Chapter 12
MPEC Methods for Bilevel Optimization Problems

Youngdae Kim, Sven Leyffer, and Todd Munson

Abstract We study optimistic bilevel optimization problems, where we assume the lower-level problem is convex with a nonempty, compact feasible region and satisfies a constraint qualification for all possible upper-level decisions. Replacing the lower-level optimization problem by its first-order conditions results in a mathematical program with equilibrium constraints (MPEC) that needs to be solved. We review the relationship between the MPEC and bilevel optimization problem and then survey the theory, algorithms, and software environments for solving the MPEC formulations.

Keywords Bilevel optimization · Mathematical program with equilibrium constraints · Stationarity · Algorithms

12.1 Introduction

Bilevel optimization problems model situations in which a sequential set of decisions are made: the leader chooses a decision to optimize its objective function, anticipating the response of the follower, who optimizes its objective function given the leader's decision. Mathematically, we have the following optimization problem:

$$\begin{aligned}
&\underset{x,y}{\text{minimize}} && f(x, y) \\
&\text{subject to} && c(x, y) \geq 0 \\
& && y \in \arg\min_{v} \{\, g(x, v) \ \text{ subject to } \ d(x, v) \geq 0 \,\},
\end{aligned} \tag{12.1.1}$$

where f and g are the objectives for the leader and follower, respectively, $c(x, y) \geq 0$ is a joint feasibility constraint, and $d(x, y) \geq 0$ defines the feasible actions the

Y. Kim · S. Leyffer · T. Munson (✉)
Argonne National Laboratory, Lemont, IL, USA
e-mail: youngdae@anl.gov; leyffer@anl.gov; tmunson@anl.gov

© Springer Nature Switzerland AG 2020

S. Dempe, A. Zemkoho (eds.), *Bilevel Optimization*, Springer Optimization and Its Applications 161, https://doi.org/10.1007/978-3-030-52119-6_12

follower can take. Throughout this survey, we assume all functions are at least continuously differentiable in all their arguments.

Our survey focuses on optimistic bilevel optimization. Under this assumption, if the solution set to the lower-level optimization problem is not a singleton, then the leader chooses the element of the solution set that benefits it the most.

12.1.1 Applications and Illustrative Example

Applications of bilevel optimization problems arise in economics and engineering domains; see [10, 24, 27–29, 78, 92, 111, 116] and the references therein. As an example bilevel optimization problem, we consider the moral-hazard problem in economics [72, 101] that models a contract between a principal and an agent who works on a project for the principal. The agent exerts effort on the project by taking an action $a \in \mathcal{A}$ that maximizes its utility, where $\mathcal{A} = \{a | d(a) \geq 0\}$ is a convex set, resulting in output value o_q with given probability $p_q(a) > 0$, where $q \in Q$ and Q is a finite set. The principal observes only the output o_q and compensates the agent with c_q defined in the contract. The principal wants to design an optimal contract consisting of a compensation schedule $\{c_q\}_{q \in Q}$ and a recommended action $a \in \mathcal{A}$ to maximize its expected utility.

Since the agent's action is assumed to be neither observable nor forcible by the principal, feasibility of the contract needs to be defined in order to guarantee the desired behavior of the agent. Two types of constraints are specified: a participation constraint and an incentive-compatibility constraint. The participation constraint says that the agent's expected utility from accepting the contract must be at least as good as the utility U it could receive by choosing a different activity. The incentive-compatibility constraint states that the recommended action should provide enough incentive, such as maximizing its utility, so that the agent chooses it.

Mathematically, the problem is defined as follows:

$$
\begin{aligned}
\underset{c,a}{\text{maximize}} \quad & \sum_{q \in Q} p_q(a) w(o_q - c_q) \\
\text{subject to} \quad & \sum_{q \in Q} p_q(a) u(c_q, a) \geq U \\
& a \in \underset{\tilde{a}}{\arg\max} \left\{ \sum_{q \in Q} p_q(\tilde{a}) u(c_q, \tilde{a}) \quad \text{subject to} \ \ d(\tilde{a}) \geq 0 \right\},
\end{aligned}
$$

where $w(\cdot)$ and $u(\cdot, \cdot)$ are the utility functions of the principal and the agent, respectively. The first constraint is the participation constraint, while the second constraint is the incentive-compatibility constraint. This problem is a bilevel optimization problem with a joint feasibility constraint.

12.1.2 Bilevel Optimization Reformulation as an MPEC

We assume throughout that the lower-level problem is convex with a nonempty, compact feasible region and that it satisfies a constraint qualification for all feasible upper-level decisions, x. Under these conditions, the Karush-Kuhn-Tucker (KKT) conditions for the lower-level optimization problem are both necessary and sufficient, and we can replace the lower-level problem with its KKT conditions to obtain a mathematical program with equilibrium constraints (MPEC) [82, 90, 98]:

$$\begin{aligned} \underset{x,y,\lambda}{\text{minimize}} \quad & f(x, y) \\ \text{subject to} \quad & c(x, y) \geq 0 \\ & \nabla_y g(x, y) - \nabla_y d(x, y)\lambda = 0 \\ & 0 \leq \lambda \ \perp \ d(x, y) \geq 0, \end{aligned} \tag{12.1.2}$$

where $\perp$ indicates complementarity, in other words, that either $\lambda_i = 0$ or $d_i(x, y) = 0$ for all i.

This MPEC reformulation is attractive because it results in a single-level optimization problem, and we show in the subsequent sections that this class of problems can be solved successfully. We note that when the lower-level problem is nonconvex, a solution of the MPEC may be infeasible for the bilevel problem [94, Example 1.1]. Thus, we consider only the case where the lower-level problem is convex when using the MPEC formulation.

In the convex case, the relationship between the original bilevel optimization problem (12.1.1) and the MPEC formulation (12.1.2) is nontrivial, as demonstrated in [30, 32] for smooth functions and [33] for nonsmooth functions. These papers show that a global (local) solution to the bilevel optimization problem (12.1.1) corresponds to a global (local) solution to the MPEC (12.1.2) if the Slater constraint qualification is satisfied by the lower-level optimization problem. Under the Slater constraint qualification, a global solution to the MPEC (12.1.2) corresponds to a global solution to the bilevel optimization problem (12.1.1). A local solution to the MPEC (12.1.2) may not correspond to a local solution to the bilevel optimization problem (12.1.1) unless stronger assumptions guaranteeing the uniqueness of multipliers are made, such as MPEC-LICQ, which is described in Sect. 12.2.2; see [30, 32] for details.

By using the Fritz-John conditions, rather than assuming a constraint qualification and applying the KKT conditions, we can ostensibly weaken the requirements to produce an MPEC reformulation [2]. However, this approach has similar theoretical difficulties with the correspondences between local solutions [31], and we have observed computational difficulties when applying MPEC algorithms to solve the Fritz-John reformulation.

12.1.3 MPEC Problems

Generically, we write MPECs such as (12.1.2) as

$$\begin{aligned} \underset{z}{\text{minimize}} \quad & f(z) \\ \text{subject to} \quad & c(z) \geq 0 \\ & 0 \leq G(z) \ \perp \ H(z) \geq 0, \end{aligned} \tag{12.1.3}$$

where $z = (x, y, \lambda)$ and where we summarize all nonlinear equality and inequality constraints generically as $c(z) \geq 0$.

MPECs are a challenging class of problem because of the presence of the complementarity constraint. The complementarity constraint can be written equivalently as the nonlinear constraint $G(z)^T H(z) \leq 0$, in which case (12.1.3) becomes a standard nonlinear program (NLP). Unfortunately, the resulting problem violates the Mangasarian-Fromowitz constraint qualification (MFCQ) at any feasible point [108]. Alternatively, we can reformulate the complementarity constraint as a disjunction. Unfortunately, the resulting mixed-integer formulation has very weak relaxations, resulting in large search trees [99].

This survey focuses on practical theory and methods for solving MPECs. In the next section, we discuss the stationarity concepts and constraint qualifications for MPECs. In Sect. 12.3, we discuss algorithms for solving MPECs. In Sect. 12.4, we focus on software environments for specifying and solving bilevel optimization problems and mathematical programs with equilibrium constraints, before providing pointers in Sect. 12.5 to related topics we do not cover in this survey.

12.2 MPEC Theory

We now survey stationarity conditions, constraint qualifications, and regularity for the MPEC (12.1.3). The standard concepts for nonlinear programs need to be rethought because of the complementarity constraint $0 \leq G(z) \ \perp \ H(z) \geq 0$, especially when the solution to (12.1.3) has a nonempty biactive set, that is, when both $G_j(z^*) = H_j(z^*) = 0$ for some indices j at the solution z^*.

We assume that the functions in the MPEC are at least continuously differentiable. When the functions are nonsmooth, enhanced M-stationarity conditions and alternative constraint qualifications have been proposed [119].

12.2.1 Stationarity Conditions

In this section, we define first-order optimality conditions for a local minimizer of an MPEC and several stationarity concepts. The stationarity concepts may involve dual variables, and they collapse to a single stationarity condition when a local minimizer has an empty biactive set. Unlike standard nonlinear programming, these concepts may not correspond to the first-order optimality of the MPEC when there is a nonempty biactive set.

One derivation of stationarity conditions for MPECs is to replace the complementarity condition with a set of nonlinear inequalities, such as $G_j(z)H_j(z) \leq 0$, and then produce the stationarity conditions for the equivalent nonlinear program:

$$\begin{aligned} &\underset{z}{\text{minimize}} && f(z) && \\ &\text{subject to} && c_i(z) \geq 0 && \forall i = 1, \ldots, m \\ & && G_j(z) \geq 0,\, H_j(z) \geq 0,\, G_j(z)H_j(z) \leq 0 && \forall j = 1, \ldots, p, \end{aligned} \tag{12.2.1}$$

where $z \in \mathbb{R}^n$. An alternative formulation as an NLP is obtained by replacing $G_j(z)H_j(z) \leq 0$ by $G(z)^T H(z) \leq 0$. Unfortunately, (12.2.1) violates the MFCQ at *any* feasible point [108] because the constraint $G_j(z)H_j(z) \leq 0$ does not have an interior. Hence, the KKT conditions may not be directly applicable to (12.2.1).

Instead, stationarity concepts are derived from several different approaches: local analysis of the NLPs associated with an MPEC [90, 108]; Clarke's nonsmooth analysis to the complementarity constraints by replacing them with the min function [23, 108]; and Mordukhovich's generalized differential calculus applied to the generalized normal cone [97]. The first method results in B-, weak-, and strong-stationarity concepts. The second and the third lead to C- and M-stationarity, respectively. These stationarity concepts coincide with each other when the solution has an empty biactive set.

We begin by defining the biactive index set $\mathcal{D}$ (or denoted by $\mathcal{D}(z)$ to emphasize its dependency on z) and its partition $\mathcal{D}_{01}$ and $\mathcal{D}_{10}$ such that

$$\mathcal{D} := \{j \mid G_j(z) = H_j(z) = 0\},\ \ \mathcal{D} = \mathcal{D}_{01} \cup \mathcal{D}_{10}, \mathcal{D}_{01} \cap \mathcal{D}_{10} = \emptyset. \tag{12.2.2}$$

If we define an NLP$_{\mathcal{D}_{01},\mathcal{D}_{10}}$

$$\begin{aligned} &\underset{z}{\text{minimize}} && f(z) && \\ &\text{subject to} && c_i(z) \geq 0 && \forall i = 1, \ldots, m \\ & && G_j(z) = 0 && \forall j : G_j(z) = 0,\, H_j(z) > 0 \\ & && H_j(z) = 0 && \forall j : G_j(z) > 0,\, H_j(z) = 0 \\ & && G_j(z) = 0,\, H_j(z) \geq 0 && \forall j \in \mathcal{D}_{01} \\ & && G_j(z) \geq 0,\, H_j(z) = 0 && \forall j \in \mathcal{D}_{10}, \end{aligned} \tag{12.2.3}$$

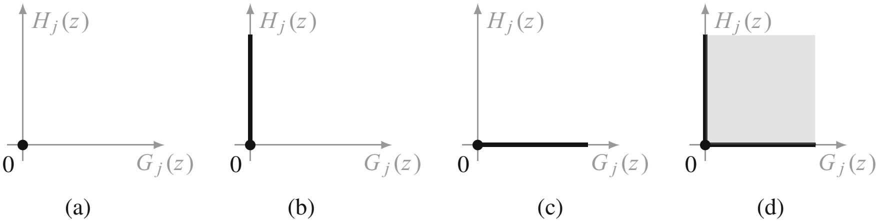

Fig. 12.1 Feasible regions of the NLPs associated with a complementarity constraint $0 \leq G_j(z) \perp H_j(z) \geq 0$ for $j \in \mathcal{D}$. (**a**) Tightened NLP. (**b**) $\text{NLP}_{\mathcal{D}_{01}}$. (**c**) $\text{NLP}_{\mathcal{D}_{10}}$. (**d**) Relaxed NLP

then z^* is a local solution to the MPEC if and only if z^* is a local solution for all the associated NLPs indexed by $(\mathcal{D}_{01}, \mathcal{D}_{10})$. The number of NLPs that need to be checked is exponential in the number of biactive indices, which can be computationally intractable.

Like many other mathematical programs, the geometric first-order optimality conditions are defined in terms of the tangent cone. A feasible point z^* is called a geometric Bouligand- or B-stationary point if $\nabla f(z^*)^T d \geq 0, \forall d \in \mathcal{T}_{\text{MPEC}}(z^*)$, where the tangent cone $\mathcal{T}(z^*)$ to a feasible region $\mathcal{F}$ at $z^* \in \mathcal{F}$ is defined as

$$\mathcal{T}(z^*) := \left\{ d \mid d = \lim_{t_k \downarrow 0} \frac{z^k - z^*}{t_k} \text{ for some } \{z^k\}, z^k \to z^*, z^k \in \mathcal{F}, \forall k \right\}. \tag{12.2.4}$$

To facilitate the analysis of the tangent cone $\mathcal{T}_{\text{MPEC}}(z)$ at $z \in \mathcal{F}_{\text{MPEC}}$, we subdivide it into a set of tangent cones of the associated NLPs:

$$\mathcal{T}_{\text{MPEC}}(z) := \bigcup_{(\mathcal{D}_{01}, \mathcal{D}_{10})} \mathcal{T}_{\text{NLP}_{\mathcal{D}_{01}, \mathcal{D}_{10}}}(z). \tag{12.2.5}$$

Therefore, z^* is a geometric B-stationary point if and only if it is a geometric B-stationary point for all of its associated NLPs indexed by $(\mathcal{D}_{01}, \mathcal{D}_{10})$.

Additionally, two more NLPs, tightened [108] and relaxed [48, 108] NLPs, are defined by replacing the last two conditions of (12.2.3) with a single condition: tightened NLP (TNLP) is defined by setting $G_j(z) = H_j(z) = 0$ for all $j \in \mathcal{D}$, whereas relaxed NLP (RNLP) relaxes the feasible region by defining $G_j(z) \geq 0$, $H_j(z) \geq 0$ for all $j \in \mathcal{D}$. These NLPs provide a foundation to define constraint qualifications for MPECs and strong stationarity.

Figure 12.1 depicts the feasible regions of the NLPs associated with a complementarity constraint $0 \leq G_j(z) \perp H_j(z) \geq 0$ for $j \in \mathcal{D}$. One can easily verify that

$$\mathcal{T}_{\text{TNLP}}(z) \subseteq \mathcal{T}_{\text{MPEC}}(z) \subseteq \mathcal{T}_{\text{RNLP}}(z). \tag{12.2.6}$$

When $\mathcal{D} = \emptyset$, the equality holds throughout (12.2.6), and *lower-level strict complementarity* [104] holds. From (12.2.6), if z^* is a local minimizer of RNLP(z^*), then z^* is a local minimizer of the MPEC, but not vice versa [108].

Algebraically, B-stationarity is defined by using a linear program with equilibrium constraints (LPEC), an MPEC with all the functions being linear, over a linearized cone. For a feasible z^*, if $d = 0$ is a solution to the following LPEC, then z^* is called a B-stationary (or an algebraic B-stationary) point:

$$\begin{aligned} \underset{z}{\text{minimize}} \quad & \nabla f(z^*)^T d \\ \text{subject to} \quad & d \in \mathcal{T}^{\text{lin}}_{\text{MPEC}}(z^*), \end{aligned} \tag{12.2.7}$$

where

$$\begin{aligned} \mathcal{T}^{\text{lin}}_{\text{MPEC}}(z^*) := \big\{d \mid \;& \nabla c_i(z^*)^T d \ge 0,\ \forall i : c_i(z^*) = 0, \\ & \nabla G_j(z^*)^T d = 0,\ \forall j : G_j(z^*) = 0, H_j(z^*) > 0, \\ & \nabla H_j(z^*)^T d = 0,\ \forall j : G_j(z^*) > 0, H_j(z^*) = 0, \\ & 0 \le \nabla G_j(z^*)^T d \perp \nabla H_j(z^*)^T d \ge 0,\ \forall j \in \mathcal{D}(z^*)\big\}. \end{aligned} \tag{12.2.8}$$

As with geometric B-stationarity, B-stationarity is difficult to check because it involves the solution of an LPEC that may require the solution of an exponential number of linear programs, unless *all* these linear programs share a common multiplier vector. Such a common multiplier vector exists if MPEC-LICQ holds, which we define in Sect. 12.2.2.

Since $\mathcal{T}_{\text{MPEC}}(z^*) \subseteq \mathcal{T}^{\text{lin}}_{\text{MPEC}}(z^*)$ [42], B-stationarity implies geometric B-stationarity, but not vice versa. A similar equivalence (12.2.5) and inclusion relationship (12.2.6) hold between the linearized cones of its associated NLPs [42].

The next important stationarity concept is strong stationarity.

Definition 12.2.1 A point z^* is called *strongly stationary* if there exist multipliers satisfying the stationarity of the RNLP. △

Note that if $\hat{\nu}_{1j}$ and $\hat{\nu}_{2j}$ are multipliers for $G_j(z^*)$ and $H_j(z^*)$, respectively, then $\hat{\nu}_{1j}, \hat{\nu}_{2j} \ge 0$ for $j \in \mathcal{D}(z^*)$. Strong stationarity implies B-stationarity due to (12.2.6) and the inclusion relationship between the tangent and linearized cones. Equivalence of stationarity between (12.2.1) and the RNLP was shown in [5, 48, 103].

Other stationarity conditions differ from strong stationarity in that the conditions on the sign of the multipliers, $\hat{\nu}_{1j}$ and $\hat{\nu}_{2j}$, are relaxed when $G_j(z^*) = H_j(z^*) = 0$. One can easily associate these stationarity concepts with the stationarity conditions for one of its associated NLPs. If we define the biactive set $\mathcal{D}^* := \mathcal{D}(z^*)$, we can

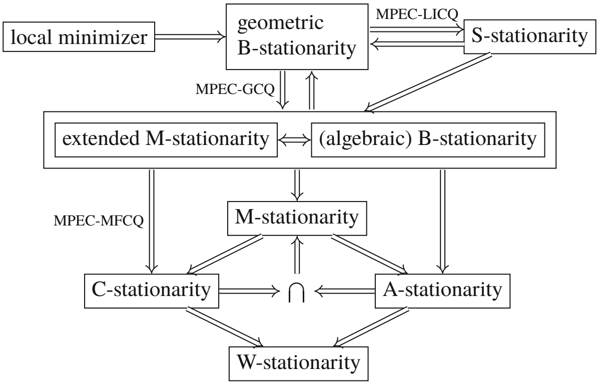

Fig. 12.2 Relationships between stationarity concepts where M-stationarity is the intersection of C- and A-stationarity

state these "stationarity" concepts by replacing the sign of the multipliers in the set $\mathcal{D}^*$ as follows:

- z^* is called weak stationary if there are no sign restrictions on $\hat{\nu}_{1j}$ and $\hat{\nu}_{2j}, \forall j \in \mathcal{D}^*$.
- z^* is called A-stationary if $\hat{\nu}_{1j} \geq 0$ or $\hat{\nu}_{2j} \geq 0, \forall j \in \mathcal{D}^*$.
- z^* is called C-stationary if $\hat{\nu}_{1j}\hat{\nu}_{2j} \geq 0, \forall j \in \mathcal{D}^*$.
- z^* is called M-stationary if either $\hat{\nu}_{1j} > 0$ and $\hat{\nu}_{2j} > 0$ or $\hat{\nu}_{1j}\hat{\nu}_{2j} = 0, \forall j \in \mathcal{D}^*$.

Of particular note is M-stationarity, which implies that if z^* is M-stationary, then z^* satisfies the first-order optimality conditions for at least one of the nonlinear programs (12.2.3). This condition seems to be the best one can achieve without exploring the combinatorial number of possible partitions for the biactive constraints. The extended M-stationarity in [56] extends the notion of M-stationarity in a way that it holds at z^* when M-stationarity is satisfied for each critical direction d defined by $d \in \mathcal{T}^{\text{lin}}_{\text{MPEC}}(z^*)$ and $\nabla f(z^*)^T d \leq 0$, possibly with different multipliers. Thus, if z^* is extended M-stationary, then z^* satisfies the first-order optimality conditions for all the associated NLPs. Hence it is also B-stationary. When a constraint qualification is satisfied, B-stationarity implies extended M-stationarity. Figure 12.2 summarizes relationships between stationarity concepts.

As in the stationarity concepts, the second-order sufficient conditions (SOSCs) of an MPEC are defined in terms of the associated NLPs. In particular, SOSC is defined at a strongly stationary point so its multipliers work for all the associated NLPs. Depending on the underlying critical cone, we have two different SOSCs:

RNLP-SOSC and MPEC-SOSC. In Definition 12.2.2, the Lagrangian $\mathcal{L}$ and the critical cone for the RNLP are defined as follows:

$$
\begin{aligned}
\mathcal{L}(z,\lambda,\hat{\nu}_1,\hat{\nu}_2) := f(z) \quad & - \sum_{i\in\mathcal{E}\cup\mathcal{I}} c_i(z)\lambda_i - \sum_{j=1}^{p} G_j(z)\hat{\nu}_{1j} - \sum_{j=1}^{p} H_j(z)\hat{\nu}_{2j}, \\
C_{\mathrm{RNLP}}(z^*) := \{d \mid \quad & \nabla c_i(z^*)^T d = 0,\ \ \forall i : \lambda_i^* > 0, \\
& \nabla c_i(z^*)^T d \geq 0,\ \ \forall i : c_i(z^*) = 0, \lambda_i^* = 0, \\
& \nabla G_j(z^*)^T d = 0,\ \ \forall j : G_j(z^*) = 0, H_j(z^*) > 0, \\
& \nabla H_j(z^*)^T d = 0,\ \ \forall j : G_j(z^*) > 0, H_j(z^*) = 0, \\
& \nabla G_j(z^*)^T d = 0,\ \ \forall j \in D(z^*), \hat{\nu}_{1j}^* > 0, \\
& \nabla G_j(z^*)^T d \geq 0,\ \ \forall j \in D(z^*), \hat{\nu}_{1j}^* = 0, \\
& \nabla H_j(z^*)^T d = 0,\ \ \forall j \in D(z^*), \hat{\nu}_{2j}^* > 0, \\
& \nabla H_j(z^*)^T d \geq 0,\ \ \forall j \in D(z^*), \hat{\nu}_{2j}^* = 0\big\}.
\end{aligned}
\tag{12.2.9}
$$

Definition 12.2.2 RNLP-SOSC is satisfied at a strongly stationary point z^* with its multipliers $(\lambda^*, \hat{\nu}_1^*, \hat{\nu}_2^*)$ if there exists a constant $\omega > 0$ such that

$$
d^T \nabla_{zz}^2 \mathcal{L}(z^*, \lambda^*, \hat{\nu}_1^*, \hat{\nu}_2^*) d \geq \omega \text{ for all } d \neq 0, d \in C_{\mathrm{RNLP}}(z^*).
$$

If the conditions hold for $C_{\mathrm{MPEC}}(z^*)$ instead of $C_{\mathrm{RNLP}}(z^*)$, where $C_{\mathrm{MPEC}}(z^*) := C_{\mathrm{RNLP}}(z^*) \cap \{d \mid \min(\nabla G_j(z^*)^T d, \nabla H_j(z^*)^T d) = 0, \forall j \in \mathcal{D}(z^*), \hat{\nu}_{1j}^* = \hat{\nu}_{2j}^* = 0\}$, then we say that MPEC-SOSC is satisfied. △

We note that $d \in C_{\mathrm{MPEC}}(z^*)$ if and only if it is a critical direction of any of the $\mathrm{NLP}_{\mathcal{D}_{01},\mathcal{D}_{10}}(z^*)$'s at z^* [104, 108]. Thus, MPEC-SOSC holds at z^* if and only if SOSC is satisfied at z^* for all of its associated NLPs, which leads to the conclusion that z^* is a strict local minimizer of the MPEC.

In a similar fashion, we define strong SOSC (SSOSC) for RNLPs. Using RNLP-SSOSC, we can obtain stability results of MPECs by applying the stability theory of nonlinear programs [77, 105] to the RNLP. The stability property can be used to show the uniqueness of a solution of regularized NLPs to solve the MPEC as in Sect. 12.3. A critical cone $C_{\mathrm{RNLP}}^{\mathrm{S}}(z^*)$ is used instead of $C_{\mathrm{RNLP}}(z^*)$, which expands it by removing the feasible directions for inequalities associated with zero multipliers:

$$
\begin{aligned}
C_{\mathrm{RNLP}}^{\mathrm{S}}(z^*) := \big\{d \mid \quad & \nabla c_i(z^*)^T d = 0,\ \ \forall i : \lambda_i^* > 0, \\
& \nabla G_j(z^*)^T d = 0,\ \ \forall j : \hat{\nu}_{1j}^* \neq 0, \\
& \nabla H_j(z^*)^T d = 0,\ \ \forall j : \hat{\nu}_{2j}^* \neq 0\big\}.
\end{aligned}
\tag{12.2.10}
$$

Definition 12.2.3 RNLP-SSOSC is satisfied at a strongly stationary point z^* with its multipliers $(\lambda^*, \hat{\nu}_1^*, \hat{\nu}_2^*)$ if there exists a constant $\omega > 0$ such that

$$d^T \nabla_{zz}^2 \mathcal{L}(z^*, \lambda^*, \hat{\nu}_1^*, \hat{\nu}_2^*) d \geq \omega \text{ for all } d \neq 0, d \in C_{\text{RNLP}}^{\text{S}}(z^*).$$

△

By definition, we have an inclusion relationship between SOSCs: RNLP-SSOSC ⇒ RNLP-SOSC ⇒ MPEC-SOSC. Reverse directions do not hold in general; an example was presented in [108] showing that MPEC-SOSC $\nRightarrow$ RNLP-SOSC.

12.2.2 Constraint Qualifications and Regularity Assumptions

Constraint qualifications (CQs) for MPECs guarantee the existence of multipliers for the stationarity conditions to hold. They are an extension of the corresponding CQs for the tightened NLP. Among many CQs in the literature, we have selected five. Three of them are frequently assumed in proving stationarity properties of a limit point of algorithms for solving MPECs described in Sect. 12.3. The remaining two are conceptual and much weaker than the first ones, but they can be used to prove that every local minimizer is at least M-stationary.

We start with the first three CQs. Because of the space limit, we do not specify the definition of the CQs in the context of NLPs. In Definition 12.2.4, LICQ denotes the linear independence constraint qualification, and CPLD represents constant positive linear dependence [102].

Definition 12.2.4 An MPEC (12.1.2) is said to satisfy *MPEC-LICQ* (*MPEC-MFCQ*, *MPEC-CPLD*) at a feasible point z if the corresponding tightened NLP satisfies LICQ (MFCQ, CPLD) at z. △

We note that MPEC-LICQ holds at a feasible point z if and only if all of its associated NLPs satisfy LICQ at z. This statement is true because active constraints at any feasible point do not change between the associated NLPs. For example, MPEC-LICQ holds if and only if LICQ holds for the relaxed NLP.

Under MPEC-LICQ, B-stationarity is equivalent to strong stationarity [108] since the multipliers are unique, thus having the same nonnegative signs for a biactive set among the associated NLPs.

The above CQs provide sufficient conditions for the following much weaker CQs to hold. These CQs are in general difficult to verify but provide insight into what stationarity we can expect for a local minimizer. In the following definition, ACQ and GCQ denote Abadie and Guignard constraint qualification, respectively, and for a cone C its polar is defined by $C^\circ := \{y \mid \langle y, x\rangle \leq 0, \forall x \in C\}$.

Definition 12.2.5 The MPEC (12.1.3) is said to satisfy *MPEC-ACQ* (*MPEC-GCQ*) at a feasible point z if $\mathcal{T}_{\text{MPEC}}(z) = \mathcal{T}_{\text{MPEC}}^{\text{lin}}(z)$ $(\mathcal{T}_{\text{MPEC}}(z)^\circ = \mathcal{T}_{\text{MPEC}}^{\text{lin}}(z)^\circ)$. △

Although MPEC-ACQ or MPEC-GCQ holds, we cannot directly apply the Farkas lemma to show the existence of multipliers because the linearized cone may not be a polyhedral convex set. However, M-stationarity is shown to hold for each local minimizer under MPEC-GCQ by using the limiting normal cones and separating the complementarity constraints from other constraints [43, 56].

Local preservation of constraint qualifications has been studied in [20], which shows that for many MPEC constraint qualifications, if z^* satisfies an MPEC constraint qualification, then all feasible points in a neighborhood of z^* also satisfy that MPEC constraint qualification.

As with standard nonlinear programming, a similar implication holds between CQs for MPECs: MPEC-LICQ ⇒ MPEC-MFCQ ⇒ MPEC-CPLD ⇒ MPEC-ACQ ⇒ MPEC-GCQ.

12.3 Solution Approaches

In this section, we classify and outline solution methods for MPECs and summarize their convergence results. Solution methods for MPECs such as (12.1.3) can be categorized into three broad classes:

1. **Nonlinear programming methods** that rewrite the complementarity constraint in (12.1.3) as a nonlinear set of inequalities, such as

$$G(z) \geq 0, \;\; H(z) \geq 0, \;\; \text{and} \;\; G(z)^T H(z) \leq 0, \tag{12.3.1}$$

   and then apply NLP techniques; see, for example, [26, 45, 74, 87, 103, 104, 109]. Unfortunately, convergence properties are generally weak for this class of methods, typically resulting in C-stationary limits unless strong assumptions are made on the limit point.
2. **Combinatorial methods** that tackle the combinatorial nature of the disjunctive complementarity constraint directly. Popular approaches include pivoting methods [40], branch-and-cut methods [8, 9, 11], and active-set methods [57, 84, 88]. This class of methods has the strongest convergence properties.
3. **Implicit methods** that assume that the complementarity constraint has a unique solution for every upper-level choice of variables. For example, if we assume that the lower-level problem has a unique solution, then we can express the lower-level variables $y = y(x)$ in (12.1.2), and use the KKT conditions to eliminate $(y(x), \lambda(x))$, resulting in a reduced nonsmooth problem

$$\underset{x}{\text{minimize}} \;\; f(x, y(x)) \quad \text{subject to} \;\; c(x, y(x)) \geq 0$$

   that can be solved by using methods for nonsmooth optimization. See the monograph [98] and the references [63, 118] for more details.

In this survey, we do not discuss implicit methods further and instead concentrate on the first two classes of methods.

12.3.1 NLP Methods for MPECs

NLP methods are attractive because they allow us to leverage powerful numerical solvers. Unfortunately, the system (12.3.1) violates a standard stability assumption for NLP at any feasible point. In [108], the authors show that (12.3.1) violates MFCQ at any feasible point. Other nonlinear reformulations of the complementarity constraint (12.3.1) are possible. In [46, 83], the authors experiment with a range of reformulations using different nonlinear complementarity functions, but they observe that the formulation (12.3.1) is the most efficient format in the context of sequential quadratic programming (SQP) methods. We also note that reformulations that use nonlinear *equality constraints* such as $G(z)^T H(z) = 0$ are not as efficient, because the redundant lower bound can slow convergence.

Because traditional analyses of NLP solvers rely heavily on a constraint qualification, it is remarkable that convergence results can still be proved. Here, we briefly review how SQP methods can be shown to converge quadratically for MPECs, provided that we reformulate the nonlinear complementarity constraint (12.3.1) using slacks as

$$s_1 = G(z),\ \ s_2 = H(z) \geq 0,\ \ s_1 \geq 0,\ \ s_2 \geq 0,\ \ \text{and}\ \ s_1^T s_2 \leq 0. \tag{12.3.2}$$

One can show that close to a strongly stationary point that satisfies MPEC-LICQ and RNLP-SOSC, an SQP method applied to this reformulation converges quadratically to a strongly stationary point, provided that all QP approximations remain feasible. The authors [48] show that these assumptions are difficult to relax, and they give a counterexample that shows that the slacks are necessary for convergence. One undesirable assumption is that all QP approximations must remain feasible, but one can show that this assumption holds if the lower-level problem satisfies a certain mixed-P property; see, for example, [90]. In practice [45], a simple heuristic is implemented that relaxes the linearization of the complementarity constraint.

In general, the failure of MFCQ motivates the use of regularizations within NLP methods, such as penalization or relaxation of the complementarity constraint, and these two classes of methods are discussed next.

12.3.1.1 Relaxation-Based NLP Methods

An attractive way to solve MPECs is to relax the complementarity constraint in (12.3.1) by using a positive relaxation parameter, $t > 0$:

$$\begin{aligned} \underset{z,u,v}{\text{minimize}} \quad & f(z) \\ \text{subject to} \quad & c_i(z) \geq 0 \\ & G_j(z) \geq 0, H_j(z) \geq 0, G_j(z)H_j(z) \leq t \quad \forall j = 1, \ldots, p. \end{aligned} \tag{12.3.3}$$

This NLP then generally satisfies MFCQ for any $t > 0$, and the main idea is to (approximately) solve these NLPs for a sequence of regularization parameters, $t_k \searrow 0$. This approach has been studied from a theoretical perspective in [104, 109]. Under the unpractical assumption that each regularized NLP is solved exactly, the authors show convergence to C-stationary points.

More recently, interior-point methods based on the relaxation (12.3.3) have been proposed [87, 103] in which the parameter t is chosen to be proportional to the barrier parameter and is updated at the end of each (approximate) barrier subproblem solve. Numerical difficulties may arise when the relaxation parameter becomes small, because the limiting feasible set of the regularized NLP (12.3.3) has no strict interior.

An alternative regularization scheme is proposed in [74], based on the reformulation of (12.3.1) as

$$G_j(z) \geq 0, \;\; H_j(z) \geq 0, \;\; \text{and} \;\; \phi(G_j(z), H_j(z), t) \leq 0, \tag{12.3.4}$$

where

$$\phi(a, b, t) = \begin{cases} ab & \text{if } a + b \geq t \\ -\frac{1}{2}(a^2 + b^2) & \text{if } a + b < t. \end{cases}$$

In [74], convergence properties are proved when the nonlinear program is solved inexactly, as one would find in practice. The authors show that typically, but not always, methods converge to M-stationary points with exact NLP solves but converge to only C-stationary points with inexact NLP solves. Other regularization methods have been proposed, and a good theoretical and numerical comparison can be found in [64], with later methods in [73]. Under suitable conditions, these methods can be shown to find M- or C-stationary points for the MPEC when the nonlinear program is solved exactly.

An interesting two-sided relaxation scheme is proposed in [26]. The scheme is motivated by the observation that the complementarity constraint in the slack formulation (12.3.2) can be interpreted as a pair of bounds, $s_{ij} \geq 0$, or, $s_{ij} \leq 0$, and strong stationarity requires a multiplier for at most one of these bounds in each pair. The authors propose a strictly feasible two-sided relaxation of the form

$$s_{1j} \geq -\delta_{1j}, \;\; s_{2j} \geq -\delta_{2j}, \;\; s_{1j} s_{2j} \leq \delta_{cj}.$$

By using the multiplier information from inexact subproblem solves, the authors decide which parameter δ_1, δ_2, or δ_c needs to be driven to zero for each complementarity pair. The authors propose an interior-point algorithm and show convergence to C-stationary points, local superlinear convergence, and identification of the optimal active set under MPEC-LICQ and RNLP-SOSC conditions.

12.3.1.2 Penalization-Based NLP Methods

An alternative regularization approach is based on penalty methods. Both exact penalty functions and augmented Lagrangian methods have been proposed for solving MPECs. The penalty approach for MPECs dates back to [41] and penalizes the complementarity constraint in the objective function, after introducing slacks:

$$\begin{aligned} \underset{z,u,v}{\text{minimize}} \quad & f(z) + \pi s_1^T s_2 \\ \text{subject to} \quad & c_i(z) \geq 0 && \forall i = 1, \ldots, m \\ & G_j(z) - s_{1j} = 0, H_j(z) - s_{2j} = 0 && \forall j = 1, \ldots, p \\ & s_2 \geq 0, s_1 \geq 0 \end{aligned} \tag{12.3.5}$$

for positive penalty parameter π. If π is chosen large enough, the solution of the MPEC can be recast as the minimization of a single penalty function problem. The appropriate value of π is unknown in advance, however, and must be estimated during the course of the minimization. In general, if limit points are only B-stationary and not strongly stationary, then the penalty parameter, π, must diverge to infinity, and this penalization is not exact.

This approach was first studied by [3] in the context of active-set SQP methods, although it had been used before to solve engineering problems [41]. It has been adopted as a heuristic to solve MPECs with interior-point methods in LOQO by [15], who present very good numerical results. A more general class of exact penalty functions was analyzed by [66], who derive global convergence results for a sequence of penalty problems that are solved exactly, while the author in [4] derives similar global results in the context of inexact subproblem solves.

In [85], the authors study a general interior-point method for solving (12.3.5). Each barrier subproblem is solved approximately to a tolerance ϵ_k that is related to the barrier parameter, μ_k. Under strict complementarity, MPEC-LICQ, and RNLP-SOSC, the authors show convergence to C-stationary points that are strongly stationary if the product of MPEC-multipliers and primal variables remains bounded. Superlinear convergence can be shown, provided that the tolerance and barrier parameter satisfy the conditions

$$\frac{(\epsilon + \mu)^2}{\epsilon} \to 0, \ \frac{(\epsilon + \mu)^2}{\mu} \to 0, \ \text{and} \ \frac{\mu}{\epsilon} \to 0.$$

A related approach to penalty functions is the elastic mode that is implemented in SNOPT [58, 59]. The elastic approach [6] combines both a relaxation of the constraints and penalization of the complementarity conditions to solve a sequence of optimization problems

$$\begin{aligned} \underset{z,u,v,t}{\text{minimize}} \quad & f(z) + \pi(t + s_1^T s_2) \\ \text{subject to} \quad & c_i(z) \geq -t && \forall i = 1, \ldots .m \\ & -t \leq G_j(z) - s_{1j} \leq t, -t \leq H_j(z) - s_{2j} \leq t && \forall j = 1, \ldots, p \\ & s_1 \geq 0, s_2 \geq 0, 0 \leq t \leq \bar{t}, \end{aligned}$$

where t is a variable with upper bound $\bar{t}$ that relaxes some of the constraints and π is the penalty term on both the constraint relaxation and the complementarity conditions. This problem can be interpreted as a mixture of ℓ_∞ and ℓ_1 penalties. The problem is solved for a sequence of π that may need to converge to infinity. Under suitable conditions such as MPEC-LICQ, the method is shown to converge to M-stationary points, even with inexact NLP solves.

An alternative to the ℓ_1 penalty-function approaches presented above are augmented Lagrangian approaches, which have been adapted to MPECs [51, 69]. These approaches are related to stabilized SQP methods and work by imposing an artificial upper bound on the multiplier. Under MPEC-LICQ one can show that if the sequence of multipliers has a bounded subsequence, then the limit point is strongly stationary. Otherwise, it is only C-stationary.

Remark 12.3.1 NLP methods for MPECs currently provide the most practical approach to solving MPECs, and form the basis for the most efficient and robust software packages; see Sect. 12.4. In general, however, NLP methods may converge only slowly to a solution or fail to converge if the limit point is not strongly stationary. △

The result in [74] seems to show that the best we can hope for from NLP solvers is convergence to C-stationary points. Unfortunately, these points may include limit points at which first-order strict descent directions exist, and which do not correspond to stationary points in the classical sense. To the best of our knowledge, the only way around this issue is to tackle the combinatorial nature of the complementarity constraint directly, and in the next section we describe methods that do so.

12.3.2 Combinatorial Methods for MPECs

Despite the success of NLP solvers in tackling a wide range of MPECs, for some classes of problems these solvers still fail. In particular, problems whose stationary points are B-stationary but not strongly stationary can cause NLP solvers to either fail or exhibit slow convergence. Unfortunately, this behavior also occurs when other pathological situations occur (such as convergence to C- or M-stationary points that are not strongly stationary), and it is not easily diagnosed or remedied.

This observation motivates the development of more robust methods for MPECs that guarantee convergence to B-stationary points. Methods with this property must resolve the combinatorial complexity of the complementarity constraint, and we discuss these methods here.

Specialized methods for solving linear programs with linear equilibrium constraints (LPECs), in which all the functions, $f(z)$, $c(z)$, $G(z)$, and $H(z)$, are linear, include methods based on disjunction for computing global solutions [11, 60, 67, 68], pivot-based methods for computing B-stationary points [40], and penalty methods based on a difference of convex functions formulation [70]. Global optimization methods for problems with a convex quadratic objective function and linear complementarity constraints are found in [8]. Algorithms for problems with a nonlinear objective function and linear complementarity constraints include combinatorial [9], active-set [52], sequential quadratic programming [88], and sequential linear programming [57] methods.

One early general method for obtaining B-stationary points is the branch-and-bound method proposed in [11]. It starts by solving a relaxation of (12.1.3) obtained by relaxing the complementarity between $G(z)$ and $H(z)$. If the solution satisfies $G(z)^T H(z) = 0$, then it is a B-stationary point. Otherwise, there exists an index j such that $G_j(z)H_j(z) > 0$, and we branch on this disjunction by creating two new child problems that set $G_j(z) = 0$ and $H_j(z) = 0$, respectively. This process is then repeated, creating a branch-and-bound tree. A branch-and-cut approach is described in [67] for solving LPECs. The algorithm is based on equivalent mixed-integer reformulations and the application of Benders decomposition [14] with linear programming (LP) relaxations. In contrast, the authors in [40] generalize LP pivoting techniques to locally solve LPECs. The authors prove convergence to a B-stationary point, but unlike [67] do not obtain globally optimal solutions.

The SQPEC approach extends SQP methods by taking special care of the complementarity constraint [110]. This method minimizes a quadratic approximation of the Lagrangian subject to a linearized feasible set and a linearized form of the complementarity constraint. Unfortunately, this method can converge to M-stationary points, rather than the desired B-stationary points, as the following counterexample shows [84]:

$$\underset{x,y}{\text{minimize}}\ (x-1)^2 + y^3 + y^2 \quad \text{subject to}\ \ 0 \le x \ \perp\ y \ge 0. \tag{12.3.6}$$

Starting from $(x_0, y_0) = (0, t)$ for $0 < t < 1$, SQPEC generates iterates

$$(x^{(k+1)}, y^{(k+1)}) = \left(0, \frac{3y^{(k)^2}}{6y^{(k)} + 2}\right)$$

that converge quadratically to the M-stationary point $(0, 0)$, at which we can easily find a descent direction $(1, 0)$ and which is hence not B-stationary.

An alternative class of methods to SQPEC methods that provide convergence to B-stationary points is extensions of SLQP methods to MPECs; see, for example, [16, 17, 21, 47]. The method is motivated by considering the linearized tangent cone, $\mathcal{T}^{\text{lin}}_{\text{MPEC}}(z^*)$ in (12.2.8) as a direction-finding problem. This method solves a sequence of LPECs inside a trust region [84] with radius Δ around the current point z:

$$\text{LPEC}(z,\Delta)\begin{cases}\underset{d}{\text{minimize}} & \nabla f(z)^T d\\ \text{subject to} & c(z)+\nabla c(z)^T d \geq 0,\\ & 0 \leq G(z)+\nabla G(z)^T d \;\perp\; H(z)+\nabla H(z)^T d \geq 0,\\ & \|d\| \leq \Delta.\end{cases}$$

The LPEC need be solved only locally, and the pivoting method of [40] is a practical and efficient method of solving the LPEC. Given a solution $d \neq 0$, we find the active sets that are predicted by the LPEC,

$$\mathcal{A}_c(z+d) := \left\{i : c_i(z)+\nabla c_i(z)^T d = 0\right\} \tag{12.3.7}$$

$$\mathcal{A}_G(z+d) := \left\{j : G_j(z)+\nabla G_j(z)^T d = 0\right\} \tag{12.3.8}$$

$$\mathcal{A}_H(z+d) := \left\{j : H_j(z)+\nabla H_j(z)^T d = 0\right\}, \tag{12.3.9}$$

and solve the corresponding equality-constrained quadratic program (EQP):

$$\text{EQP}(z+d)\begin{cases}\underset{d}{\text{minimize}} & \nabla f(z)^T d + \frac{1}{2}d^T\nabla^2\mathcal{L}(z)d\\ \text{subject to} & c_i(z)+a_i(z)^T d = 0,\ \forall i \in \mathcal{A}_c(z+d)\\ & G_j(z)+\nabla G_j(z)^T d = 0,\ \forall j \in \mathcal{A}_G(z+d)\\ & H_j(z)+\nabla H_j(z)^T d = 0,\ \forall j \in \mathcal{A}_H(z+d).\end{cases}$$

We note that EQP$(z + d)$ can be solved as a linear system of equations. The goal of the EQP step is to provide fast convergence near a local minimum. Global convergence is promoted through the use of a three-dimensional filter that separates the complementarity error and the nonlinear infeasibility.

The SLPEC-EQP method has an important advantage over NLP reformulations: the solution of the LPEC matches exactly the definition of B-stationarity, and we therefore always work with the correct tangent cone. In particular, if the zero vector solves the LPEC, then we can conclude that the current point is B-stationary. To our knowledge, this algorithm is the only one that guarantees global convergence to B-stationary points.

12.3.3 Globally Optimal Methods for MPECs

The methods discussed above typically guarantee convergence to a stationary point at best. They do not guarantee convergence to a global minimum, even if the problem functions, f, c, G, and H are convex, because the feasible set of even the simplest complementarity constraint, $0 \leq x_1 \perp x_2 \geq 0$, is nonconvex.

Here, we briefly summarize existing results for obtaining global solutions to certain classes of MPECs. We limit our discussion to cases where the lower-level follower's problems is convex. One approach to obtaining global solutions would be to simply apply global optimization techniques to the nonlinear program (12.2.1). However, state-of-the-art solvers such as BARON [107, 115] or Couenne [12] require finite bounds on all variables to construct valid underestimators, and it is not clear that such bounds are easy to obtain on the multipliers. If we assume that such bounds exist, then we can apply BARON and Couenne. Unfortunately, the effectiveness of these solvers is limited to a few dozen variables at most.

If the MPEC has some special structure, then we can employ formulations and methods that guarantee convergence to a global minimum. One example are LPECs (or QPECs), when $f(z)$ is a linear (or a convex quadratic function), $c(z) = Az - b$ is an affine function, and the complementarity constraint is affine, i.e. $0 \leq G(z) = Mz - c \perp Nz - d = H(z) \geq 0$. Then it is possible to model the complementarity condition using binary variables, $y \in \{0, 1\}^p$, as

$$\Theta y \geq Mz - c \geq 0 \text{ and } (1 - \Theta)y \geq Nz - d \geq 0,$$

where $\Theta > 0$ is an upper bound on $Mz - c$ and $Nz - d$ that can be computed by solving an LP. Unfortunately, the resulting mixed-integer linear program often has a very weak continuous relaxation, resulting in massive branch-and-cut trees. Moreover, numerical issues can cause both $Mz - c$ and $Nz - d$ to be positive. In [8, 67], the authors extend a logical Benders decomposition technique [65] to this class of MPECs that avoids these pitfalls. The approach is based on a minimax principle and derives valid inequalities from LP relaxations. These approaches easily generalize to MPECs with more general (convex) $f(x)$ and $c(x)$.

12.4 Software Environments

Several modeling languages and solvers support bilevel optimization problems and MPECs. Table 12.1 presents a list of them. GAMS/EMP [53] and Pyomo [62] directly support bilevel optimization problems—they provide constructs that allow users to formulate bilevel optimization problems in their natural algebraic form (12.1.1) without applying any reformulations. GAMS introduced an extended mathematical programming (EMP) framework in which users can annotate the variables, functions, and constraints of a model to specify to which level, either upper or lower, they belong. In this case, a single monolithic model is defined, and annotations are provided via a separate text file, called the `empinfo` file. Pyomo takes a different approach by requiring users to define the lower-level problem explicitly as a model using their `Submodel()` construct and to link it to the upper-level model. In this way, not only bilevel optimization problems, but also multilevel

Table 12.1 Modeling languages and solvers that support bilevel or MPECs

Modeling language		Solver	
Bilevel	MPECs	Bilevel	MPECs
GAMS/EMP [53]	AMPL [50]	None	FilterMPEC [44]
Pyomo [62]	AIMMS [106]		KNITRO [7]
	GAMS [55]		NLPEC [55]
	Julia/JuMP [79]		Pyomo [61]
	Pyomo [61]		

problems [92] can be specified by defining submodels recursively and linking them together.

In contrast to bilevel optimization problems, most modeling languages support MPECs by providing a dedicated construct to take complementarity constraints in their natural form. AMPL [50] and Julia [79] provide `complements` and `@complements` keywords, respectively, GAMS [55] has a dot . construct, AIMMS [106] defines `ComplementarityVariables`, and Pyomo [61] defines `Complementarity` along with a `complements` expression. All these constructs enable complementarity constraints to be written as first-class expressions so that users can seamlessly use them in their models.

Regarding solvers, to the best of our knowledge, no dedicated, robust, and large-scale solvers are available for bilevel optimization problems at this time. The aforementioned modeling languages for bilevel optimization problems transform these problems into a computationally tractable formulation as an MPEC or generalized disjunctive program and call the associated solvers. Bilevel-specific features, such as optimal value functions, are not exploited in these solution procedures.

Although there are some recent efforts [75, 76, 94] exploiting the value function to globally solve the bilevel problem (12.1.1) using a branch-and-bound scheme, even in the case when the lower-level problem is nonconvex, these methods require repeated global solutions of nonconvex subproblems to compute the lower bounds. This requirement makes these methods unsuitable for large-scale bilevel problems. In particular, the requirement for global optimality may not be relaxed easily as discussed in [75].

In the case of MPECs, a few solvers are available. Most reformulate the MPECs into nonlinear or mixed-integer programming problems by applying relaxation, penalization, or disjunction techniques to the complementarity constraints. FilterMPEC [44] and KNITRO [7] transform the given problem into an NLP and solve it using their own algorithms by taking special care with the complementarity constraints. NLPEC [55] and Pyomo [61] are metasolvers that apply a reformulation and invoke an NLP solver, rather than supplying their own NLP solver. In contrast to the NLP-based approaches, Pyomo additionally provides a disjunctive programming method that formulates the MPEC as a mixed-integer nonlinear program. We believe that a combination of modeling languages for bilevel programs and MPEC solvers is the most promising and viable approach for quickly prototyping and solving bilevel problems when the lower-level problem is convex.

A number of bilevel optimization problems and MPECs are available online. The GAMS/EMP library [54] contains examples of bilevel optimization problems written by using the EMP framework. GAMS also provides MPEC examples [38]. The MacMPEC collection [81] is a set of MPEC examples written in AMPL that are frequently used to test performance of MPEC algorithms. QPECgen [71] generates MPEC examples with quadratic objectives and affine variational inequality constraints for lower-level problems.

12.5 Extensions

Our focus in this survey is on the optimistic bilevel optimization problem where the lower-level optimization problem is convex for all upper-level decisions. This approach contrasts with the pessimistic bilevel optimization problem [34, 35, 89, 117] where the leader plans for the worst by choosing an element of the solution set leading to the worst outcome and results in a problem of the form

$$\begin{aligned} \underset{x,\theta}{\text{minimize}} \quad & \theta \\ \text{subject to} \quad & f(x,y) \le \theta \quad \forall\, y \in Y(x) \\ & c(x,y) \ge 0 \quad \forall\, y \in Y(x), \end{aligned} \tag{12.5.1}$$

where $Y(x)$ is the solution set of the lower-level optimization problem parameterized by x. This formulation is consistent with robust optimization problems [13] with a complicated uncertainty set $Y(x)$. The two robustness constraints determine the worst objective function value and guarantee satisfaction of the joint feasibility constraint, respectively.

If the lower-level optimization problem is nonconvex, then the global solution of the bilevel optimization problem (12.1.1) may not even be a stationary point for the MPEC reformulation (12.1.2) because the MPEC reformulation is a relaxation of (12.1.1) in these cases; see [93, Example 1]. In the nonconvex case, special algorithms can be devised by a two-layer bounding scheme: bounding the optimal value function of the lower-level problem and computing lower and upper bounds of the upper-level problem, see [75, 76, 94]. These bounds need to be refined as the algorithm progresses, to ensure convergence to a global solution.

Many other extensions of bilevel optimization problems were not covered in this survey, including problems with lower-level second-order cone programs [18, 19], stochastic bilevel optimization [1, 22], discrete bilevel optimization with integer variables in both the upper- and lower-level decisions [36, 96], multiobjective bilevel optimization [25, 39, 91, 100], and multilevel optimization problems [80, 86, 92, 116]. Alternatives to the MPEC reformulations also exist that we did not cover in this survey; see [37, 49, 95, 112–114] for semi-infinite reformulations and [32, 120] for nonsmooth reformulations based on the value function.

Acknowledgements This work was supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research under Contract No. DE-AC02-06CH11357 at Argonne National Laboratory.

References

1. S.M. Alizadeh, P. Marcotte, G. Savard, Two-stage stochastic bilevel programming over a transportation network. Transp. Res. Part B: Methodol. **58**, 92–105 (2013)
2. G.B. Allende, G. Still, Solving bilevel programs with the KKT-approach. Math. Program. **138**(1), 309–332 (2013)
3. M. Anitescu, On solving mathematical programs with complementarity constraints as nonlinear programs. Preprint ANL/MCS-P864-1200, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL (2000)
4. M. Anitescu, Global convergence of an elastic mode approach for a class of mathematical programs with complementarity constraints. Preprint ANL/MCS-P1143–0404, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL (2004)
5. M. Anitescu, On using the elastic mode in nonlinear programming approaches to mathematical programs with complementarity constraints. SIAM J. Optim. **15**(4), 1203–1236 (2005)
6. M. Anitescu, P. Tseng, S.J. Wright, Elastic-mode algorithms for mathematical programs with equilibrium constraints: global convergence and stationarity properties. Math. Program. **110**(2), 337–371 (2007)
7. Artelys, Artelys KNITRO user's manual. Available at https://www.artelys.com/docs/knitro/index.html
8. L. Bai, J.E. Mitchell, J.-S. Pang, On convex quadratic programs with linear complementarity constraints. Comput. Optim. Appl. **54**(3), 517–554 (2013)
9. J.F. Bard, Convex two-level optimization. Math. Program. **40**(1), 15–27 (1988)
10. J.F. Bard, *Practical Bilevel Optimization: Algorithms and Applications*, vol. 30 (Kluwer Academic Publishers, Dordrecht, 1998)
11. J.F. Bard, J.T. Moore, A branch and bound algorithm for the bilevel programming problem. SIAM J. Sci. Stat. Comput. **11**(2), 281–292 (1990)
12. P. Belotti, J. Lee, L. Liberti, F. Margot, A. Wächter, Branching and bounds tightening techniques for non-convex MINLP. Optim. Methods Softw. **24**(4–5), 597–634 (2009)
13. A. Ben-Tal, L. El Ghaoui, A. Nemirovski, *Robust Optimization* (Princeton University Press, Princeton, 2009)
14. J.F. Benders, Partitioning procedures for solving mixed-variable programming problems. Numerische Mathematik **4**, 238–252 (1962)
15. H. Benson, A. Sen, D.F. Shanno, R. Vanderbei, Interior-point algorithms, penalty methods and equilibrium problems. Comput. Optim. Appl. **34**(2), 155–182 (2006)
16. R.H. Byrd, J. Nocedal, R.A. Waltz, Knitro: an integrated package for nonlinear optimization, in *Large-Scale Nonlinear Optimization* (Springer, New York, 2006), pp. 35–59
17. R.H. Byrd, J. Nocedal, R.A. Waltz, Steering exact penalty methods for nonlinear programming. Optim. Methods Softw. **23**(2), 197–213 (2008)
18. X. Chi, Z. Wan, Z. Hao, The models of bilevel programming with lower level second-order cone programs. J. Inequal. Appl. **2014**(1), 168 (2014)
19. X. Chi, Z. Wan, Z. Hao, Second order sufficient conditions for a class of bilevel programs with lower level second-order cone programming problem. J. Ind. Manage. Optim. **11**(4), 1111–1125 (2015)

20. N.H. Chieu, G.M. Lee, Constraint qualifications for mathematical programs with equilibrium constraints and their local preservation property. J. Optim. Theory Appl. **163**(3), 755–776 (2014)
21. C.M. Chin, R. Fletcher, On the global convergence of an SLP-filter algorithm that takes EQP steps. Math. Program. **96**(1), 161–177 (2003)
22. S. Christiansen, M. Patriksson, L. Wynter, Stochastic bilevel programming in structural optimization. Struct. Multidiscipl. Optim. **21**(5), 361–371 (2001)
23. F.H. Clarke, A new approach to Lagrange multipliers. Math. Oper. Res. **1**(2), 165–174 (1976)
24. B. Colson, P. Marcotte, G. Savard, Bilevel programming: a survey. 4OR **3**(2), 87–107 (2005)
25. F.F. Dedzo, L.P. Fotso, C.O. Pieume, Solution concepts and new optimality conditions in bilevel multiobjective programming. Appl. Math. **3**(10), 1395 (2012)
26. V. DeMiguel, M.P. Friedlander, F.J. Nogales, S. Scholtes, A two-sided relaxation scheme for mathematical programs with equilibrium constraints. SIAM J. Optim. **16**(2), 587–609 (2005)
27. S. Dempe, *Foundations of Bilevel Programming* (Springer Science & Business Media, New York, 2002)
28. S. Dempe, Annotated bibliography on bilevel programming and mathematical programs with equilibrium constraints. Optimization **52**, 333–359 (2003)
29. S. Dempe, *Bilevel Optimization: Theory, Algorithms and Applications* (TU Bergakademie Freiberg, Fakultät für Mathematik und Informatik, 2018)
30. S. Dempe, J. Dutta, Is bilevel programming a special case of a mathematical program with complementarity constraints? Math. Program. **131**(1–2), 37–48 (2012)
31. S. Dempe, S. Franke, On the solution of convex bilevel optimization problems. Comput. Optim. Appl. **63**(3), 685–703 (2016)
32. S. Dempe, A.B. Zemkoho, The bilevel programming problem: reformulations, constraint qualifications and optimality conditions. Math. Program. **138**(1), 447–473 (2013)
33. S. Dempe, A.B. Zemkoho, KKT reformulation and necessary conditions for optimality in nonsmooth bilevel optimization. SIAM J. Optim. **24**(4), 1639–1669 (2014)
34. S. Dempe, B.S. Mordukhovich, A.B. Zemkoho, Necessary optimality conditions in pessimistic bilevel programming. Optimization **63**(4), 505–533 (2014)
35. S. Dempe, G. Luo, S. Franke, *Pessimistic Bilevel Linear Optimization* (Technische Universität Bergakademie Freiberg, Fakultät für Mathematik und Informatik, 2016)
36. S. Dempe, F. Mefo Kue, P. Mehlitz, Optimality conditions for mixed discrete bilevel optimization problems. Optimization **67**(6), 737–756 (2018)
37. M. Diehl, B. Houska, O. Stein, P. Steuermann, A lifting method for generalized semi-infinite programs based on lower level Wolfe duality. Comput. Optim. Appl. **54**(1), 189–210 (2013)
38. S.P. Dirkse, MPEC world, 2001. Available at http://www.gamsworld.org/mpec/mpeclib.htm
39. G. Eichfelder, Multiobjective bilevel optimization. Math. Program. **123**(2), 419–449 (2010)
40. H.-R. Fang, S. Leyffer, T. Munson, A pivoting algorithm for linear programming with linear complementarity constraints. Optim. Methods Softw. **27**(1), 89–114 (2012)
41. M.C. Ferris, F. Tin-Loi, On the solution of a minimum weight elastoplastic problem involving displacement and complementarity constraints. Comput. Methods Appl. Mech. Eng. **174**, 107–120 (1999)
42. M.L. Flegel, C. Kanzow, Abadie-type constraint qualification for mathematical programs with equilibrium constraints. J. Optim. Theory Appl. **124**(3), 595–614 (2005)
43. M.L. Flegel, C. Kanzow, A direct proof of M-stationarity under MPEC-GCQ for mathematical programs with equilibrium constraints, in *Optimization with Multivalued Mappings: Theory, Applications, and Algorithms*, ed. by S. Dempe, V. Kalashnikov (Springer, New York, 2006), pp. 111–122
44. R. Fletcher, S. Leyffer, FilterMPEC. Available at https://neos-server.org/neos/solvers/cp:filterMPEC/AMPL.html
45. R. Fletcher, S. Leyffer, Numerical experience with solving MPECs as NLPs. Numerical Analysis Report NA/210, Department of Mathematics, University of Dundee, Dundee, 2002
46. R. Fletcher, S. Leyffer, Solving mathematical program with complementarity constraints as nonlinear programs. Optim. Methods Softw. **19**(1), 15–40 (2004)

47. R. Fletcher, E. Sainz de la Maza, Nonlinear programming and nonsmooth optimization by successive linear programming. Math. Program. **43**, 235–256 (1989)
48. R. Fletcher, S. Leyffer, D. Ralph, S. Scholtes, Local convergence of SQP methods for mathematical programs with equilibrium constraints. SIAM J. Optim. **17**(1), 259–286 (2006)
49. C. Floudas, O. Stein, The adaptive convexification algorithm: a feasible point method for semi-infinite programming. SIAM J. Optim. **18**(4), 1187–1208 (2008)
50. R. Fourer, D.M. Gay, B.W. Kernighan, AMPL: a modeling language for mathematical programming, 2003. Available at https://ampl.com/resources/the-ampl-book/chapter-downloads
51. M. Fukushima, G.-H. Lin, Smoothing methods for mathematical programs with equilibrium constraints, in *International Conference on Informatics Research for Development of Knowledge Society Infrastructure, 2004. ICKS 2004.* (IEEE, New York, 2004), pp. 206–213
52. M. Fukushima, P. Tseng, An implementable active-set algorithm for computing a B-stationary point of the mathematical program with linear complementarity constraints. SIAM J. Optim. **12**, 724–739 (2002)
53. GAMS Development Corporation, Bilevel programming using GAMS/EMP. Available at https://www.gams.com/latest/docs/UG_EMP_Bilevel.html
54. GAMS Development Corporation, The GAMS EMP library. Available at https://www.gams.com/latest/emplib_ml/libhtml/index.html
55. GAMS Development Corporation, NLPEC (Nonlinear programming with equilibrium constraints). Available at https://www.gams.com/latest/docs/S_NLPEC.html
56. H. Gfrerer, Optimality conditions for disjunctive programs based on generalized differentiation with application to mathematical programs with equilibrium constraints. SIAM J. Optim. **24**(2), 898–931 (2014)
57. G. Giallombardo, D. Ralph, Multiplier convergence in trust-region methods with application to convergence of decomposition methods for MPECs. Math. Program. **112**(2), 335–369 (2008)
58. P.E. Gill, W. Murray, M.A. Saunders, SNOPT: an SQP algorithm for large-scale constrained optimization. SIAM J. Optim. **12**(4), 979–1006 (2002)
59. P.E. Gill, W. Murray, M.A. Saunders, SNOPT: an SQP algorithm for large-scale constrained optimization. SIAM Rev. **47**(1), 99–131 (2005)
60. P. Hansen, B. Jaumard, G. Savard, New branch-and-bound rules for linear bilevel programming. SIAM J. Sci. Stat. Comput. **13**(5), 1194–1217 (1992)
61. W. Hart, J.D. Siirola, Modeling mathematical programs with equilibrium constraints in Pyomo. Technical report, Sandia National Laboratories, Albuquerque, NM, 2015
62. W. Hart, R. Chen, J.D. Siirola, J.-P. Watson, Modeling bilevel programs in Pyomo. Technical report, Sandia National Laboratories, Albuquerque, NM, 2016
63. M. Hintermüller, T. Surowiec, A bundle-free implicit programming approach for a class of elliptic MPECs in function space. Math. Program. **160**(1), 271–305 (2016)
64. T. Hoheisel, C. Kanzow, A. Schwartz, Theoretical and numerical comparison of relaxation methods for mathematical programs with complementarity constraints. Math. Program. **137**(1), 257–288 (2013)
65. J.N. Hooker, G. Ottosson, Logic-based Benders decomposition. Math. Program. **96**(1), 33–60 (2003)
66. X. Hu, D. Ralph, Convergence of a penalty method for mathematical programming with complementarity constraints. J. Optim. Theory Appl. **123**(2), 365–390 (2004)
67. J. Hu, J. Mitchell, J.-S. Pang, K. Bennett, G. Kunapuli, On the global solution of linear programs with linear complementarity constraints. SIAM J. Optim. **19**(1), 445–471 (2008)
68. J. Hu, J. Mitchell, J.-S. Pang, B. Yu, On linear programs with linear complementarity constraints. J. Glob. Optim. **53**(1), 29–51 (2012)
69. A.F. Izmailov, M.V. Solodov, E.I. Uskov, Global convergence of augmented lagrangian methods applied to optimization problems with degenerate constraints, including problems with complementarity constraints. SIAM J. Optim. **22**(4), 1579–1606 (2012)

70. F. Jara-Moroni, J.-S. Pang, A. Wächter, A study of the difference-of-convex approach for solving linear programs with complementarity constraints. Math. Program. **169**(1), 221–254 (2018)
71. H. Jiang, D. Ralph, QPECgen, a MATLAB generator for mathematical programs with quadratic objectives and affine variational inequality constraints. Comput. Optim. Appl. **13**, 25–59 (1999)
72. K.L. Judd, C.-L. Su, Computation of moral-hazard problems, in *Society for Computational Economics, Computing in Economics and Finance*, 2005
73. C. Kanzow, A. Schwartz, Convergence properties of the inexact Lin-Fukushima relaxation method for mathematical programs with complementarity constraints. Comput. Optim. Appl. **59**(1), 249–262 (2014)
74. C. Kanzow, A. Schwartz, The price of inexactness: convergence properties of relaxation methods for mathematical programs with complementarity constraints revisited. Math. Oper. Res. **40**(2), 253–275 (2015)
75. P.-M. Kleniati, C.S. Adjiman, Branch-and-sandwich: a deterministic global optimization algorithm for optimistic bilevel programming problems. Part I: Theoretical development. J. Global Optim. **60**(3), 425–458 (2014)
76. P.-M. Kleniati, C.S. Adjiman, Branch-and-sandwich: a deterministic global optimization algorithm for optimistic bilevel programming problems. Part II: Convergence analysis and numerical results. J. Global Optim. **60**(3), 459–481 (2014)
77. M. Kojima, Strongly stable stationary solutions in nonlinear programming, in *Analysis and Computation of Fixed Points*, ed. by S.M. Robinson, pp. 93–138 (Academic, New York, 1980)
78. C.D. Kolstad, A review of the literature on bi-level mathematical programming. Technical report, Los Alamos National Laboratory Los Alamos, NM, 1985
79. C. Kwon, Complementarity package for Julia/JuMP. Available at https://github.com/chkwon/Complementarity.jl
80. K. Lachhwani, A. Dwivedi, Bi-level and multi-level programming problems: taxonomy of literature review and research issues. Arch. Comput. Methods Eng. **25**(4), 847–877 (2018)
81. S. Leyffer, MacMPEC: AMPL collection of MPECs, 2000. Available at https://wiki.mcs.anl.gov/leyffer/index.php/MacMPEC
82. S. Leyffer, Mathematical programs with complementarity constraints. SIAG/OPT Views-and-News **14**(1), 15–18 (2003)
83. S. Leyffer, Complementarity constraints as nonlinear equations: theory and numerical experience, in emphOptimization and Multivalued Mappings, ed. by S. Dempe, V. Kalashnikov (Springer, Berlin, 2006), pp. 169–208
84. S. Leyffer, T. Munson, A globally convergent filter method for MPECs. Preprint ANL/MCS-P1457-0907, Argonne National Laboratory, Mathematics and Computer Science Division, 2007
85. S. Leyffer, G. Lopez-Calva, J. Nocedal, Interior methods for mathematical programs with complementarity constraints. SIAM J. Optim. **17**(1), 52–77 (2006)
86. G. Li, Z. Wan, J.-W. Chen, X. Zhao, Necessary optimality condition for trilevel optimization problem. J. Ind. Manage. Optim. **41**, 282–290 (2018)
87. X. Liu, J. Sun, Generalized stationary points and an interior-point method for mathematical programs with equilibrium constraints. Math. Program. **101**(1), 231–261 (2004).
88. X. Liu, G. Perakis, J. Sun, A robust SQP method for mathematical programs with linear complementarity constraints. Comput. Optim. Appl. **34**, 5–33 (2006)
89. J. Liu, Y. Fan, Z. Chen, Y. Zheng, Pessimistic bilevel optimization: a survey. Int. J. Comput. Intell. Syst. **11**(1), 725–736 (2018)
90. Z.-Q. Luo, J.-S. Pang, D. Ralph, *Mathematical Programs with Equilibrium Constraints* (Cambridge University Press, Cambridge, 1996)
91. Y. Lv, Z. Wan, Solving linear bilevel multiobjective programming problem via exact penalty function approach. J. Inequal. Appl. **2015**(1), 258 (2015)
92. A. Migdalas, P.M. Pardalos, P. Värbrand (eds.) *Multilevel Optimization: Algorithms and Applications* (Kluwer Academic Publishers, Dordrecht, 1997)

93. J.A. Mirrlees, The theory of moral hazard and unobservable behaviour: part I. Rev. Econ. Stud. **66**(1), 3–21 (1999)
94. A. Mitsos, P. Lemonidis, P.I. Barton, Global solution of bilevel programs with a nonconvex inner program. J. Global Optim. **42**(4), 475–513 (2008)
95. A. Mitsos, P. Lemonidis, C. Lee, P.I. Barton, Relaxation-based bounds for semi-infinite programs. SIAM J. Optim. **19**(1), 77–113 (2008)
96. J.T. Moore, J.F. Bard, The mixed integer linear bilevel programming problem. Oper. Res. **38**(5), 911–921 (1990)
97. J.V. Outrata, Optimality conditions for a class of mathematical programs with equilibrium constraints. Math. Oper. Res. **24**(3), 627–644 (1999)
98. J.V. Outrata, M. Kočvara, J. Zowe, *Nonsmooth Approach to Optimization Problems with Equilibrium Constraints* (Kluwer Academic Publishers, Dordrecht, 1998)
99. J.-S. Pang, Three modeling paradigms in mathematical programming. Math. Program. **125**(2), 297–323 (2010)
100. C.O. Pieume, P. Marcotte, L.P. Fotso, P. Siarry, Solving bilevel linear multiobjective programming problems. Am. J. Oper. Res. **1**(4), 214–219 (2011)
101. E.S. Prescott, A primer on moral-hazard models. Feder. Reserve Bank Richmond Q. Rev. **85**, 47–77 (1999)
102. L. Qi, Z. Wei, On the constant positive linear dependence condition and its application to SQP methods. SIAM J. Optim. **10**(4), 963–981 (2000)
103. A. Raghunathan, L.T. Biegler, An interior point method for mathematical programs with complementarity constraints (MPCCs). SIAM J. Optim. **15**(3), 720–750 (2005)
104. D. Ralph, S.J. Wright, Some properties of regularization and penalization schemes for MPECs. Optim. Methods Softw. **19**(5), 527–556 (2004)
105. S.M. Robinson, Strongly regular generalized equations. Math. Oper. Res. **5**(1), 43–62 (1980)
106. M. Roelofs, J. Bisschop, The AIMMS language reference. Available at https://aimms.com/english/developers/resources/manuals/language-reference
107. N.V. Sahinidis, BARON 17.8.9: Global Optimization of Mixed-Integer Nonlinear Programs, User's Manual, 2017
108. H. Scheel, S. Scholtes, Mathematical program with complementarity constraints: stationarity, optimality and sensitivity. Math. Oper. Res. **25**, 1–22 (2000)
109. S. Scholtes, Convergence properties of a regularization schemes for mathematical programs with complementarity constraints. SIAM J. Optim. **11**(4), 918–936 (2001)
110. S. Scholtes, Nonconvex structures in nonlinear programming. Oper. Res. **52**(3), 368–383 (2004)
111. K. Shimizu, Y. Ishizuka, J.F. Bard, *Nondifferentiable and Two-level Mathematical Programming* (Kluwer Academic Publishers, Dordrecht, 1997)
112. O. Stein, *Bi-Level Strategies in Semi-Infinite Programming*, vol. 71 (Springer Science and Business Media, New York, 2013)
113. O. Stein, P. Steuermann, The adaptive convexification algorithm for semi-infinite programming with arbitrary index sets. Math. Program. **136**(1), 183–207 (2012)
114. O. Stein, A. Winterfeld, Feasible method for generalized semi-infinite programming. J. Optim. Theory Appl. **146**(2), 419–443 (2010)
115. M. Tawarmalani, N.V. Sahinidis, A polyhedral branch-and-cut approach to global optimization. Math. Program. **103**, 225–249 (2005)
116. L.N. Vicente, P.H. Calamai, Bilevel and multilevel programming: a bibliography review. J. Global Optim. **5**(3), 291–306 (1994)
117. W. Wiesemann, A. Tsoukalas, P.-M. Kleniati, B. Rustem, Pessimistic bilevel optimization. SIAM J. Optim. **23**(1), 353–380 (2013)
118. H. Xu, An implicit programming approach for a class of stochastic mathematical programs with complementarity constraints. SIAM J. Optim. **16**(3), 670–696 (2006)

119. J.J. Ye, J. Zhang, Enhanced Karush-Kuhn-Tucker conditions for mathematical programs with equilibrium constraints. J. Optim. Theory Appl. **163**(3), 777–794 (2014)
120. J.J. Ye, D. Zhu, New necessary optimality conditions for bilevel programs by combining the MPEC and value function approaches. SIAM J. Optim. **20**(4), 1885–1905 (2010)