepsilon$ is very large which may cause an ill-conditioned problem and numeric instability. One advantage of this iterative procedure is that the optimal solution in iteration k can be used as the initial guess in iteration $k+1$, since the feasible region does not change, and the solution in every iteration is feasible in the next one. A downside of this approach is that the NLP solver generally identifies a local optimum. In consequence, a smaller ε may not necessarily lead to a solution that gets closer to the feasible region.

c. Smoothing method [34, 35]

This approach employs the perturbed Fischer-Burmeister function

$$\phi(x, y, \varepsilon) = x + y - \sqrt{x^2 + y^2 + \varepsilon} \quad (\text{D.51})$$

which is firstly introduced in [36] for LCPs, and shown particularly useful in SQP methods for solving MPCCs in [37]. Clearly, when $\varepsilon = 0$, the function ϕ reduces to the standard Fischer-Burmeister function

$$\phi(x, y, 0) = 0 \iff x \geq 0, y \geq 0, xy = 0 \quad (\text{D.52})$$

$\phi(x, y, 0)$ is not smooth at the origin $(0, 0)$. When $\varepsilon > 0$, the function ϕ satisfies

$$\phi(x, y, \varepsilon) = 0 \iff x \geq 0, y \geq 0, xy = \varepsilon/2 \quad (\text{D.53})$$

and is smooth in x and y .

In view of this, complementarity and slackness condition (D.49) can be replaced by $\phi(x, y, \varepsilon) = 0$ and further embedded in NLP models. When ε tends to 0, (D.49) is enforced approximately.

d. Sequential quadratic programming (SQP) [36, 37, 38]

SQP is a general purpose NLP method. In each iteration of SQP, the quadratic functions in complementarity constraints are approximated by a linear one, and the nonlinear objective function is replaced with their second-order Taylor series, constituting a quadratic program with linear constraints (maybe in conjunction with trust region bounds). At the optimal solution, nonlinear constraints are linearized, the objective function is approximated again, and then the SQP algorithm proceeds to the next iteration.

When applied to an MPCC, the SQP method is often capable of finding a local optimal solution, without a sequence of user-specified ε_k approaching to 0, probably because the SQP solver itself is endowed with some softening ability, e.g., when a quadratic program encounters numeric issues, the SQP

solver SNOPT automatically relaxes some hard constraints and penalizes violations in the objective function.

The aforementioned classical methods are discussed in [27], and numeric experiences are reported in [39].

e. MINLP methods

Due to the wide applications in various engineering disciplines, solution methods of MPCCs continue to be an active research area. Recall that the complementarity constraints in form of $g(x) \geq 0$, $h(x) \geq 0$, $g(x)h(x) = 0$ is equivalent to

$$0 \leq g(x) \leq Mz, \quad 0 \leq h(x) \leq M(1 - z)$$

where z is a binary variable, M is a sufficiently large constant. Therefore, an MPCC can be converted to a mixed integer nonlinear program (MINLP). MINLP removes the numeric difficulty in MPCC; however, the computation complexity remains. If all functions in (D.48) are linear, or there are only a few complementarity constraints, the resulting MILP or MINLP model may be solved within reasonable time; otherwise, the branch-and-bound algorithm could offer upper and lower bounds on the optimal value.

f. Convex relaxation/approximation methods

If all functions in MPCC (D.48) are linear, it is a non-convex QCQP in which non-convexity originates from the complementarity constraints. When the problem scale is large, the MILP method may be time-consuming. Inspired by the success of convex relaxation methods in non-convex QCQPs, there have been increasing interests for developing convex relaxation methods for MPCCs. An SDP relaxation method is proposed in [40], which is embedded in a branch-and-bound algorithm to solve the MPCC. For the MPCC derived from a bilevel polynomial program, it is proposed to solve a sequence of SDPs with increasing problem sizes, so as to solve the original problem globally [41, 42]. Convex relaxation methods have been applied to power market problems in [43, 44]. Numerical experiments show that the combination of MILP and SDP relaxation can greatly reduce the computation time. Nonetheless, please bear in mind that in the SDP relaxation model, the decision variable is a matrix with a dimension of $n \times n$, so solving the SDP model may still be a challenging task, although it is convex.

Recently, a DC programming approach is proposed in [45] to solve LPCC in the penalized version. In this approach, the quadratic penalty term is decomposed into the difference of two convex quadratic functions, and the concave part is then linearized. Computational performances reported in [45] are very promising.

D.3.2 Special Bilevel Programs

Although general bilevel programs are difficult, there are special cases which can be solved relatively easily. One of such classes of programs is the linear bilevel program, in which objective functions are linear and constraints are polyhedra. The linear max-min problem is a special case of the linear bilevel program, in which the leader and the follower have completely opposite targets. Furthermore, two special market models are studied.

1. Linear bilevel program

A linear bilevel program can be written as

$$\begin{aligned}
 \max_x \quad & c^T x + d^T y(x) \\
 \text{s.t.} \quad & Cx \leq d \\
 & y(x) \in \arg \min_y f^T y \\
 & \text{s.t. } By \leq b - Ax
 \end{aligned} \tag{D.54}$$

In problem (D.54), the follower makes a decision y after the leader deploys its action x , which influences the feasible region of y . Meanwhile, the leader can predict the follower's optimal response $y(x)$, and choose a strategy that finally optimizes $c^T x + d^T y(x)$. Other matrices and vectors are constant coefficients.

Given the upper level decision x , the follower is facing an LP, whose KKT optimality condition is given by

$$\begin{aligned}
 B^T u &= f \\
 0 &\geq u \perp Ax + By - b \leq 0
 \end{aligned}$$

The last constraint is equivalent to the following linear constraints

$$\begin{aligned}
 -M(1 - z) &\leq u \leq 0 \\
 -Mz &\leq Ax + By - b \leq 0
 \end{aligned}$$

where z is a vector consisting of binary variables, and M is a large enough constant.

In problem (D.54), replacing follower's LP with its KKT condition gives rise to an MILP

$$\begin{aligned}
 \max_{x,y,u,z} \quad & c^T x + d^T y \\
 \text{s.t.} \quad & Cx \leq d, \quad B^T u = f \\
 & -M(1 - z) \leq u \leq 0 \\
 & -Mz \leq Ax + By - b \leq 0
 \end{aligned} \tag{D.55}$$

If the number of complementarity constraints is moderate, MILP (D.55) can be often solved efficiently, despite of its NP-hard complexity in the worst-case.

Since MILP solvers and computation hardware keep improving nowadays, it is always worthy of bearing this technique in mind. Please also be aware that the big-M parameter notably impacts the performance of solving MILP (D.55). A heuristic method to determine such a parameter in linear bilevel programs is proposed in [46]. This method firstly solves two LPs and generates a feasible solution of the equivalent MPCC; then solves a regularized version of the MPCC model using NLP solvers and identifies a local optimal solution near the obtained feasible point; finally, the big-M parameter and the binary variables are initiated according to the local optimal solution. In this way, no manually-supplied parameter is needed, and the MILP model is properly strengthened.

Another optimality certification of follower's LP is the following primal-dual optimality condition

$$\begin{aligned} B^T u &= f, \quad u \leq 0, \quad Ax + By \leq b \\ u^T (b - Ax) &= f^T y \end{aligned}$$

The first line summarizes feasible regions of the primal and dual variables. The last equation enforces equal values on the optimums of the primal and the dual problems, which is known as the strong duality condition.

Replacing follower's LP with the primal-dual optimality condition gives an NLP:

$$\begin{aligned} \max_{x,y,u} \quad & c^T x + d^T y \\ \text{s.t.} \quad & Cx \leq d, \quad Ax + By \leq b \\ & u \leq 0, \quad B^T u = f \\ & u^T (b - Ax) = f^T y \end{aligned} \tag{D.56}$$

The following discussion are divided in two categories based on the type of variable x .

a. x is **continuous**. In such a general situation, there is no effective way to solve problem (D.56), due to the last bilinear equality. Notice the fact that $f^T y \geq u^T (b - Ax)$ always holds on the feasible region because of the weak duality, the last constraint can be relaxed and penalized in the objective function, resulting in a bilinear program over a polyhedron [47, 48, 49]

$$\begin{aligned} \max_{x,y,u} \quad & c^T x + d^T y - \sigma [f^T y - u^T (b - Ax)] \\ \text{s.t.} \quad & Cx \leq d, \quad Ax + By \leq b \\ & u \leq 0, \quad B^T u = f \end{aligned} \tag{D.57}$$

where $\sigma > 0$ is a penalty parameter. In problem (D.57), the constraints on u and (x, y) are decoupled, so this problem can be solved by Algorithm C.1 (mountain climbing) in Appendix C.2.3, if global optimality is not mandatory.

In some problems, the upper-level decision influences the lower-level cost function, and has no impact on the feasible region in the lower level. For example, the tax rate design or a retail market pricing belongs to such category. The same procedure can be performed to solve this kind of bilevel problem. We recommend the MILP model, because in the penalized model, both $f^T y$ and $u^T Ax$ are non-convex. A tailored retail market model will be introduced later.

b. x is **binary**. In such circumstance, the bilinear term $u^T Ax = \sum_{ij} A_{ij} u_i x_j$ can be linearized by replacing $u_i x_j$ with a new continuous variable v_{ij} together with auxiliary linear inequalities enforcing $v_{ij} = u_i x_j$. In this way, the last inequality translates into

$$u^T b - \sum_{ij} A_{ij} v_{ij} = f^T y$$

$$u_i^l x_j \leq v_{ij} \leq 0, \quad u_i^l (1 - x_j) \leq u_i - v_{ij} \leq 0, \quad \forall i, j$$

where u_i^l is a proper bound that does not discard the original optimal solution. As we can see, a bilevel linear program with binary upper-level variables is not necessarily harder than all continuous instances. This formulation is very useful to model interdiction problems in which x mimics attack strategy.

c. x can be **discretized**. Even if x is continuous, we can approximate it via binary expansion

$$x_i = x_i^l + \Delta_i \sum_{k=0}^K 2^k z_{ik}, \quad z_{ik} \in \{0, 1\}$$

where x_i^l (x_i^m) is the lower (upper) bound of x_i , and $\Delta_i = (x_i^m - x_i^l)/2^{K+1}$ is the step size. With this transformation, the bilinear term $u^T Ax$ becomes

$$\sum_{ij} A_{ij} u_i x_i^l + \sum_{ij} A_{ij} u_i \Delta_j \sum_{k=0}^K 2^k z_{jk}$$

The first term is linear, and $u_i z_{jk}$ in the second term can be linearized in a similar way. However, this entails introducing continuous variable with respect to indices i , k and k . A low-complexity linearization method is suggested in [50]. It re-orders the summations in the second term as

$$\sum_j \sum_{k=0}^K \Delta_j 2^k z_{jk} \sum_i A_{ij} u_i$$

which can be linearized through defining an auxiliary continuous variable $v_{jk} = z_{jk} \sum_i A_{ij} u_i$ and stipulating

$$-M z_{jk} \leq v_{jk} \leq M z_{jk}, \quad -M(1 - z_{jk}) \leq \sum_i A_{ij} u_i - v_{jk} \leq M(1 - z_{jk})$$

where M is a large enough constant.

The core idea behind this trick is to treat $u^T A$ as a whole vector which has the same dimension as x , because for bilinear form $x^T v = \sum_i x_i v_i$, the dimension of summation is one, while for $x^T Q v = \sum_{ij} Q_{ij} x_i v_j$, the dimension of summation is two. This observation inspires us to conform vector dimensions while deploying such linearization.

2. Linear max-min problem

A linear max-min problem is a special case of the linear bilevel program, which can be written as

$$\begin{aligned} \max_x \quad & c^T x + d^T y(x) \\ \text{s.t.} \quad & x \in X \\ & y(x) \in \arg \min_y c^T x + d^T y \\ & \text{s.t. } y \in Y, \quad By \leq b - Ax \end{aligned} \tag{D.58}$$

In problem (D.58), the follower seeks an objective that is completely opposite to that of the leader. This kind of problem frequently arises in robust optimization and has been discussed in Appendix C.2.3 from the computational perspective. Here we revisit it from a game theoretical point of view.

Problem (D.58) can be expressed as a two-person zero-sum game

$$\max_{x \in X} \min_{y \in Y} \{c^T x + d^T y \mid Ax + By \leq b\} \tag{D.59}$$

However, the coupled constraints make it different from a saddle point problem in the sense of a Nash game or a matrix game. Indeed, it is a Stackelberg game. Let us investigate the interchangeability of the max and min operators (decision sequence). We have already shown in Appendix D.1.4 that swapping the order of max and min operators in a two-person zero-sum matrix game does not influence the equilibrium. However, this is not the case of (D.59) [51], because

$$\begin{aligned}
& \max_{x \in X} \min_{y \in Y} \{c^T x + d^T y \mid Ax + By \leq b\} \\
&= \max_{x \in X} \left\{ c^T x + \min_{y \in Y} \{d^T y \mid By \leq b - Ax\} \right\} \\
&\geq \max_{x \in X} \left\{ c^T x + \min_{y \in Y} d^T y \right\} \\
&= \max_{x \in X} c^T x + \min_{y \in Y} d^T y \\
&= \min_{y \in Y} \left\{ d^T y + \max_{x \in X} c^T x \right\} \\
&\geq \min_{y \in Y} \left\{ d^T y + \max_{x \in X} \{c^T x \mid Ax \leq b - By\} \right\} \\
&= \min_{y \in Y} \max_{x \in X} \{c^T x + d^T y \mid Ax + By \leq b\}
\end{aligned}$$

In fact, strict inequality usually holds in the third and sixth line. This result implies that owing to the presence of strategy coupling, the leader rests in a superior status, which is different from the Nash game in which players possess the same positions.

To solve linear max-min problem (D.59), there is no doubt that the aforementioned MILP transformation for general linear bilevel programs gives a possible mean for this task. Nevertheless, the special structure of (D.59) allows several alternatives which are more dedicated and effective. To this end, we will transform it into an equivalent optimization problem using LP duality theory. For the ease of notation, we merge polytope Y into the coupled constraint, and the dual of lower-level LP in (D.58) (or the inner LP in (D.59)) reads

$$\max_u \{u^T(b - Ax) \mid u \in U\}$$

where $U = \{u \mid B^T u = d, u \leq 0\}$ is the feasible region of dual variable u . As strong duality always holds for LPs, we have $d^T y = u^T(b - Ax)$. Substituting it into (D.58) we obtain

$$\begin{aligned}
& \max \quad c^T x + u^T b - u^T Ax \\
& \text{s.t.} \quad x \in X, u \in U
\end{aligned} \tag{D.60}$$

Problem (D.60) is a bilinear program due to the product term $u^T Ax$ in variables u and x . Several methods for solving such a problem locally or globally have been set forth in Appendix C.2.3, as a fundamental methodology in robust optimization. Although variable y of the follower does not appear in (D.60), it can be easily recovered from the lower level of (D.58) with the obtained leader's strategy x .

3. A retail market problem

In a retail market, a retailer releases the prices of some goods; according to the retail prices, the customer decides on the optimal purchasing strategy subject to the demands on each goods as well as production constraints; finally, the retailer produces or trades with a higher level market to manage the inventory, and delivers the goods to customers. This retail market can be modeled through a bilevel program. In the upper level

$$\max_{x,z} \quad x^T D_C y(x) - p^T D_M z \quad (\text{D.61a})$$

$$\text{s.t.} \quad Ax \leq a \quad (\text{D.61b})$$

$$B_1 y(x) + B_2 z \leq b \quad (\text{D.61c})$$

(D.61a)-(D.61c) form retailer's problem, where vector x denotes the prices of goods released by the retailer; vector $y(x)$ stands for the amounts of goods purchased by the customer, which is determined from an optimal production planning problem; p is the production cost or the price in the higher level market; z represents the production/purchase strategy of the retailer. Other matrices and vectors are constant coefficients. The first term in objective function (D.61a) is the income paid by the customer, and the second term is the payoff of the retailer. The objective function is the total profit to be maximized. Because there is no competition and the retailer has full market power, to avoid unfair retail prices, we assume that both sides have reached certain agreements on the pricing policy, which is modeled through constraint (D.61b). It includes simple lower and upper bounds as well as other bilateral contract, such as the restriction on the average price over a certain period or the price correlation among multiple goods. The inventory dynamics and other technique constraints are depicted by constraint (D.61c).

Given the retail prices, customers solve the optimal production planning problem in the lower level

$$\begin{aligned} \min_y \quad & x^T D_C y \\ \text{s.t.} \quad & Fy \geq f \end{aligned} \quad (\text{D.62})$$

and determine the optimal purchasing strategy. The objective function in (D.62) is the total cost of customers, where the price vector x is constant coefficient; constraints capture the demands and all other technique requirements in the production process.

Bilevel program (D.61)-(D.62) are not linear, although (D.62) is indeed an LP, because of the bilinear term $x^T D_C y$ in (D.61a), where both x and y are variables (the retailer controls y indirectly through prices). The KKT condition of LP (D.62) reads

$$\begin{aligned} D_C^T x &= F^T u \\ 0 &\leq u \perp Fy - f \geq 0 \end{aligned}$$

where u is the dual variable. The complementarity constraints can be linearized via binary variables, which has been clarified in Appendix B.3.5. Furthermore, strong duality gives

$$x^T D_C y = f^T u$$

The right-hand side is linear in u . Therefore, problem (D.61)-(D.62) and the following MILP

$$\begin{aligned} \max_{x,y,u,v,z} \quad & f^T u - p^T D_M z \\ \text{s.t.} \quad & Ax \leq a, \quad B_1 y + B_2 z \leq b \\ & v \in \mathbb{B}^{N_f}, \quad D_C^T x = F^T u \\ & 0 \leq u \leq M(1 - v) \\ & 0 \leq Fy - f \leq Mv \end{aligned} \tag{D.63}$$

have the same optimal solution in primal variables, where N_f is the dimension of f .

We can learn from this example that when the problem exhibits a certain structure, the non-convexity can be eliminated without introducing additional dimensions of complexity. In problem (D.61), the price is a primal variable quoted by a decision maker, and is equal to the knock-down price. This scheme is called pay-as-bid. Next, we give an example of a marginal pricing market where the price is determined by the dual variables of a market clearing problem.

4. A wholesale market problem

In a wholesale market, a provider bids its offering prices to a market organizer. The organizer collects information on available resources and the bidding of the provider, and then clears the market by scheduling the production in the most economic way. The provider is paid at the marginal cost. This problem can be modeled by a bilevel program

$$\max_{\beta} \lambda(\beta)^T p(\beta) - f(p(\beta)) \tag{D.64}$$

where β is the offering price vector of the provider, $p(\beta)$ is the quantity of goods ordered by the market organizer, function $f(p) = \sum_i f_i(p_i)$, where $f_i(p_i)$ is a univariate convex function representing the production cost, and $\lambda(\beta)$ is the marginal prices of each kind of goods. Both of them depend on the value of β , and are determined from the market clearing problem in the lower level

$$\min_{p,u} \quad \beta^T p + c^T u \quad (\text{D.65a})$$

$$\text{s.t.} \quad p_n \leq p \leq p_m : \eta_n, \eta_m \quad (\text{D.65b})$$

$$p + Fu = d : \lambda \quad (\text{D.65c})$$

$$Au \leq a : \xi \quad (\text{D.65d})$$

where u includes all other variables, such as the amount of each kind of goods collected from other providers or produced locally, the system operating variable, and so on; c is the coefficient including prices of goods offered by other providers, and the production cost if the organizer wishes to produce the goods by itself. Objective function (D.65a) represents the total cost in the market to be minimized. Constraint (D.65b) defines offering limits of the upper-level provider; constraint (D.65c) is the system-wide production-demand balancing condition of each goods, the dual variable λ at the optimal solution gives the marginal cost of each goods; (D.65d) imposes constraints which the system operation must obey, such as network flow and inventory dynamics.

In the provider's problem (D.64), the offering price β is not restricted by finite upper bounds pricing policies (but such a policy can certainly be modeled), because the competition appears in the lower level: if β is not reasonable, the market organizer would resort to other providers or count on its own production capability.

Compared with the situation in a retail market, problems (D.64)-(D.65) are even more complicated: the dual variable λ appears in the objective function of the provider, and the term $\lambda^T p$ is non-convex. In the following, we reveal that it can be exactly expressed as a linear function in the primal and dual variables via (somehow tricky) algebraic transformations.

KKT conditions of the market clearing LP (D.65) are summarized as follows

$$\beta = \lambda + \eta_n + \eta_m \quad (\text{D.66a})$$

$$\eta_n^T (p - p_n) = 0 \quad (\text{D.66b})$$

$$\eta_m^T (p_m - p) = 0 \quad (\text{D.66c})$$

$$c = A^T \xi + F^T \lambda \quad (\text{D.66d})$$

$$\xi^T (Au - a) = 0 \quad (\text{D.66e})$$

$$\eta_n \geq 0, \eta_m \leq 0, \xi \leq 0 \quad (\text{D.66f})$$

$$(D.65b) - (D.65d) \quad (\text{D.66g})$$

According to (D.66a),

$$\beta^T p = \lambda^T p + \eta_n^T p + \eta_m^T p \quad (\text{D.67a})$$

From (D.66b) and (D.66c) we have

$$\eta_n^T p = \eta_n^T p_n, \quad \eta_m^T p = \eta_m^T p_m, \quad (\text{D.67b})$$

Substituting (D.67b) in (D.67a) renders

$$\lambda^T p = \beta^T p - \eta_n^T p_n - \eta_m^T p_m \quad (\text{D.67c})$$

Furthermore, strong duality of LP implies the following equality

$$\beta^T p + c^T u = \eta_n^T p_n + \eta_m^T p_m + d^T \lambda + a^T \xi$$

or

$$\beta^T p - \eta_n^T p_n - \eta_m^T p_m = d^T \lambda + a^T \xi - c^T u \quad (\text{D.67d})$$

Substituting (D.67d) in (D.67c) results in

$$\lambda^T p = d^T \lambda + a^T \xi - c^T u \quad (\text{D.67e})$$

The right-hand side is a linear expression for $\lambda^T p$ in primal variable u and dual variables λ and ξ .

Combining the KKT condition (D.66) and (D.67e) gives an MPCC which is equivalent to the bilevel wholesale market problem (D.64)-(D.65)

$$\begin{aligned} \max \quad & d^T \lambda + a^T \xi - c^T u - f(p(\beta)) \\ \text{s.t.} \quad & (\text{D.66a}) - (\text{D.66g}) \end{aligned} \quad (\text{D.68})$$

Because complementarity conditions (D.66b), (D.66c), (D.66e) can be linearized, and convex function $f(p)$ can be approximated by PWL functions, MPCC (D.68) can be recast as an MILP.

D.3.3 Bilevel Mixed-integer Program

Although LP can tackle many economic problems and market activities in real life, there are indeed even more decision-making problems which are beyond the reach of LP, for example, power market clearing considering unit commitment [52]. KKT optimality condition or strong duality from LP theory do not apply to discrete optimization problems due to their intrinsic non-convexity. Furthermore, this is no computationally viable approach to express the optimality condition of a general discrete program in closed form, making a bilevel mixed-integer programs much more challenging to solve than a bilevel linear program. Some traditional algorithms either rely on enumerative branch-and-bound strategies based on a weak relaxation or depends on complicated operations that are problem-specific. To our knowledge, the reformulation and decomposition algorithm proposed in [53] is the first approach that can solve general bilevel mixed-integer programs in a systematic way, and will be introduced in this section.

The bilevel mixed-integer program has the following form

$$\begin{aligned}
\min \quad & f^T x + g^T y + h^T z \\
\text{s.t.} \quad & Ax \leq b, \ x \in \mathbb{R}^{m_c} \times \mathbb{B}^{m_d} \\
& (y, z) \in \arg \max \ w^T y + v^T z \\
& \text{s.t.} \ P y + N z \leq r - K x \\
& y \in \mathbb{R}^{n_c}, \ z \in \mathbb{B}^{n_d}
\end{aligned} \tag{D.69}$$

where x is the upper-level decision variable and appears in constraints of the lower-level problem; y and z represent lower-level continuous decision variable and discrete decision variable, respectively. We do not distinguish upper-level continuous variable and discrete variable because they have little impact on the exposition of the algorithm, unlike the ones appeared in the lower level. If the lower-level has multiple solutions, the follower chooses the one in favor of the leader. In the current form, the upper-level constraints are independent of lower-level variables. Nevertheless, coupling constraints in the upper level can be easily incorporated [54].

In this section, we assume that the relatively complete recourse property in [53] holds, i.e., for any feasible pair (x, z) , the feasible set for lower-level continuous variable y is non-empty. Under this premise, the optimal solution exists. This assumption is mild because we can add slack variables in the lower-level constraints and penalize constraint violation in the lower-level objective function. For instances in which the relatively complete recourse property is missing, please refer to the remedy in [54].

To eliminate \in qualifier in (D.69), we duplicate decision variables and constraints of the lower-level problem and set up an equivalent formulation:

$$\begin{aligned}
\min \quad & f^T x + g^T y^0 + h^T z^0 \\
\text{s.t.} \quad & Ax \leq b, \ x \in \mathbb{R}^{m_c} \times \mathbb{B}^{m_d} \\
& Kx + P y^0 + N z^0 \leq r \\
& w^T y^0 + v^T z^0 \geq \max \ w^T y + v^T z \\
& \text{s.t.} \ P y + N z \leq r - K x \\
& y \in \mathbb{R}^{n_c}, \ z \in \mathbb{B}^{n_d}
\end{aligned} \tag{D.70}$$

In this formulation, the leader controls its original variable x as well as replicated variables y^0 and z^0 . Conceptually, the leader will use (y^0, z^0) to anticipate the response of follower and its impact on his objective function. Clearly, if the lower-level problem has a unique optimal solution, it must be equal to (y^0, z^0) . It is worth mentioning that although more variables and constraints are incorporated in (D.70), this formulation is actually an informative and convenient expression for algorithm development, as \geq would be more friendly to general purpose mathematical programming solvers.

Up to now, the obstacle of solving (D.70) remains: discrete variable z in the lower level, which prevents the use of optimality condition of LP. To overcome this difficulty, we treat y and z separately and restructure the lower-level problem as:

$$w^T y^0 + v^T z^0 \geq \max_{z \in Z} v^T z + \max_y \{w^T y | Py \leq r - Kx - Nz\} \quad (\text{D.71})$$

where Z represents the set consisting of all possible values of z . Despite the large cardinality of Z , the second optimization is a pure LP, and can be replaced with its KKT condition, resulting in:

$$\begin{aligned} w^T y^0 + v^T z^0 &\geq \max_{z \in Z} v^T z + w^T y \\ \text{s.t. } P^T \pi &= w \\ 0 &\leq \pi \perp r - Kx - Nz - Py \geq 0 \end{aligned} \quad (\text{D.72})$$

The complementarity constraints can be linearized via the method in Sect. B.3.5. Then, by enumerating z^j over Z with associated variables (y^j, π^j) , we arrive at an MPCC that is equivalent to problem (D.70)

$$\begin{aligned} \min \quad & f^T x + g^T y^0 + h^T z^0 \\ \text{s.t. } \quad & Ax \leq b, \quad x \in \mathbb{R}^{m_c} \times \mathbb{B}^{m_d} \\ & Kx + Py^0 + Nz^0 \leq r, \quad P^T \pi^j = w, \quad \forall j \\ & 0 \leq \pi^j \perp r - Kx - Nz^j - Py^j \geq 0, \quad \forall j \\ & w^T y^0 + v^T z^0 \geq w^T y^j + v^T z^j, \quad \forall j \end{aligned} \quad (\text{D.73})$$

Without particular mention, (D.73) is compatible with MILP solvers.

Except for the KKT optimality condition, another popular approach entails applying primal-dual condition for the LP regarding the lower-level continuous variable y . Following this line, rewrite this LP in (D.71) by strong duality, we obtain

$$\begin{aligned} w^T y^0 + v^T z^0 &\geq \max_{z \in Z} v^T z + \min \pi^T (r - Kx - Nz) \\ \text{s.t. } \quad & P^T \pi = w, \quad \pi \geq 0 \end{aligned} \quad (\text{D.74})$$

In (D.74), if all variables in x are binary, the bilinear terms $\pi^T Kx$ and $\pi^T Nz$ from the leader's point of view can be linearized via the method in Sect. B.2.2. The min operator in the right-hand side can be omitted because the upper-level objective function is to be minimized, giving rise to

$$\begin{aligned}
& \min \quad f^T x + g^T y^0 + h^T z^0 \\
& \text{s.t.} \quad Ax \leq b, \quad x \in \mathbb{R}^{m_c} \times \mathbb{B}^{m_d} \\
& \quad \quad Kx + Py^0 + Nz^0 \leq r \\
& \quad \quad w^T y^0 + v^T z^0 \geq v^T z^j + (r - Kx - Nz)^T \pi^j, \quad \forall j \\
& \quad \quad P^T \pi^j = w, \quad \pi^j \geq 0, \quad \forall j
\end{aligned} \tag{D.75}$$

Clearly, (D.75) has fewer constraints compared to (D.73). Nevertheless, whenever x contains continuous variables, linearizing $\pi^T Kx$ would incur more binary variables.

One may think that it is hopeless to solve above enumeration forms (D.73) and (D.75) due to the large cardinality of Z . In a way similar to the CCG algorithm for solving robust optimization, we can start with a subset of Z and solve relaxed version of problem (D.73), until the lower bound and upper bound of optimal value converge. The flowchart is shown in Algorithm D.5

Because Z has finite elements, Algorithm D.5 must terminate in a finite number of iterations, which is bounded by the cardinality of Z . When it converges, LB equals to UB without a positive gap.

To see this, suppose that in iteration l_1 , (x^*, y^{0*}, z^{0*}) is obtained in step 2 with $\text{LB} < \text{UB}$, and z^* is produced in step 4. Particularly, we assume that z^* was previously derived in some iteration $l_0 < l_1$. Then, in step 5, new variables and cuts associated with $z^* = z^{l_1+1}$ will be generated and augmented with the master problem. As those variables and constraints already exist after iteration l_0 , the augmentation is essentially redundant, and the optimal value of master problem in iteration $l_1 + 1$ remains the same as that in iteration l_1 , so does LB. Consequently, in iteration $l_1 + 1$

$$\begin{aligned}
\text{LB} &= f^T x^* + g^T y^{0*} + h^T z^{0*} \\
&= f^T x^* + \min g^T y^0 + h^T z^0 \\
&\quad \text{s.t.} \quad Py^0 + Nz^0 \leq r - Kx^*, \quad P^T \pi^j = w, \quad \forall j \leq l_1 + 1 \\
&\quad \quad 0 \leq \pi^j \perp r - Kx^* - Nz^j - Py^j \geq 0, \quad \forall j \leq l_1 + 1 \\
&\quad \quad w^T y^0 + v^T z^0 \geq w^T y^j + v^T z^j, \quad \forall j \leq l_1 + 1 \\
&\geq f^T x^* + \min g^T y^0 + h^T z^0 \\
&\quad \text{s.t.} \quad Py^0 + Nz^0 \leq r - Kx^*, \quad P^T \pi^j = w, \quad j = l_1 + 1 \\
&\quad \quad 0 \leq \pi^j \perp r - Kx^* - Nz^j - Py^j \geq 0, \quad j = l_1 + 1 \\
&\quad \quad w^T y^0 + v^T z^0 \geq w^T y^j + v^T z^j, \quad j = l_1 + 1 \\
&\geq f^T x^* + \min g^T y^0 + h^T z^0 \\
&\quad \text{s.t.} \quad Py^0 + Nz^0 \leq r - Kx^* \\
&\quad \quad w^T y^0 + v^T z^0 \geq \theta(x^*) \\
&= f^T x^* + \Theta(x^*)
\end{aligned}$$

Algorithm D.5 : CCG algorithm for bilevel MILP

- 1: Set $LB = -\infty$, $UB = +\infty$, and $l = 0$;
- 2: Solve the following master problem

$$\begin{aligned}
\min \quad & f^T x + g^T y^0 + h^T z^0 \\
\text{s.t.} \quad & Ax \leq b, \quad x \in \mathbb{R}^{m_c} \times \mathbb{B}^{m_d} \\
& Kx + Py^0 + Nz^0 \leq r, \quad P^T \pi^j = w, \quad \forall j \leq l \\
& 0 \leq \pi^j \perp r - Kx - Nz^j - Py^j \geq 0, \quad \forall j \leq l \\
& w^T y^0 + v^T z^0 \geq w^T y^j + v^T z^j, \quad \forall j \leq l
\end{aligned} \tag{D.76}$$

The optimal solution is $(x^*, y^{0*}, z^{0*}, y^{1*}, \dots, y^{l*}, \pi^{1*}, \dots, \pi^{l*})$, and the optimal value is v^* . Update lower bound $LB = v^*$.

- 3: Solve the following lower-level MILP with obtained x^*

$$\begin{aligned}
\theta(x^*) = \max \quad & w^T y + v^T z \\
\text{s.t.} \quad & Py + Nz \leq r - Kx^* \\
& y \in \mathbb{R}^{n_c}, \quad z \in \mathbb{B}^{n_d}
\end{aligned} \tag{D.77}$$

The optimal value is $\theta(x^*)$.

- 4: Solve an additional MILP to refine a solution that is favor of the leader

$$\begin{aligned}
\Theta(x^*) = \min \quad & g^T y + h^T z \\
\text{s.t.} \quad & w^T y + v^T z \geq \theta(x^*) \\
& Py + Nz \leq r - Kx^* \\
& y \in \mathbb{R}^{n_c}, \quad z \in \mathbb{B}^{n_d}
\end{aligned} \tag{D.78}$$

The optimal solution is (y^*, z^*) , and the optimal value is $\Theta(x^*)$. Update upper bound $UB = \min\{UB, f^T x^* + \Theta(x^*)\}$.

- 5: If $UB - LB = 0$, terminate and report optimal solution; otherwise, set $z^{l+1} = z^*$, create new variables (y^{l+1}, π^{l+1}) , adding the following cuts to master problem

$$\begin{aligned}
& w^T y^0 + v^T z^0 \geq w^T y^{l+1} + v^T z^{l+1} \\
& 0 \leq \pi^{l+1} \perp r - Kx - Nz^{l+1} - Py^{l+1} \geq 0, \quad P^T \pi^{l+1} = w
\end{aligned}$$

Update $l \leftarrow l + 1$, and go to step 2.

The second \geq follows from the fact that z^{l+1} is the optimal solution to problem (D.77) and KKT condition in constraints warrants that $v^T z^{l+1} + w^T y^{l+1} = \theta(x^*)$. In the next iteration, the algorithm terminates since $LB \geq UB$.

It should be pointed out that although a large amount of variables and constraints are generated in step 5, in practice, Algorithm D.5 often converges to an optimal solution within a small number of iterations that could be drastically smaller than the cardinality of Z , because the most critical scenarios in Z can be discovered from problem (D.77).

It is suggested in [53] that the master problem could be tightened by introducing variables $(\hat{y}, \hat{\pi})$ representing the primal and dual variables of lower-level problem corresponding to (x, z^0) and augmenting the following constraints

$$\begin{aligned} w^T y^0 + v^T z^0 &\geq w^T \hat{y} + v^T z^0 \\ 0 \leq \hat{\pi} \perp r - Kx - Nz^0 - P\hat{y} &\geq 0, \quad P^T \hat{\pi} = w \end{aligned}$$

It is believed that such constraints includes some useful information that is parametric not only to x but also to z^0 , and is not available from any fixed samples z^1, \dots, z^l . It is also pointed out that for instance with pure integer variables in the lower-level problem, this strategy is generally ineffective.

D.4 Mathematical Programs with Equilibrium Constraints

A mathematical program with equilibrium constraints (MPEC) is an extension of the bilevel program by incorporating multiple followers competing with each other, resulting in a GNEP in the lower level. In this regard, an MPEC is a single-leader-multi-follower Stackelberg game. In a broader sense, MPEC is an optimization problem with variational inequalities. MPECs are difficult to solve because of the complementarity constraints.

D.4.1 Mathematical Formulation

In an MPEC, the leader deploys its action x prior to the followers; then each follower selects its optimal decision y_j taking the decision of the leader x and rivals' strategies y_{-j} as given. The MPEC can be formulated in two levels:

$$\begin{aligned}
\text{Leader:} \quad & \begin{cases} \min_{x, \bar{y}, \bar{\lambda}, \bar{\mu}} & F(x, \bar{y}, \bar{\lambda}, \bar{\mu}) \\ \text{s.t.} & G(x, \bar{y}) \leq 0 \\ & (\bar{y}, \bar{\lambda}, \bar{\mu}) \in S(x) \end{cases} \quad (\text{D.79a}) \\
\text{Followers:} \quad & \begin{cases} \min_{y_j, \lambda_j, \mu_j} & f_j(x, y_j, y_{-j}) \\ \text{s.t.} & g_j(x, y_j) \leq 0 : \mu_j \\ & h(x, y) \leq 0 : \lambda_j \end{cases}, \quad \forall j \quad (\text{D.79b})
\end{aligned}$$

In (D.79a), the leader minimizes its payoff function F which depends on the choice of its own x , the decisions of the followers y , and the dual variables λ and μ from the lower level, because these dual variables may represent the prices of goods determined by the lower-level market clearing model. Constraints include inequalities and equalities (as a pair of opposite inequalities), as well as the optimality condition of the lower-level problem. In (D.79b), x is treated as a parameter, and the competition among followers comes down to a GNEP with shared convex constraints: the payoff function $f_j(x, y_j, y_{-j})$ of follower j is assumed to be convex in y_j ; inequality $g_j(x, y_j) \leq 0$ defines a local constraint of follower j which is convex in y_j and does not involve y_{-j} ; inequality $h(x, y) \leq 0$ is the shared constraint which is convex in y . Since each follower's problem is convex, the KKT condition is both necessary and sufficient for optimality. We assume that the set of GNEPs $S(x)$ is always non-empty.

The GNEP encompasses several special cases in the lower level. If the global constraint is absent, it degenerates into an NEP; moreover, if the objective functions of followers are also decoupled, the lower level reduces to independent convex optimization programs.

By replacing the lower-level GNEP with its KKT condition (D.35), the MPEC (D.79) becomes an MPCC, which can be solved by some suitable methods explained before. As the lower level GNEP usually possesses infinitely many equilibria, the outcome found by the MPCC reformulation is the favourite one from the leader's perspective. We can also require the Lagrange multipliers for the shared constraints should be equal, so as to restrict the GNEP to VEs. If the followers' problems are linear, the primal-dual optimality condition is an alternative choice in addition to the KKT condition, as it often involves fewer constraints. Nevertheless, the strong duality may introduce products involving primal and dual variables, such as those in (D.56) and (D.57), which remain non-convex and require special treatments.

D.4.2 Linear Complementarity Problem

A linear complementarity problem (LCP) requires finding a feasible solution subject to the following constraints

$$0 \leq x \perp Px + q \geq 0 \quad (\text{D.80})$$

where P is a square matrix; q is a vector. Their dimensions are compatible with x .

LCP is a special case of MPCC without an objective function. This type of problem frequently arises in various disciplines including market equilibrium analysis, computational mechanics, game theory, and mathematical programming. The theory of LCPs is a well-developed field. Detailed discussions can be found in [55]. In general, an LCP is NP-hard, although it is polynomially solvable for some special cases. One situation is when the matrix P is positive semidefinite. In such circumstance, problem (D.80) can be solved via the following convex quadratic program

$$\begin{aligned} \min \quad & x^T Px + q^T x \\ \text{s.t.} \quad & x \geq 0, \quad Px + q \geq 0 \end{aligned} \quad (\text{D.81})$$

(D.81) is a CQP which is readily solvable. Its optimum must be non-negative according to the constraints. If the optimal value of (D.81) is 0, then its optimal solution also solves LCP (D.80); otherwise, if the optimal value is strictly positive, LCP (D.80) is infeasible. In fact, this conclusion holds no matter whether P is positive semidefinite or not. However, if P is indefinite, identifying the global optimum of a non-convex QP (D.81) is also NP-hard, and thus does not facilitate solving the LCP.

There is a large body of literature discussing algorithms for solving LCPs. One of the most representative ones is the Lemke's pivoting method developed in [56], and another emblematic one is the interior-point method proposed in [57]. One drawback of the former method is its exponentially growing worst-case complexity, which makes it less efficient for large problems. The latter approach runs in polynomial time, but it requires the positive semidefiniteness of P , which is a strong assumption and limits its application. In this section, we will not present comprehensive reviews on the algorithms for LCP. We will introduce MILP formulations for problem (D.80) devised in [58, 59]. They make no reference on any special structure of matrix P . More importantly, they offer an option to access the solutions of practical problems in a systematic way.

Recall the MILP formulation techniques presented in Appendix B.3.5, it is easy to see that problem (D.80) can be equivalently expressed as linear constraints with additional binary variable z as follows

$$0 \leq x \leq Mz, \quad 0 \leq Px + q \leq M(1 - z) \quad (\text{D.82})$$

Integrality of z maintains the element-wise complementarity of x and $Px + q$: at most one of x_i and $(Px + q)_i$ can be strictly positive. Formulation (D.82) entails a manually specified parameter M , which is not instantly available at hand. On the one hand, it must be big enough to preserve all extreme points of (D.80). On the other hand, it is expected to be as small as possible from

a computational perspective, otherwise, the continuous relaxation of (D.82) would be very loose. In this regard, (D.82) is too cursory, although it might work well.

To circumvent above difficulty, it is proposed in [58] to solve a bilinear program without a big-M parameter

$$\begin{aligned} \min_{x,z} \quad & z^T(Px + q) + (\mathbf{1} - z)^T x \\ \text{s.t.} \quad & x \geq 0, Px + q \geq 0, z \text{ binary} \end{aligned} \quad (\text{D.83})$$

If (D.80) has a solution x^* , the optimal value of (D.83) is 0: for $x_i^* > 0$, we have $z_i^* = 1$ and $(Px^* + q)_i = 0$; for $(Px^* + q)_i > 0$, we have $z_i^* = 0$ and $x_i^* = 0$. The optimal solution is consistent with the feasible solution of (D.82). The objective can be linearized by introducing auxiliary variables $w_{ij} = z_i x_j$, $\forall i, j$. However, applying normal integer formulation techniques in Appendix B.2.2 on variable w_{ij} again needs the upper bound of x_i , another interpretation of the big-M parameter.

A parameter-free MILP formulation is suggested in [58]. To understand the basic idea, recall the fact that $(1 - z_i)x_i = 0$; if we impose $x_i = w_{ii} = x_i z_i$, $i = 1, 2, \dots$ in the constraint, $(\mathbf{1} - z)^T x$ in the objective can be omitted. Furthermore, multiplying both sides of $\sum_j P_{kj}x_j + q_k \geq 0$, $k = 1, 2, \dots$ with z_i gives $\sum_j P_{kj}w_{ij} + q_k z_i \geq 0$, $\forall i, k$. Since $z_i \in \{0, 1\}$, $\sum_j P_{kj}x_j + q_k \geq \sum_j P_{kj}w_{ij} + q_k z_i$, $\forall i, k$ and $0 \leq w_{ij} \leq x_j$, $\forall i, j$ naturally hold. Collecting up these valid inequalities, we obtain an MILP

$$\begin{aligned} \min_{x,z,w} \quad & q^T z + \sum_i \sum_j P_{ij} w_{ij} \\ \text{s.t.} \quad & \sum_j P_{kj} x_j + q_k \geq \sum_j P_{kj} w_{ij} + q_k z_i, \forall i, k \\ & 0 \leq w_{ij} \leq x_j, \forall i, j, w_{jj} = x_j, \forall j, z \text{ binary} \end{aligned} \quad (\text{D.84})$$

Instead of enforcing every $\sum_j P_{kj}w_{ij} + q_k z_i$ being at 0, we relax them as inequalities and minimize their summation. More valid inequalities can be added in (D.84) by exploiting linear cuts of z . It is proved in [58] that relation $w_{ij} = z_i x_j$, $\forall i, j$ is implicitly guaranteed at the optimal solution of (D.84). In view of this, MILP (D.84) is equivalent to LCP (D.80) in the following sense: (D.80) has a solution if and only if (D.84) has an optimal value equal to zero, and the optimal solution to (D.84) incurring a zero objective value is a solution of LCP (D.80). MILP (D.84) is superior compared with (D.82) and big-M linearization based MILP formulation of MINLP (D.83) because it is parameter-free and gives tighter continuous relaxation. Nevertheless, the number of constraints in (D.84) is significantly larger than that in formulation (D.82). This method has been further analyzed in [59] and extended to binary-constrained mixed LCPs.

Another parameter-free MILP formulation is suggested in [60], which takes the form of

$$\begin{aligned}
 & \max_{\alpha, y, z} \quad \alpha \\
 & \text{s.t.} \quad 0 \leq (Py)_i + q_i \alpha \leq 1 - z_i, \quad \forall i \\
 & \quad \quad 0 \leq y_i \leq z_i, \quad z_i \in \{0, 1\}, \quad \forall i \\
 & \quad \quad 0 \leq \alpha \leq 1
 \end{aligned} \tag{D.85}$$

Since $\alpha = 0$, $y = 0$, $z = 0$ is always feasible, MILP (D.85) is feasible and has an optimum no greater than 1. By observing the constraints, we can conclude that if MILP (D.85) has a feasible solution with $\bar{\alpha} > 0$, then $x = y/\bar{\alpha}$ solves problem (D.80). If the optimal solution $\bar{\alpha} = 0$, then problem (D.80) has no solution; otherwise, suppose \bar{x} solves (D.80), and let $\bar{\alpha}^{-1} = \max\{\bar{x}_i, (P\bar{x})_i + q_i, i = 1, \dots\}$, then for any $0 < \alpha \leq \bar{\alpha}$, $\bar{y} = \alpha\bar{x}$ is feasible in (D.85). As a result, the optimal solution should be no less than $\bar{\alpha}$, rather than 0. Compared with formulation (D.82), the big-M parameter is adaptively scaled by optimizing α .

Because (D.85) works with an intermediate variable y , when LCP (D.80) should be jointly solved with other conditions on x , formulation (D.85) is not advantageous, because non-convex variable transformation $x = y/\alpha$ must be appended to link both parts.

Robust solutions of LCPs with uncertain P and q are discussed in [61]. It is found that when $P \succeq 0$, robust solutions can be extracted from an SOCP under some mild assumptions on the uncertainty set; otherwise, the more general problem with uncertainty can be reduced to a deterministic non-convex QCQP. This technique is particularly useful in uncertain traffic equilibrium problems and uncertain Nash-Cournot games. Uncertain VI problems and MPCCs can be tackled in the similar vein after some proper transformations.

It is shown in [62] that a linear bilevel program or its equivalent MPEC can be globally solved via a sequential LCP method. A hybrid enumerative method is suggested which substantially reduces the effort for searching a solution of the LCP or certifying that the LCP has no solution. When the LCP is easy to solve, this approach is attractive.

Several extensions of LCP, including the discretely-constrained mixed LCP, discretely-constrained Nash-Cournot game, discretely-constrained MPEC, and logic constrained equilibrium problem as well as their applications in energy markets and traffic system equilibrium have been investigated in [63, 64, 65, 66]. In a word, due to its wide applications, LCP is still an active research field, and MILP remains an appealing method for solving LCPs for practical problems.

D.4.3 Linear Programs with Complementarity Constraints

A linear program with complementarity constraints (LPCC) entails solving a linear optimization problem with linear complementarity constraints. It is a special case of MPCC if all functions in the problem are linear, and a generalization of LCP by incorporating an objective function to be optimized. An LPCC has the following form

$$\begin{aligned} \max_{x,y} \quad & c^T x + d^T y \\ \text{s.t.} \quad & Ax + By \geq f \\ & 0 \leq y \perp q + Nx + My \geq 0 \end{aligned} \tag{D.86}$$

A standard approach for solving (D.86) is to linearize the complementarity constraint by introducing a binary vector z and solve the following MILP

$$\begin{aligned} \max_{x,y,z} \quad & c^T x + d^T y \\ \text{s.t.} \quad & Ax + By \geq f \\ & 0 \leq q + Nx + My \leq Mz \\ & 0 \leq y \leq M(1 - z) \\ & z \in \{0, 1\}^m \end{aligned} \tag{D.87}$$

If both of x and y are bounded variables, we can readily derive the proper value of M in each inequality; otherwise, finding high quality bounds is non-trivial even if they do exist. The method in [46] can be used to determine proper bounds of M , if the NLP solver can successfully find local solutions of the bounding problems.

Using an arbitrarily large value may solve the problem correctly. Nevertheless, parameter-free method is still of great theoretical interests. A smart Benders decomposition algorithm is proposed in [67] to solve (D.87) without requiring the value of M . The completely positive programming method developed in [68] can also be used to solve (D.87). For more theory and algorithm for LPCC, please see [45], [69]-[75] and references therein. Interesting connections among conic QPCCs, QCQPs, and completely positive programs are revealed in [76].

D.5 Equilibrium Programs with Equilibrium Constraints

An equilibrium program with equilibrium constraints (EPEC) is the most general extension of the bilevel program. It incorporates multiple leaders and

multiple followers competing with each other in the upper level and the lower level, respectively, resulting in two GNEPs in both levels. In this regard, an EPEC is a multi-leader-follower Stackelberg game.

D.5.1 Mathematical model

In an EPEC, each leader i deploys an action x_i prior to the followers while taking movements of other leaders x_{-i} into account and anticipating the best responses $y(x)$ from the followers; then each follower selects its optimal decision y_j by taking the strategies of leaders x and rivals' actions y_{-j} as given. The EPEC can be formulated in two levels

$$\text{Leaders: } \left\{ \begin{array}{l} \min_{x_i, \bar{y}, \bar{\lambda}, \bar{\mu}} F_i(x_i, x_{-i}, \bar{y}, \bar{\lambda}, \bar{\mu}) \\ \text{s.t. } G_i(x_i) \leq 0 \\ (\bar{y}, \bar{\lambda}, \bar{\mu}) \in S(x_i, x_{-i}) \end{array} \right\}, \forall i \quad (\text{D.88a})$$

$$\text{Followers: } \left\{ \begin{array}{l} \min_{y_j} f_j(x, y_j, y_{-j}) \\ \text{s.t. } g_j(x, y_j) \leq 0 : \mu_j \\ h(x, y) \leq 0 : \lambda_j \end{array} \right\}, \forall j \quad (\text{D.88b})$$

In (D.88a), each leader minimizes its payoff function F_i which depends on its own choice x_i , the decisions of followers y , dual variables λ and μ are parameterized in competitors' strategies x_{-i} . Tuple $(\bar{y}, \bar{\lambda}, \bar{\mu})$ in the upper level is restricted by the optimality condition of the lower-level problem. Although the inequality constraints of leaders are decoupled, and we do not explicitly consider global constraints in the upper-level GNEP, the leaders' strategy sets as well as their payoff functions are still correlated through the best reaction map $S(x_i, x_{-i})$, and hence (D.88a) itself is a GNEP, which is non-convex. The followers' problem (D.88b) is a GNEP with shared constraints, which is the same as the situation in an MPEC. The same convexity assumptions are made in (D.88b). The structure of EPEC (D.88) is depicted in Fig. D.4. The equilibrium solution of EPEC (D.88) is defined as the GNE among leaders' MPECs. It is common knowledge that EPECs often have no pure strategy equilibrium due to the intrinsic non-convexity of MPECs.

D.5.2 Methods for Solving an EPEC

An EPEC can be viewed as a set of coupled MPEC problems: leader i is facing an MPEC composed of problem i in (D.88a) together with all followers' problems in (D.88b), which is parameterized in x_{-i} . By replacing lower-level

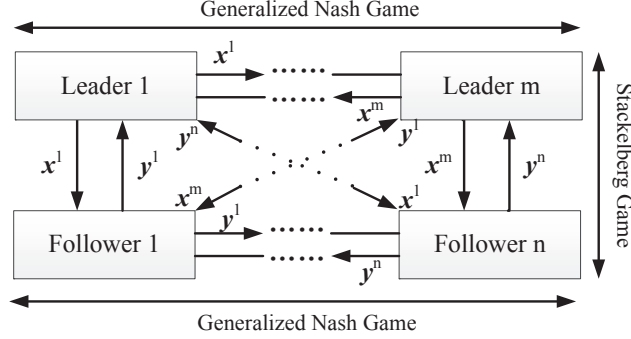


Fig. D.4 The structure of an EPEC.

GNEP with its KKT optimality conditions, it can be imaged that the GNEP among leaders have non-convex constraints which inherit the tough properties of complementarity constraints. Thus, solving an EPEC is usually extremely challenging. To our knowledge, systematic algorithms of EPEC are firstly developed in dissertations [77, 78, 79]. The primary application of such an equilibrium model is found in energy market problems, see [80] for an excellent introduction.

Unlike the NEP and GNEP discussed in Sect. D.1 and Sect. D.2, where the strategy sets are convex or jointly convex, because the lower-level problems are replaced with KKT optimality conditions and the MPEC for the leader is intrinsically non-convex, provable existence and uniqueness guarantees for the solution to EPECs are non-trivial. There is sustainable attempt on the analysis of EPEC solution properties. For example, the existence of a unique equilibrium for certain EPEC instances are discussed in [81, 82], and in [83] for a nodal price based power market model. However, the existence and uniqueness of solution are only guaranteed under restrictive conditions. Counterexamples have been given in [84] to demonstrate that there is no general result for the existence of solutions to EPECs due to their non-convexity. The non-uniqueness issue is studied in [78, 85]. It is shown that even in the simplest instances, local uniqueness of the EPEC equilibrium solution may not be guaranteed, and a manifold of equilibria may exist. This can be understood because EPEC is a generalization of GNEP, whose solution property is illustrated in Sect. D.2.1. When the payoff functions possess special structures, say, a potential function exists, then the existence of a global equilibrium can be investigated using the theory of potential games [86, 87, 88]. In summary, the theory of EPEC solutions are much more complicated than the single-level NEP and GNEP.

This section reviews several representative algorithms which are widely used in literature. The former two are generic and seen in [77, 78]; the third one is motivated by the convenience brought by the property of potential

games, and reported in [88]; at last, a pricing game in a competitive market, which appears to be non-convex at first sight, is presented to show the hidden convexity in such a special equilibrium model.

1. Best response algorithm

Since the equilibrium of an EPEC is a GNEP among leaders' MPEC problems, the most intuitive strategy for identifying an equilibrium solution is the best response algorithm. In some literature, it is also called diagonalization method or sequential MPEC method. This approach can be further categorized into Jacobian type and Gauss-Seidel type method, according to the information used when players update their strategies.

To explain the algorithmic details, denote by $\text{MPEC}(i)$ the problem of leader i : the upper level is problem i in (D.88a), and the lower level is the GNEP described in (D.88b) given all leaders' strategies. Let x_i^k be the strategy of leader i in iteration k , and $x^k = (x_1^k, \dots, x_m^k)$ the strategy profile of leaders. The Gauss-Seidel type algorithm proceeds as follows [89]:

Algorithm D.6 : Best-response (Diagonalization) algorithm for EPEC

- 1: Choose an initial strategy profile x^0 for leaders, set convergence tolerance $\varepsilon > 0$, an allowed number of iterations K , and the iteration index $k = 0$;
 - 2: Let $x^{k+1} = x^k$. Loop for players $i = 1, \dots, m$:
 - a. Solve $\text{MPEC}(i)$ for leader i given x_{-i}^{k+1} .
 - b. Replace x_i^{k+1} with the optimal strategy of leader i just obtained.
 - 3: If $\|x^{k+1} - x^k\|_2 \leq \varepsilon$, the upper level converges; solve lower-level GNEP (D.88b) with $x^* = x^{k+1}$ using the algorithms elaborated in Sect. D.2.2, and the equilibrium among followers is y^* . Report (x^*, y^*) and terminate.
 - 4: If $k = K$, report failure of convergence and quit.
 - 5: Update $k \leftarrow k + 1$, and go to step 2.
-

Without an executable criterion to judge the existence and uniqueness of solution, possible outcomes of Algorithm D.6 are discussed in three situations.

1. There is no equilibrium. Algorithm D.6 does not converge. In such circumstance, one may turn to seeking a mixed-strategy Nash equilibrium, which always exists. Examples are given in [89]: if there are two leaders, we can list possible strategy combinations and solve the lower-level GNEP among followers, then compute respective payoffs of the two leaders, and then build a bimatrix game, whose mixed-strategy Nash equilibrium can be calculated from solving an LCP, as explained in Sect. D.1.4.

2. There is a unique equilibrium, or there are multiple equilibria. Algorithm D.6 may converge or not, and which equilibrium will be found (if it converges) depends on the initial strategy profile offered in step 1.

3. Algorithm D.6 may converge to a local equilibrium in the sense of [83], if each MPEC is solved by a local NLP method which does not guarantee global optimality. The true equilibrium can be found only if each leader's MPEC can be globally solved. The MILP reformulation (if possible) offers one plausible way for this task.

2. KKT system method

To tackle the divergence issue in the best response algorithm, it is proposed to apply the KKT condition to each leader's MPEC and solve the resulting KKT systems simultaneously [79, 90]. The solution turns out to be a strong stationary equilibrium point of EPEC (D.88). There is no convergence issue in this approach, since no iteration is deployed. However, special attention should be paid to some potential problems mentioned below.

1. Since the EPEC is essentially a GNEP among leaders, the concentrated KKT system may have non-isolated solutions. To refine a meaningful outcome, we can manually specify a secondary objective function, which is optimized subject to the KKT system.

2. The embedded (twice) application of KKT condition for the lower-level problems and upper-level problems inevitably introduces extensive of complementarity and slackness conditions, which greatly challenges solving the concentrated KKT system. In this regard, scalability may be a main bottleneck for this approach. If the lower-level GNEP is linear, it may be better to use primal-dual optimality condition first for followers, and then KKT condition for leaders.

3. Because each leader's MPEC is non-convex, a stationary point of the KKT condition is not necessarily an optimal solution of the leader; as a result, the solution of the concentrated KKT system may not be an equilibrium of the EPEC. To validate the result, one can conduct the best-response method initiated at the candidate solution with a slight perturbation.

3. Potential MPEC method

When the upper-level problems among leaders admit a potential function satisfying (D.25), the EPEC can be reformulated as an MPEC, and the relations of their solutions are revealed by comparing the KKT condition of the normalized Nash stationary points of the EPEC and the KKT condition of the associated MPEC [88].

For example, if the leaders' objectives are given by

$$F_i(x_i, x_{-i}, y) = F_i^S(x_i) + H(x, y)$$

or in other words, the payoff function $F_i(x_i, x_{-i}, y)$ can be decomposed as the sum of two parts: the first one $F_i^S(x_i)$ only depends on the local variable x_i , and the second one $H(x, y)$ is common to all leaders. In such circumstance, the potential function can be expressed as

$$U(x, y) = H(x, y) + \sum_{i=1}^m F_i^S(x_i)$$

Please see Sect. D.1.5 for the condition under which a potential function exists and special instances in which a potential function can be easily found.

Suppose that leaders' local constraints are given by $x_i \in X_i$ which is independent of x_{-i} and y , and the best reaction map of followers with fixed x is given by $(\bar{y}, \bar{\lambda}, \bar{\mu}) \in S(x)$. Clearly, the solution of MPEC

$$\begin{aligned} \min_{x, \bar{y}, \bar{\lambda}, \bar{\mu}} \quad & U(x, \bar{y}) \\ \text{s.t.} \quad & x_i \in X_i, \quad i = 1, \dots, m \\ & (\bar{y}, \bar{\lambda}, \bar{\mu}) \in S(x) \end{aligned}$$

must be an equilibrium solution of the original EPEC.

This approach leverages the property of potential games and is superior over the previous two methods (if a potential function exists): KKT condition is applied only once to the lower level problems, and the equilibrium can be retrieved by solving MPEC only once.

4. A pricing game in a competitive market

We consider an EPEC taken from the examples in [91], which models a strategic pricing game in a competitive market. The hidden convexity in this EPEC is revealed. For ease of exposition, we study the case with two leaders and one follower. The results can be extended to the situation where more than two leaders exist. The pricing game with two leaders can be formulated by the following EPEC

$$\text{Leader 1:} \quad \max_{x_1} \{y^T(x_1, x_2)A_1x_1 \mid B_1x_1 \leq b_1\} \quad (\text{D.89a})$$

$$\text{Leader 2:} \quad \max_{x_2} \{y^T(x_1, x_2)A_2x_2 \mid B_2x_2 \leq b_2\} \quad (\text{D.89b})$$

$$\text{Follower:} \quad \max_y \{f(y) - y^T A_1x_1 - y^T A_2x_2 \mid Cy = d\} \quad (\text{D.89c})$$

In (D.89a) and (D.89b), two leaders announce their offering prices x_1 and x_2 , respectively, subject to some certain pricing policy described in their corresponding constraints. The follower then decides how many goods should be purchased from each leader, according to the optimal solution of problem (D.89c), where the profit of the follower

$$f(y) = -\frac{1}{2}y^T Qy + c^T y$$

is a strongly concave quadratic function, i.e. $Q \succ 0$, and matrix C has full rank in its rows. Each player in the market wishes to maximize his own profit.

The utilities of leaders are the payments from trading with the follower; the profit of follower is the revenue minus the purchasing cost.

At first sight, EPEC (D.89) is non-convex, not only because the leaders' objective functions are bilinear, but also because the best response mapping is generally non-convex. In light of the strong convexity of (D.89c), the following KKT condition:

$$\begin{aligned} c - Qy - A_1x_1 - A_2x_2 - C^T\lambda &= 0 \\ Cy - d &= 0 \end{aligned}$$

is necessary and sufficient for a global optimum. Because constraints in (D.89c) are all equalities, there is no complementarity and slackness condition. Solve this set of linear equations, we can obtain the optimal solution y in a closed form. To this end, substituting

$$y = Q^{-1}(c - A_1x_1 - A_2x_2 - C^T\lambda)$$

into the second equation, we have

$$\lambda = M[N(c - A_1x_1 - A_2x_2) - d]$$

where

$$M = [CQ^{-1}C^T]^{-1}, \quad N = CQ^{-1}$$

Moreover, eliminating λ in the expression of y gives the best reaction map

$$y = r + D_1x_1 + D_2x_2 \tag{D.90}$$

where

$$\begin{aligned} r &= Q^{-1}c + N^T M d - N^T M N c \\ D_1 &= N^T M N A_1 - Q^{-1}A_1 \\ D_2 &= N^T M N A_2 - Q^{-1}A_2 \end{aligned}$$

Substituting (D.90) into the objective functions of leaders, EPEC (D.89) reduces to a standard Nash game

$$\begin{aligned} \text{Leader 1: } & \max_{x_1} \{ \theta_1(x_1, x_2) \mid B_1x_1 \leq b_1 \} \\ \text{Leader 2: } & \max_{x_2} \{ \theta_2(x_1, x_2) \mid B_2x_2 \leq b_2 \} \end{aligned}$$

where

$$\begin{aligned} \theta_1(x_1, x_2) &= r^T A_1x_1 + x_1^T D_1^T A_1x_1 + x_2^T D_2^T A_1x_1 \\ \theta_2(x_1, x_2) &= r^T A_2x_2 + x_2^T D_2^T A_2x_2 + x_1^T D_1^T A_2x_2 \end{aligned}$$

The partial Hessian matrix of $\theta_1(x_1, x_2)$ can be calculated as

$$\nabla_{x_1}^2 \theta_1(x_1, x_2) = 2A_1^T (N^T M N - Q^{-1}) A_1$$

As $Q \succ 0$, its inverse matrix $Q^{-1} \succ 0$; denote by $Q^{-1/2}$ the square root of Q^{-1} , and

$$P_J = I - Q^{-1/2} C^T (C Q^{-1} C^T)^{-1} C Q^{-1/2}$$

It is easy to check that P_J is a projection matrix, which is symmetric and idempotent, i.e., $P_J = P_J^2 = P_J^3 = \dots$. Moreover, it can be verified that the Hessian matrix $\nabla_{x_1}^2 \theta_1(x_1, x_2)$ can be expressed via

$$\nabla_{x_1}^2 \theta_1(x_1, x_2) = 2A_1^T (N^T M N - Q^{-1}) A_1 = -2A_1^T Q^{-1/2} P_J Q^{-1/2} A_1$$

For any vector z with a proper dimension,

$$\begin{aligned} z^T \nabla_{x_1}^2 \theta_1(x_1, x_2) z &= -2z^T \left(A_1^T Q^{-1/2} P_J Q^{-1/2} A_1 \right) z \\ &= -2z^T A_1^T Q^{-1/2} P_J^T P_J Q^{-1/2} A_1 z \\ &= -2(P_J Q^{-1/2} A_1 z)^T (P_J Q^{-1/2} A_1 z) \leq 0 \end{aligned}$$

We can see that $\nabla_{x_1}^2 \theta_1(x_1, x_2) \preceq 0$. The similar analysis also applies to $\nabla_{x_2}^2 \theta_2(x_1, x_2)$. Therefore, the problems of leaders are actually convex programs, and a pure-strategy Nash equilibrium exists.

D.6 Conclusions and Further Reading

Equilibrium problems entail solving interactive optimization problems simultaneously, and serve as the foundation for modeling competitive behaviors among strategic decision makers, and analyzing the stable outcome of a game. This chapter provides an overview on two kinds of equilibrium problems that frequently arise in various economic and engineering applications.

One-level equilibrium problems, including the NEP and GNEP, are introduced first. The existence of equilibrium can be ensured under some convexity and monotonicity assumptions. Distributed methods for solving one-level games are presented. When each player solves a strictly convex optimization problem, distributed algorithms converge with provable guarantee, and thus are preferred, whereas the KKT system renders nonlinear equations and is relatively difficult to solve. To address incomplete information and uncertainty in player's decision making, a robust optimization based game model is proposed in [92], which is distribution-free and relaxes Harsanyi's assumptions on Bayesian games. Particularly, the robust Nash equilibrium of a bimatrix game with uncertain payoffs can be characterized via the solution of a second-order cone complementarity problem [93], and more general cases involving n players and continuous payoffs are discussed in [94]. Distributional uncertainty is tackled in [95], in which the mixed-strategy Nash equilibrium of a distributionally robust chance-constrained game is studied. A generalized Nash game arises when the strategy sets of players are coupled. Due to prac-

tical interests from a variety of engineering disciplines, the solution method for GNEPs is still an active research area. The volume of articles is growing quickly in recent years, say, [96, 97, 98, 99, 100, 101, 102, 103, 104], to name just a few. GNEPs with uncertainties are studied in [105, 106].

Bilevel equilibrium problems, including the bilevel program, MPEC, and EPEC, are investigated. These problems are intrinsically hard to solve, due to the non-convexity induced by the best reaction map of followers, and solution properties have been revealed for specific instances under restrictive assumptions. We recommend [89, 107, 108] for theoretical foundations and energy market applications of bilevel equilibrium models, and [109] for an up-to-date survey. The theories on bilevel programs and MPEC are relatively mature. Recent research efforts have been spent on new constraint qualifications and optimality conditions, for example, the work in [110, 111, 112, 113]. The MILP reformulation is preferred by most power system applications, because the ability of MILP solvers keep improving, and a global optimal solution can be found. Stochastic MPEC is proposed in [114] to model uncertainty using probability distributions. Algorithms are developed in [115, 116, 117, 118], and a literature review can be found in [119]. Owing to the inherent hardness, discussions on EPEC models are limited to special cases, such as those with shared P-matrix linear complementarity constraints [120], power market models [120, 121, 122], those with convex quadratic objectives and linear constraints [120], and Markov game models [123]. Methods for solving EPEC are based on relaxing or regularizing complementarity constraints [123, 124], as well as evolutionary algorithms [125]. Robust equilibria of EPEC are discussed in [126]. An interesting connection between the bilevel program and the GNEP has been revealed in [127], establishing a new look on these game models.

We believe that the equilibrium programming models will become an imperative tool for designing and analyzing interconnected energy systems and related markets, in view of the physical interdependence of heterogeneous energy flows and strategic interactions among different network operators.

References

1. Nash, J.: Equilibrium points in n -person games. *Proc. Natl. Acad. Sci.* 36(1): 48-49 (1950)
2. Nash, J.: Non-cooperative games. *Ann. Math.* 54(2): 286-295 (1951)
3. Granas, A., Dugundji, J.: *Fixed Point Theory*. Springer New York, NY, USA (2003)
4. Dreves, A., Facchinei, F., Kanzow, C., Sagratella, S.: On the solution of the KKT conditions of generalized Nash equilibrium problems. *SIAM J. Optimiz.* 21(3): 1082-1108 (2011)
5. Scutari, G., Palomar, D.P., Facchinei, F., Pang, J.S.: Monotone games for cognitive radio systems. In: *Distributed Decision Making and Control*. Springer, London UK 83-112 (2012)

6. Facchinei, F., Pang, J.S.: Finite-Dimensional Variational Inequalities and Complementarity Problem. Springer, New York, NY, USA (2003)
7. Scutari, G., Palomar, D.P., Facchinei, F., Pang, J.S.: Convex optimization, game theory, and variational inequality theory. *IEEE Signal Proc. Mag.* 27(3): 35-49 (2010)
8. Ortega, J.M., Rheinboldt, W.C.: Iterative Solution of Nonlinear Equations in Several Variables. SIAM, Philadelphia, PA (1987)
9. Scutari, G., Palomar, D.P., Barbarossa, S.: Asynchronous iterative water-filling for Gaussian frequency-selective interference channels. *IEEE Trans. Inform. Theory* 54(7): 2868-2878 (2008)
10. Bertsekas, D.P., Tsitsiklis, J.N.: Parallel and Distributed Computation: Numerical Methods, 2nd Ed. Athena Scientific Press, Belmont, MA (1989)
11. Kannan, A., and Shanbhag, U.V.: Distributed computation of equilibria in monotone Nash games via iterative regularization techniques. *SIAM J. Optimiz.* 22(4): 1177-1205 (2012)
12. Dantzig, G.B.: Linear Programming and Extensions. Princeton University Press, Princeton, New Jersey (1963)
13. Lemke, C.E., Howson, J.T.: Equilibrium points of bimatrix games, *SIAM J. Appl. Math.* 12(2): 413-423 (1964)
14. Lemke, C.E.: Bimatrix equilibrium points and mathematical programming, *Management Sci.* 11(7): 681-689 (1965)
15. Monderer, D., Shapley, L.S.: Potential games. *Games Econ. Behav.* 14(1): 124-143 (1996)
16. Lă, Q.D., Chew, Y.H., Soong, B.H.: Potential Game Theory. Springer, Switzerland (2016)
17. Harker, P.: Generalized Nash games and quasi-variational inequalities. *Eur. J. Oper. Res.* 54(1): 81-94. (1991)
18. Facchinei, F., Kanzow, C.: Generalized Nash equilibrium problems. *A Quarterly Journal of Operations Research (4OR)* 5(3): 173-210 (2007)
19. Fukushima, M.: Restricted generalized Nash equilibria and controlled penalty algorithm. *Comput. Manag. Sci.* 8(3): 201-218 (2011)
20. Rosen J.B.: Existence and uniqueness of equilibrium points for concave N-person games. *Econometrica* 33(3): 520-534 (1965)
21. Facchinei, F., Fischer, A., Piccialli, V.: On generalized Nash games and variational inequalities. *Oper. Res. Lett.* 35(2): 159-164 (2007)
22. Facchinei, F., Sagratella, S.: On the computation of all solutions of jointly convex generalized Nash equilibrium problems. *Optim. Lett.* 5(3): 531-547 (2011)
23. Schwartz, A. Mathematical programs with complementarity constraints: Theory, methods and applications. Ph.D. thesis, Institute of Applied Mathematics and Statistics, University of Würzburg (2011)
24. Guo, L., Lin, G.H.: Notes on some constraint qualifications for mathematical programs with equilibrium constraints. *J. Optim. Theory Appl.* 156(3): 600-616 (2013)
25. Ye, J.: Constraint qualifications and KKT conditions for bilevel programming problems. *Math. Oper. Res.* 31(4): 811-824 (2006)
26. Guo, L., Lin, G.H., Jane, J.Y.: Second-order optimality conditions for mathematical programs with equilibrium constraints. *J. Optim. Theory Appl.* 158(1): 33-64 (2013)
27. Ralph, D.: Mathematical programs with complementarity constraints in traffic and telecommunications networks. *Philos. Trans. R. Soc. A* 366(1872):1973-1987 (2008)
28. Scholtes, S.: Convergence properties of a regularization scheme for mathematical programs with complementarity constraints. *SIAM J. Optimiz.* 11(4): 918-936 (2001)

29. Ralph, D., Wright, S.J.: Some properties of regularization and penalization schemes for MPECs. *Optim. Method. Softw.* 19(5): 527-556 (2004)
30. Hoheisel, T., Kanzow, C., Schwartz, A.: Theoretical and numerical comparison of relaxation methods for mathematical programs with complementarity constraints. *Math. Program.* 137(1): 257-288 (2013)
31. Hu, X., Ralph, D.: Convergence of a penalty method for mathematical programming with complementarity constraints. *J. Optim. Theory Appl.* 123(2): 365-390 (2004)
32. Lin, G.H., Fukushima, M.: Some exact penalty results for nonlinear programs and mathematical programs with equilibrium constraints. *J. Optim. Theory Appl.* 118(1): 67-80 (2003)
33. Leyffer S, López-Calva G, Nocedal J. Interior methods for mathematical programs with complementarity constraints. *SIAM J. Optimiz.* 17(1): 52-77 (2006)
34. Facchinei, F., Jiang, H., Qi, L.: A smoothing method for mathematical programs with equilibrium constraints. *Math. Program.* 85(1): 107-134 (1999)
35. Fukushima, M., Pang, J.S.: Convergence of a smoothing continuation method for mathematical programs with complementarity constraints. In: *Ill-posed Variational Problems and Regularization Techniques*. Springer Berlin, Heidelberg 99-110 (1999)
36. Kanzow, C.: Some noninterior continuation methods for linear complementarity problems. *SIAM J. Matrix Anal. Appl.* 17(4): 851-868 (1996)
37. Fukushima, M., Luo, Z.Q., Pang, J.S.: A globally convergent sequential quadratic programming algorithm for mathematical programs with linear complementarity constraints. *Comput. Optim. Appl.* 10(1): 5-34 (1998)
38. Fletcher, R., Leyffer, S., Ralph, D., Scholtes, S.: Local convergence of SQP methods for mathematical programs with equilibrium constraints. *SIAM J. Optimiz.* 17(1): 259-286 (2006)
39. Fletcher, R., Leyffer, S.: Solving mathematical programs with complementarity constraints as nonlinear programs. *Optim. Method. Softw.* 19(1): 15-40 (2004)
40. Fampa, M.H.C., Melo, W.A.X., Maculan, N.: Semidefinite relaxation for linear programs with equilibrium constraints. *Int. Trans. Oper. Res.* 20(2): 201-212 (2013)
41. Jeyakumar, V., Lasserre, J.B., Li, G., Pham, T.S.: Convergent semidefinite programming relaxations for global bilevel polynomial optimization problems. *SIAM J. Optimiz.* 26(1): 753-780 (2016)
42. Nie, J., Wang, L., Ye, J.: Bilevel polynomial programs and semidefinite relaxation methods. *SIAM J. Optimiz.* 27(3): 1728-1757 (2017)
43. Fampa, M., Pimentel, W.: Linear programming relaxations for a strategic pricing problem in electricity markets. *Int. Trans. Oper. Res.* 24(1-2): 159-172 (2017)
44. Ghamkhari, M., Sadeghi-Mobarakeh, A., Mohsenian-Rad, H.: Strategic bidding for producers in nodal electricity markets: A convex relaxation approach. *IEEE Trans. Power Syst.* 32(3): 2324-2336 (2017)
45. Jara-Moroni, F., Pang, J.S., Wächter A.: A study of the difference-of-convex approach for solving linear programs with complementarity constraints. *Math. Program.* 169(1): 221-254 (2018)
46. Pineda, S., Bylling, H., Morales, J.M.: Efficiently solving linear bilevel programming problems using off-the-shelf optimization software. *Optim. Eng.* 19(1): 187-211 (2018)
47. White, D.J., Anandalingam, G.: A penalty function approach for solving bi-level linear programs. *J. Global Optim.* 3(4): 397-419 (1993)
48. Lv, Y., Hu, T., Wang, G., Wan, Z.: A penalty function method based on Kuhn-Tucker condition for solving linear bilevel programming. *Appl. Math. Comput.* 188(1): 808-813 (2007)

49. Wan, Z., Wang, G., Lv, Y.: A dual-relax penalty function approach for solving nonlinear bilevel programming with linear lower level problem. *Acta Math. Sci.* 31(2): 652-660 (2011)
50. Zare, M., Borrero, J., Zeng, B., Prokopyev, O.: A note on linearized reformulations for a class of bilevel linear integer problems. *Ann. Oper. Res.* 1-19 (2017)
51. Falk, J.: A linear max-min problem. *Math. Program.* 5(1): 169-188 (1973)
52. Haghighat, H., Zeng, B.: Bi-level mixed integer transmission planning. *IEEE Trans. Power Syst.* 33(6): 7309-7312 (2018)
53. Zeng, B., An, Y.: Solving bilevel mixed integer program by reformulations and decomposition. *Optimization online* (2014)
54. Yue, D., Gao, J., Zeng, B., You, F.: A projection-based reformulation and decomposition algorithm for global optimization of a class of mixed integer bilevel linear programs. *J. Global Optim.* (2018)
55. Cottle, R.W., Pang, J.S., Stone, R.E.: *The linear complementarity problem.* Society for Industrial and Applied Mathematics, Philadelphia (2009)
56. Lemke, C.E.: Bimatrix equilibrium points and mathematical programming. *Manage. Sci.* 11(7):681-689 (1965)
57. Kojima, M., Mizuno, S., Yoshise, A.: A polynomial time algorithm for a class of linear complementarity problems. *Math. Program.* 44(1): 1-26, (1988)
58. Sherali, H.D., Krishnamurthy, R.S., Al-Khayyal, F.A.: Enumeration approach for linear complementarity problems based on a reformulation-linearization technique. *J. Optim. Theory Appl.* 99(2): 481-507 (1998)
59. Fomeni, F.D., Gabriel, S.A., Anjos, M.F.: An RLT approach for solving the binary-constrained mixed linear complementarity problem. *Tech. Rep. G-2015-60*, GERAD-HEC Montréal 2015
60. Pardalos, P.M., Rosen, J.B.: Global optimization approach to the linear complementarity problem. *SIAM J. Sci. Stat. Comp.* 9(2): 341-353 (1988)
61. Xie, Y., Shanbhag, U.V.: On robust solutions to uncertain linear complementarity problems and their variants. *SIAM J. Optimiz.* 26(4): 2120-2159 (2016)
62. Júdice, J.J., Faustino, A.M.: A sequential LCP method for bilevel linear programming. *Ann. Oper. Res.* 34(1): 89-106 (1992)
63. Gabriel, S.A., Conejo, A.J., Ruiz, C., Siddiqui, S.: Solving discretely constrained, mixed linear complementarity problems with applications in energy. *Comput. Oper. Res.* 40(5): 1339-1350 (2013)
64. Gabriel, S.A., Siddiqui, S.A., Conejo, A.J., Ruiz, C.: Solving discretely-constrained Nash-Cournot games with an application to power markets. *Netw. Spat. Econ.* 13(3): 307-326 (2013)
65. Gabriel, S.A., Leuthold, F.U. Solving discretely-constrained MPEC problems with applications in electric power markets. *Energy Econ.* 32(1):3-14 (2010)
66. Fomeni, F.D., Gabriel, S.A., Anjos, M.F.: Applications of logic constrained equilibria to traffic networks and to power systems with storage. *Tech. Rep. 10.13140/RG.2.2.23671.424072016* (2016)
67. Hu, J., Mitchell, J.E., Pang, J.S., Bennett, K.P., Kunapuli, G.: On the global solution of linear programs with linear complementarity constraints. *SIAM J. Optimiz.* 19(1): 445-471 (2008)
68. Chen, J., Burer, S.: Globally solving nonconvex quadratic programming problems via completely positive programming. *Math. Prog. Comp.* 4(1): 33-52 (2012)
69. Hu, J., Mitchell, J.E., Pang, J.S., Yu, B.: On linear programs with linear complementarity constraints. *J. Global Optim.* 53(1): 29-51 (2012)
70. Bai, L., Mitchell, J.E., Pang, J.S.: On convex quadratic programs with linear complementarity constraints. *Comput. Optim. Appl.* 54(3): 517-554 (2013)
71. Fang, H., Leyffer, S., Munson, T.: A pivoting algorithm for linear programming with linear complementarity constraints. *Optim. Method Softw.* 27(1): 89-114 (2012)

72. Júdice, J.J.: Algorithms for linear programming with linear complementarity constraints. *Top* 20(1): 4-25 (2012)
73. Watson, L.T., Billups, S.C., Mitchell, J.E., Easterling, D.R.: A globally convergent probability-one homotopy for linear programs with linear complementarity constraints. *SIAM J. Optimiz.* 23(2): 1167-1188 (2013)
74. de Farias, I.R., Kozyreff, E., Zhao, M.: Branch-and-cut for complementarity-constrained optimization. *Math. Prog. Comp.* 2014, 6(4): 365-403.
75. Yu, B., Mitchell, J.E., Pang, J.S.: Solving linear programs with complementarity constraints using branch-and-cut. *arXiv preprint arXiv:1802.02941* (2018)
76. Bai, L., Mitchell, J.E., Pang, J.S.: On conic QPCCs, conic QCQPs and completely positive programs. *Math. Program.* 159(1-2): 109-136 (2016)
77. Hu, X.: *Mathematical Programs with Complementarity Constraints and Game Theory Models in Electricity Markets*. Doctor Dissertation, University of Melbourne (2003)
78. Ehrenmann A. *Equilibrium problems with equilibrium constraints and their application to electricity markets*. Doctor Dissertation, University of Cambridge (2004)
79. Su, C.L.: *Equilibrium Problems with Equilibrium Constraints: Stationarities, Algorithms, and Applications*. Doctor Dissertation, Stanford University (2005)
80. Gabriel, S.A., Conejo, A.J., Fuller, J.D., Ruiz, C.: *Complementarity Modeling in Energy Markets*. Springer, New York, NY USA (2012)
81. Sherali, H.D.: A multiple leader Stackelberg model and analysis. *Oper. Res.* 32(2): 390-404 (1984)
82. DeMiguel, V., Xu, H.: A stochastic multiple-leader Stackelberg model: Analysis, computation, and application. *Oper. Res.* 57(5): 1220-1235 (2009)
83. Hu, X., Ralph, D.: Using EPECs to model bilevel games in restructured electricity markets with locational prices. *Oper. Res.* 55(5): 809-827 (2007)
84. Pang, J.S., Fukushima, M.: Quasi-variational inequalities, generalized Nash equilibria, and multi-leader-follower games. *Comput. Manag. Sci.* 2(1): 21-56 (2005)
85. Ehrenmann, A.: Manifolds of multi-leader Cournot equilibria. *Oper. Res. Lett.* 32(2): 121-125 (2004)
86. Kulkarni, A.A., Shanbhag, U.V.: A shared-constraint approach to multi-leader multi-follower games. *Set-Valued Var. Anal.* 22(4): 691-720 (2014)
87. Kulkarni, A.A., Shanbhag, U.V.: An existence result for hierarchical Stackelberg v/s Stackelberg games. *IEEE Trans. Automat. Contr.* 60(12): 3379-3384 (2015)
88. Guo, L., Lin, G.H., Zhang, D., Zhu, D.: An MPEC reformulation of an EPEC model for electricity markets. *Oper. Res. Lett.* 43(3): 262-267 (2015)
89. Gabriel, S.A., Conejo, A.J., Fuller, J.D., Ruiz, C.: *Complementarity Modeling in Energy Markets*. Springer, New York NY (2012)
90. Leyffer, S., Munson, T.: Solving multi-leader-common-follower games. *Optim. Method. Softw.* 25(4): 601-623 (2010)
91. Hu, M., Fukushima, M.: Variational inequality formulation of a class of multi-leader-follower games. *J. Optim. Theory Appl.* 151(3): 455-473 (2011)
92. Aghassi M, Bertsimas D. Robust game theory. *Math. Program.* 107(1): 231-273 (2006)
93. Hayashi, S., Yamashita, N., Fukushima, M.: Robust Nash equilibria and second-order cone complementarity problems. *J. Nonlinear Convex Anal.* 6(2): 283-296 (2005)
94. Nishimura, R., Hayashi, S., Fukushima, M.: Robust Nash equilibria in N-person non-cooperative games: Uniqueness and reformulation. *Pac. J. Optim.* 5(2): 237-259 (2009)
95. Singh, V.V., Jouini, O., Lissner, A.: Distributionally robust chance-constrained games: Existence and characterization of Nash equilibrium. *Optim. Lett.* 11(7): 1385-1405. (2017)

96. Facchinei, F., Fischer, A., Piccialli, V.: Generalized Nash equilibrium problems and Newton methods. *Math. Program.* 117(1): 163-194 (2009)
97. Facchinei, F., Kanzow, C.: Penalty methods for the solution of generalized Nash equilibrium problems. *SIAM J. Optimiz.* 20(5): 2228-2253 (2010)
98. von Heusinger, A., Kanzow, C.: Optimization reformulations of the generalized Nash equilibrium problem using Nikaido-Isoda-type functions. *Comput. Optim. Appl.* 43(3): 353-377 (2009)
99. Nabetani, K., Tseng, P., Fukushima, M.: Parametrized variational inequality approaches to generalized Nash equilibrium problems with shared constraints. *Comput. Optim. Appl.* 48(3): 423-452 (2011)
100. Kubota, K., Fukushima, M.: Gap function approach to the generalized Nash equilibrium problem. *J. Optim. Theory Appl.* 144(3): 511-531 (2010)
101. von Heusinger, A., Kanzow, C.: Relaxation methods for generalized Nash equilibrium problems with inexact line search. *J. Optim. Theory Appl.* 143(1): 159-183 (2009)
102. Schiro, D.A., Pang, J.S., Shanbhag, U.V.: On the solution of affine generalized Nash equilibrium problems with shared constraints by Lemke's method. *Math. Program.* 142(1): 1-46 (2013)
103. Dreves, A., Sudermann-Merx, N.: Solving linear generalized Nash equilibrium problems numerically. *Optim. Method. Softw.* 31(5): 1036-1063 (2016)
104. Kanzow, C., Steck, D.: Augmented Lagrangian methods for the solution of generalized Nash equilibrium problems. *SIAM J. Optimiz.* 26(4): 2034-2058 (2016)
105. Luna, J.P., Sagastizábal C., Solodov, M.: An approximation scheme for a class of risk-averse stochastic equilibrium problems. *Math. Program.* 157(2): 451-481 (2016)
106. Yu, C.K., van der Schaar, M., Sayed, A.H.: Distributed learning for stochastic generalized Nash equilibrium problems. *IEEE Trans. Signal Proces.* 65(15): 3893-3908 (2017)
107. Dempe, S.: *Foundations of Bilevel Programming*. Kluwer Academic Publishers, New York (2002)
108. Dempe, S., Kalashnikov, V., Pérez-Valdés G.A., Kalashnykova, N.: *Bilevel Programming Problems: Theory, Algorithms and Applications to Energy Networks*. Springer, Berlin Heidelberg (2015)
109. Pozo, D., Sauma, E., Contreras, J.: Basic theoretical foundations and insights on bilevel models and their applications to power systems. *Ann. Oper. Res.* 254(1-2): 303-334 (2017)
110. Chieu, N.H., Lee, G.M.: Constraint qualifications for mathematical programs with equilibrium constraints and their local preservation property. *J. Optim. Theory Appl.* 163(3): 755-776 (2014)
111. Guo, L., Lin, G.H., Jane, J.Y.: Second-order optimality conditions for mathematical programs with equilibrium constraints. *J. Optim. Theory Appl.* 158(1): 33-64 (2013)
112. Jane, J.Y., Zhang, J.: Enhanced Karush-Kuhn-Tucker conditions for mathematical programs with equilibrium constraints. *J. Optim. Theory Appl.* 163(3): 777-794 (2014)
113. Gfrerer, H., Ye, J.J.: New constraint qualifications for mathematical programs with equilibrium constraints via variational analysis. *SIAM J. Optimiz.* 27(2): 842-865 (2017)
114. Patriksson, M., Wynter, L.: Stochastic mathematical programs with equilibrium constraints. *Oper Res. Lett.* 25(4): 159-167 (1999)
115. Lin, G., Fukushima, M.: A class of stochastic mathematical programs with complementarity constraints: Reformulations and algorithms. *J. Ind. Manag. Optim.* 1(1): 99-122 (2005)

- 116. Lin, G., Chen, X., Fukushima, M.: New restricted NCP functions and their applications to stochastic NCP and stochastic MPEC. *Optimization* 56(5-6): 641-653 (2007)
- 117. Lin, G., Chen, X., Fukushima, M.: Solving stochastic mathematical programs with equilibrium constraints via approximation and smoothing implicit programming with penalization. *Math. Program.* 116(1): 343-368 (2009)
- 118. Shapiro, A., Xu, H.: Stochastic mathematical programs with equilibrium constraints, modelling and sample average approximation. *Optimization* 57(3): 395-418 (2008)
- 119. Lin, G.H., Fukushima, M.: Stochastic equilibrium problems and stochastic mathematical programs with equilibrium constraints: A survey. *Pac. J. Optim.* 6(3): 455-482 (2010)
- 120. Hu, M.: Studies on Multi-Leader-Follower Games and Related Issues. Doctor Dissertation, Kyoto University, Kyoto (2012)
- 121. Aussel, D., Bendotti, P., Pištěk, M.: Nash equilibrium in a pay-as-bid electricity market: Part 1-existence and characterization. *Optimization* 66(6): 1013-1025 (2017)
- 122. Aussel, D., Bendotti, P., Pištěk, M.: Nash equilibrium in a pay-as-bid electricity market: Part 2-best response of a producer. *Optimization* 66(6): 1027-1053 (2017)
- 123. Solis, C.U., Clempner, J.B., Poznyak, A.S.: Modeling multi-leader-follower non-cooperative Stackelberg games. *Cybernet. Syst.* 47(8): 650-673 (2016)
- 124. Steffensen, S., Bittner, M.: Relaxation approach for equilibrium problems with equilibrium constraints. *Comput. Oper. Res.* 41: 333-345 (2014)
- 125. Sinha, A., Malo, P., Frantsev, A., Deb, K.: Finding optimal strategies in a multi-period multi-leader-follower Stackelberg game using an evolutionary algorithm. *Comput. Oper. Res.* 41: 374-385 (2014)
- 126. Hu, M., Fukushima, M.: Existence, uniqueness, and computation of robust Nash equilibria in a class of multi-leader-follower games. *SIAM J. Optimiz.* 23(2): 894-916 (2013)
- 127. Lampariello, L., Sagratella, S.: A bridge between bilevel programs and Nash games. *J. Optim. Theory Appl.* 174(2): 613-635 (2017)