

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.

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:

$$\begin{aligned} & \underset{x,y}{\text{minimize}} && f(x, y) \\ & \text{subject to} && c(x, y) \geq 0 \\ & && y \in \underset{v}{\text{argmin}} \{ g(x, v) \text{ subject to } d(x, v) \geq 0 \}, \end{aligned} \tag{1}$$

Youngdae Kim
Argonne National Laboratory, Lemont, IL 60439, e-mail: youngdae@anl.gov
Sven Leyffer
Argonne National Laboratory, Lemont, IL 60439, e-mail: leyffer@anl.gov
Todd Munson
Argonne National Laboratory, Lemont, IL 60439, e-mail: tmunson@anl.gov

* Argonne National Laboratory, MCS Division Preprint, ANL/MCS-P9195-0719

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 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.

1.1 Applications and Illustrative Example

Applications of bilevel optimization problems arise in economics and engineering domains; see [10, 23, 29–31, 76, 90, 108, 112] and the references therein. As an example bilevel optimization problem, we consider the moral-hazard problem in economics [70, 99] 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 $q \in \mathcal{Q}$ and \mathcal{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 $\{c_q\}_{q \in \mathcal{Q}}$ 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:

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

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.

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) [80, 88, 96]:

$$\begin{aligned}
 & \underset{x,y,\lambda}{\text{minimize}} && f(x, y) \\
 & \text{subject to} && c(x, y) \geq 0 \\
 & && \nabla_y g(x, y) - \nabla_y d(x, y)\lambda = 0 \\
 & && 0 \leq \lambda \perp d(x, y) \geq 0,
 \end{aligned} \tag{2}$$

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.

In the convex case, the relationship between the original bilevel optimization problem (1) and the MPEC formulation (2) is nontrivial, as demonstrated in [32, 37] for smooth functions and [38] for nonsmooth functions. These papers show that a global (local) solution to the bilevel optimization problem (1) corresponds to a global (local) solution to the MPEC (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 (2) corresponds to a global solution to the bilevel optimization problem (1). A local solution to the MPEC (2) may not correspond to a local solution to the bilevel optimization problem (1) unless more assumptions are made; see [32, 37] 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 [33], and we have observed computational difficulties when applying MPEC algorithms to solve the Fritz-John reformulation.

1.3 MPEC Problems

Generically, we write MPECs such as (2) as

$$\begin{aligned}
& \underset{z}{\text{minimize}} && f(z) \\
& \text{subject to} && c(z) \geq 0 \\
& && 0 \leq G(z) \perp H(z) \geq 0,
\end{aligned} \tag{3}$$

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 (3) becomes a standard nonlinear program (NLP). Unfortunately, the resulting problem violates the Mangasarian-Fromowitz constraint qualification (MFCQ) at any feasible point [105]. 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 [97].

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 Section 3, we discuss algorithms for solving MPECs. In Section 4, we focus on software environments for specifying and solving bilevel optimization problems and mathematical programs with equilibrium constraints, before providing pointers in Section 5 to related topics we do not cover in this survey.

2 MPEC Theory

We now survey stationarity conditions, constraint qualifications, and regularity for the MPEC (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 (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 [115].

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.

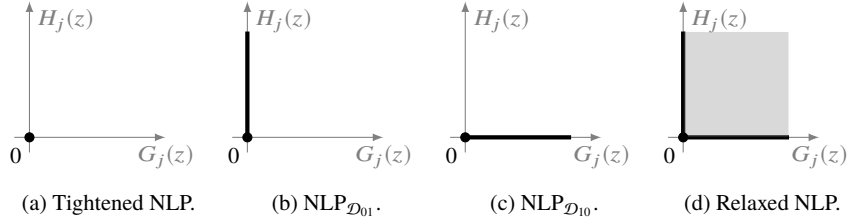


Fig. 1: Feasible regions of the NLPs associated with a complementarity constraint $0 \leq G_j(z) \perp H_j(z) \geq 0$ for $j \in \mathcal{D}$.

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:

$$\begin{aligned}
 & \underset{z}{\text{minimize}} && f(z) \\
 & \text{subject to} && c_i(z) \geq 0 && \forall i = 1, \dots, m \\
 & && G_j(z) \geq 0, H_j(z) \geq 0, G_j(z)H_j(z) \leq 0 && \forall j = 1, \dots, p,
 \end{aligned} \tag{4}$$

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, (4) violates the MFCQ at any feasible point [105] 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 (4).

Instead, stationarity concepts are derived from several different approaches: local analysis of the NLPs associated with an MPEC [88, 105]; Clarke's nonsmooth analysis to the complementarity constraints by replacing them with the min function [22, 105] and Mordukhovich's generalized differential calculus applied to the generalized normal cone [95]. 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

$$\mathcal{D} := \{j \mid G_j(z) = H_j(z) = 0\}, \quad \mathcal{D} = \mathcal{D}_{01} \cup \mathcal{D}_{10}, \quad \mathcal{D}_{01} \cap \mathcal{D}_{10} = \emptyset. \tag{5}$$

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

$$\begin{aligned}
& \underset{z}{\text{minimize}} && f(z) \\
& \text{subject to} && c_i(z) \geq 0 && \forall i = 1, \dots, m \\
& && G_j(z) = 0 && \forall j : G_j(z) = 0, H_j(z) > 0 \\
& && H_j(z) = 0 && \forall j : G_j(z) > 0, H_j(z) = 0 \\
& && G_j(z) = 0, H_j(z) \geq 0 && \forall j \in \mathcal{D}_{01} \\
& && G_j(z) \geq 0, H_j(z) = 0 && \forall j \in \mathcal{D}_{10},
\end{aligned} \tag{6}$$

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

$$\mathcal{T}(z^*) := \left\{ d \mid d = \lim_{t_k \downarrow 0} \frac{z^k - z^*}{t_k} \text{ for some } \{z^k\}, z^k \rightarrow z^*, z^k \in \mathcal{F}, \forall k \right\}. \tag{7}$$

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:

$$\mathcal{T}_{\text{MPEC}}(z) := \bigcup_{(\mathcal{D}_{01}, \mathcal{D}_{10})} \mathcal{T}_{\text{NLP}_{\mathcal{D}_{01}, \mathcal{D}_{10}}}(z). \tag{8}$$

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 [105] and relaxed [49, 105] NLPs, are defined by replacing the last two conditions of (6) 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 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

$$\mathcal{T}_{\text{TNLP}}(z) \subseteq \mathcal{T}_{\text{MPEC}}(z) \subseteq \mathcal{T}_{\text{RNLP}}(z). \tag{9}$$

When $\mathcal{D} = \emptyset$, the equality holds throughout (9), and *lower-level strict complementarity* [102] holds. From (9), a local minimizer of the RNLP is a local minimizer of the MPEC, but not vice versa [105].

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:

$$\begin{aligned}
& \underset{z}{\text{minimize}} && \nabla f(z^*)^T d \\
& \text{subject to} && d \in \mathcal{T}_{\text{MPEC}}^{\text{lin}}(z^*),
\end{aligned} \tag{10}$$

where

$$\begin{aligned}
\mathcal{T}_{\text{MPEC}}^{\text{lin}}(z^*) := \{d \mid & \nabla c_i(z^*)^T d \geq 0, \forall i : c_i(z^*) = 0, \\
& \nabla G_j(z^*)^T d = 0, \forall j : G_j(z^*) = 0, H_j(z^*) > 0, \\
& \nabla H_j(z^*)^T d = 0, \forall j : G_j(z^*) > 0, H_j(z^*) = 0, \\
& 0 \leq \nabla G_j(z^*)^T d \perp \nabla H_j(z^*)^T d \geq 0, \forall j \in \mathcal{D}(z^*)\}.
\end{aligned} \tag{11}$$

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 Section 2.2.

Since $\mathcal{T}_{\text{MPEC}}(z^*) \subseteq \mathcal{T}_{\text{MPEC}}^{\text{lin}}(z^*)$ [44], B-stationarity implies geometric B-stationarity, but not vice versa. A similar equivalence (8) and inclusion relationship (9) hold between the linearized cones of its associated NLPs [44].

The next important stationarity concept is strong stationarity.

Definition 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 $\hat{\nu}_{1j}, \hat{\nu}_{2j} \geq 0$ for $j \in \mathcal{D}(z^*)$. Strong stationarity implies B-stationarity due to (9) and the inclusion relationship between the tangent and linearized cones. Equivalence of stationarity between (4) and the RNLP was shown in [5, 49, 101].

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 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 program (6). 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 [55] 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 $d \in \mathcal{T}_{\text{MPEC}}^{\text{lin}}(z^*)$ 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 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 2, the Lagrangian \mathcal{L} and the critical cone for the RNLP are defined as follows:

$$\begin{aligned} \mathcal{L}(z, \lambda, \hat{\nu}_1, \hat{\nu}_2) &:= f(z) - \sum_{i \in \mathcal{E} \cup \mathcal{I}} c_i(z) \lambda_i - \sum_{j=1}^p G_j(z) \hat{\nu}_{1j} - \sum_{j=1}^p H_j(z) \hat{\nu}_{2j}, \\ \mathcal{C}_{\text{RNLP}}(z^*) &:= \{d \mid \begin{aligned} &\nabla c_i(z^*)^T d = 0, \quad \forall i : \lambda_i^* > 0, \\ &\nabla c_i(z^*)^T d \geq 0, \quad \forall i : c_i(z^*) = 0, \lambda_i^* = 0, \\ &\nabla G_j(z^*)^T d = 0, \quad \forall j : G_j(z^*) = 0, H_j(z^*) > 0, \\ &\nabla H_j(z^*)^T d = 0, \quad \forall j : G_j(z^*) > 0, H_j(z^*) = 0, \\ &\nabla G_j(z^*)^T d = 0, \quad \forall j \in \mathcal{D}(z^*), \hat{\nu}_{1j}^* > 0, \\ &\nabla G_j(z^*)^T d \geq 0, \quad \forall j \in \mathcal{D}(z^*), \hat{\nu}_{1j}^* = 0, \\ &\nabla H_j(z^*)^T d = 0, \quad \forall j \in \mathcal{D}(z^*), \hat{\nu}_{2j}^* > 0, \\ &\nabla H_j(z^*)^T d \geq 0, \quad \forall j \in \mathcal{D}(z^*), \hat{\nu}_{2j}^* = 0 \}. \end{aligned} \quad (12) \end{aligned}$$

Definition 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 $d^T \nabla_{zz}^2 \mathcal{L}(z^*, \lambda^*, \hat{\nu}_1^*, \hat{\nu}_2^*) d \geq \omega$ for all $d \neq 0, d \in \mathcal{C}_{\text{RNLP}}(z^*)$. If the conditions hold for $\mathcal{C}_{\text{MPEC}}(z^*)$ instead of $\mathcal{C}_{\text{RNLP}}(z^*)$, where $\mathcal{C}_{\text{MPEC}}(z^*) := \mathcal{C}_{\text{RNLP}}(z^*) \cap \{d \mid \min(\nabla G_j(z^*)^T d, \nabla H_j(z^*)^T d) = 0, \forall j \in \mathcal{D}(z^*), \hat{\nu}_{1j}^* = \hat{\nu}_{2j}^* = 0\}$, then we say that MPEC-SOSC is satisfied.

We note that $d \in \mathcal{C}_{\text{MPEC}}(z^*)$ if and only if it is a critical direction of any of the NLP $_{\mathcal{D}_{01}, \mathcal{D}_{10}}(z^*)$'s at z^* [102, 105]. 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 [75, 103] 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 Section 3. A critical cone $\mathcal{C}_{\text{RNLP}}^S(z^*)$ is used instead of $\mathcal{C}_{\text{RNLP}}(z^*)$, which expands it by removing the feasible directions for inequalities associated with zero multipliers:

$$\begin{aligned} \mathcal{C}_{\text{RNLP}}^S(z^*) &:= \{d \mid \begin{aligned} &\nabla c_i(z^*)^T d = 0, \quad \forall i : \lambda_i^* > 0, \\ &\nabla G_j(z^*)^T d = 0, \quad \forall j : \hat{\nu}_{1j}^* \neq 0, \\ &\nabla H_j(z^*)^T d = 0, \quad \forall j : \hat{\nu}_{2j}^* \neq 0 \}. \end{aligned} \quad (13) \end{aligned}$$

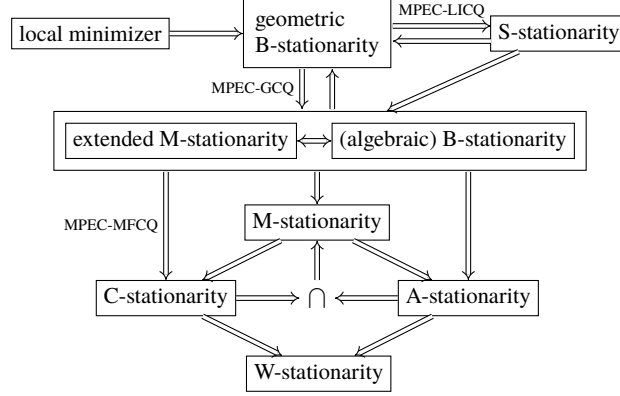


Fig. 2: Relationships between stationarity concepts where M-stationarity is the intersection of C- and A-stationarity.

Definition 3 RNLP-SSOSC is satisfied at a strongly stationary point z^* with its multipliers $(\lambda^*, \hat{v}_1^*, \hat{v}_2^*)$ if there exists a constant $\omega > 0$ such that $d^T \nabla_{zz}^2 \mathcal{L}(z^*, \lambda^*, \hat{v}_1^*, \hat{v}_2^*) d \geq \omega$ for all $d \neq 0, d \in C_{\text{RNLP}}^S(z^*)$.

By definition, we have an inclusion relationship between SOSCs: RNLP-SSOSC \Rightarrow RNLP-SOSC \Rightarrow MPEC-SOSC. Reverse directions do not hold in general; an example was presented in [105] showing that MPEC-SOSC $\not\Rightarrow$ RNLP-SOSC.

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 Section 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 4, LICQ denotes linear independence constraint qualification, and CPLD represents constant positive linear dependence [100].

Definition 4 An MPEC (2) is said to satisfy an *MPEC-LICQ* (*MPEC-MFCQ* and *MPEC-CPLD*) at a feasible point z if the corresponding tightened NLP satisfies an LICQ (MFCQ and CPLD) at z .

We note that MPEC-LICQ holds at a feasible point z if and only if all of its associated NLPs satisfy an 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 [105] 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 5 The MPEC (3) is said to satisfy an *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 [45, 55].

Local preservation of constraint qualifications has been studied in [19], 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 \Rightarrow MPEC-MFCQ \Rightarrow MPEC-CPLD \Rightarrow MPEC-ACQ \Rightarrow MPEC-GCQ.

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 (3) can be categorized into three broad classes:

1. **Nonlinear programming methods** that rewrite the complementarity constraint in (3) as a nonlinear set of inequalities, such as

$$G(z) \geq 0, H(z) \geq 0, \text{ and } G(z)^T H(z) \leq 0, \quad (14)$$

and then apply NLP techniques; see, for example, [28, 47, 72, 87, 101, 102, 106]. 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 meth-

ods [42], branch-and-cut methods [8, 9, 11], and active-set methods [56, 83, 86]. 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 (2), and use the KKT conditions to eliminate the $(y(x), \lambda(x))$, resulting in a reduced nonsmooth problem

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

that can be solved by using methods for nonsmooth optimization. See the monograph [96] and the references [62, 114] for more details.

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

3.1 NLP Methods for MPECs

NLP methods are attractive because they allow us to leverage powerful numerical solvers. Unfortunately, the system (14) violates a standard stability assumption for NLP at any feasible point. In [105], the authors show that (14) violates the MFCQ at any feasible point. Other nonlinear reformulations of the complementarity constraint (14) are possible. In [48, 81], the authors experiment with a range of reformulations using different nonlinear complementarity functions, but they observe that the formulation (14) 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 (14) using slacks as

$$s_1 = G(z), \quad s_2 = H(z) \geq 0, \quad s_1 \geq 0, \quad s_2 \geq 0, \quad \text{and} \quad s_1^T s_2 \leq 0. \quad (15)$$

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 [49] 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, [88]. In practice [47], 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.

3.1.1 Relaxation-Based NLP Methods

An attractive way to solve MPECs is to relax the complementarity constraint in (14) by using a positive relaxation parameter, $t > 0$:

$$\begin{aligned} & \underset{z,u,v}{\text{minimize}} && f(z) \\ & \text{subject to} && c_i(z) \geq 0 \\ & && G_j(z) \geq 0, H_j(z) \geq 0, G_j(z)H_j(z) \leq t \quad \forall j = 1, \dots, p. \end{aligned} \tag{16}$$

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 [102, 106]. 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 (16) have been proposed [87, 101] 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 (16) has no strict interior.

An alternative regularization scheme is proposed in [72], based on the reformulation of (14) as

$$G_j(z) \geq 0, H_j(z) \geq 0, \text{ and } \phi(G_j(z), H_j(z), t) \leq 0, \tag{17}$$

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 [72], 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 [63], with later methods in [71]. 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 [28]. The scheme is motivated by the observation that the complementarity constraint in the slack formulation (15) 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

$$s_{1j} \geq -\delta_{1j}, \quad s_{2j} \geq -\delta_{2j}, \quad s_{1j}s_{2j} \leq \delta_{c_j}.$$

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.

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 [43] and penalizes the complementarity constraint in the objective function, after introducing slacks:

$$\begin{aligned} & \underset{z, u, v}{\text{minimize}} && f(z) + \pi s_1^T s_2 \\ & \text{subject to} && c_i(z) \geq 0 && \forall i = 1, \dots, m \\ & && G_j(z) - s_{1j} = 0, H_j(z) - s_{2j} = 0 && \forall j = 1, \dots, p \\ & && s_2 \geq 0, s_1 \geq 0 \end{aligned} \quad (18)$$

for positive penalty parameters 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 [43]. It has been adopted as a heuristic to solve MPECs with interior methods in LOQO by [14], 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 [82], the authors study a general interior-point method for solving (18). 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

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

A related approach to penalty functions is the elastic mode that is implemented in SNOPT [57, 58]. The elastic approach [6] combines both a relaxation of the constraints and penalization of the complementarity conditions to solve a sequence of optimization problems

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

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 [53, 67]. 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 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 Section 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 [72] 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 descend 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.

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 complementarity constraints (LPCCs), 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, 59, 64, 65], pivot-based methods for computing B-stationary points [42], and penalty methods based on a difference of convex functions formulation [68]. 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 [54], sequential quadratic programming [86], and sequential linear programming [56] 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 (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 [64] for solving LPCCs. The algorithm is based on equivalent mixed-integer reformulations and the application of Benders decomposition [13] with linear programming (LP) relaxations. In contrast, the authors in [42] generalize LP pivoting techniques to locally solve LPCCs. The authors prove convergence to a B-stationary point but unlike [64] do not obtain globally optimal solutions.

The SQPEC approach extends SQP methods by taking special care of the complementarity constraint [107]. 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 [83]:

$$\underset{x,y}{\text{minimize}} (x-1)^2 + y^3 + y^2 \quad \text{subject to } 0 \leq x \perp y \geq 0. \quad (19)$$

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 descend 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, [15, 16, 20, 50]. The method is motivated by considering the linearized tangent cone,

$\mathcal{T}_{\text{MPEC}}^{\text{lin}}(z^*)$ in (11) as a direction-finding problem. This method solves a sequence of LPCCs inside a trust region [83] with radius Δ around the current point z :

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

The LPCC need be solved only locally, and the pivoting method of [42] is a practical and efficient method of solving the LPCC. Given a solution $d \neq 0$, we find the active sets that are predicted by the LPCC,

$$\mathcal{A}_c(z + d) := \{i : c_i(z) + \nabla c_i(z)^T d = 0\} \quad (20)$$

$$\mathcal{A}_G(z + d) := \{j : G_j(z) + \nabla G_j(z)^T d = 0\} \quad (21)$$

$$\mathcal{A}_H(z + d) := \{j : H_j(z) + \nabla H_j(z)^T d = 0\}, \quad (22)$$

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

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

We note that $\text{EQP}(z + d)$ 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 LPCC 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 LPCC, 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.

4 Software Environments

Several modeling languages and solvers support bilevel optimization problems and MPECs. Table 1 presents a list of them. GAMS/EMP [24] and Pyomo [60] directly support bilevel optimization problems – they provide constructs that allow users to formulate bilevel optimization problems in their natural algebraic form (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 problems [90] 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 [52] and Julia [77] provide `complements` and `@complements` keywords, respectively, GAMS [26] has a dot `.` construct, AIMMS [104] 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.

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 [46] 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 [26] and Pyomo [61] are metasolvers that apply a reformulations 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.

A number of bilevel optimization problems and MPECs are available online. The GAMS/EMP library [25] contains examples of bilevel optimization problems written by using the EMP framework. GAMS also provides MPEC examples [40]. The MacMPEC collection [79] is a set of MPEC examples written in AMPL that are frequently used to test performance of MPEC algorithms. QPECgen [69] generates MPEC examples with quadratic objectives and affine variational inequality constraints for lower-level problems.

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, 36, 85, 113] where the leader plans for the worst by choosing an element of the solution set leading

Table 1: Modeling languages and solvers that support bilevel or MPECs.

Modeling Language		Solver	
Bilevel	MPECs	Bilevel	MPECs
GAMS/EMP [24]	AMPL [52]	None	FilterMPEC [46]
Pyomo [60]	AIMMS [104]		KNITRO [7]
	GAMS [26]		NLPEC [26]
	Julia/JuMP [77]		Pyomo [61]
	Pyomo [61]		

to the worst outcome and results in a problem of the form

$$\begin{aligned}
 & \underset{x, \theta}{\text{minimize}} && \theta \\
 & \text{subject to} && f(x, y) \leq \theta \quad \forall y \in Y(x) \\
 & && c(x, y) \geq 0 \quad \forall y \in Y(x),
 \end{aligned} \tag{23}$$

where $Y(x)$ is the solution set of the lower-level optimization problem parameterized by x . This formulation is consistent with robust optimization problems [12] 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 (1) may not be a stationary point for the MPEC reformulation (2); see [91]. In the nonconvex case, special algorithms can be devised by using convexifications of the lower-level problem [73, 74, 92]. These convexifications need to be refined as the algorithm progresses, and convergence to a global solution to the lower-level problem is assured.

Many other extensions of bilevel optimization problems were not covered in this survey, including problems with lower-level second-order cone programs [17, 18], stochastic bilevel optimization [1, 21], discrete bilevel optimization with integer variables in both the upper- and lower-level decisions [35, 94], multiobjective bilevel optimization [27, 41, 89, 98], and multilevel optimization problems [78, 84, 90, 112]. Alternatives to the MPEC reformulations also exist that we did not cover in this survey; see [39, 51, 93, 109–111] for semi-infinite reformulations and [37, 116] 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, and G. Savard. Two-stage stochastic bilevel programming over a transportation network. *Transportation Research Part B: Methodological*, 58:92–105, 2013.
2. G. B. Allende and G. Still. Solving bilevel programs with the KKT-approach. *Mathematical Programming*, 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 Journal on Optimization*, 15(4):1203–1236, 2005.
6. M. Anitescu, P. Tseng, and S. J. Wright. Elastic-mode algorithms for mathematical programs with equilibrium constraints: global convergence and stationarity properties. *Mathematical Programming*, 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, and J.-S. Pang. On convex quadratic programs with linear complementarity constraints. *Computational Optimization and Applications*, 54(3):517–554, 2013.
9. J. F. Bard. Convex two-level optimization. *Mathematical Programming*, 40(1):15–27, 1988.
10. J. F. Bard. *Practical Bilevel Optimization: Algorithms and Applications*, volume 30. Kluwer Academic Publishers, 1998.
11. J. F. Bard and J. T. Moore. A branch and bound algorithm for the bilevel programming problem. *SIAM Journal on Scientific and Statistical Computing*, 11(2):281–292, 1990.
12. A. Ben-Tal, L. El Ghaoui, and A. Nemirovski. *Robust Optimization*. Princeton University Press, 2009.
13. J. F. Benders. Partitioning procedures for solving mixed-variable programming problems. *Numerische Mathematik*, 4:238–252, 1962.
14. H. Benson, A. Sen, D. F. Shanno, and R. Vanderbei. Interior-point algorithms, penalty methods and equilibrium problems. Technical Report ORFE-03-02, Princeton University, Operations Research and Financial Engineering, October 2003. To appear in *Computational Optimization and Applications*.
15. R. H. Byrd, J. Nocedal, and R. A. Waltz. Knitro: an integrated package for nonlinear optimization. In *Large-Scale Nonlinear Optimization*, pages 35–59. Springer, 2006.
16. R. H. Byrd, J. Nocedal, and R. A. Waltz. Steering exact penalty methods for nonlinear programming. *Optimization Methods and Software*, 23(2):197–213, 2008.
17. X. Chi, Z. Wan, and Z. Hao. The models of bilevel programming with lower level second-order cone programs. *Journal of Inequalities and Applications*, 2014(1):168, 2014.
18. X. Chi, Z. Wan, and Z. Hao. Second order sufficient conditions for a class of bilevel programs with lower level second-order cone programming problem. *Journal of Industrial and Management Optimization*, 11(4):1111–1125, 2015.
19. N. H. Chieu and G. M. Lee. Constraint qualifications for mathematical programs with equilibrium constraints and their local preservation property. *Journal of Optimization Theory and Applications*, 163(3):755–776, 2014.
20. C. M. Chin and R. Fletcher. On the global convergence of an SLP-filter algorithm that takes EQP steps. *Mathematical Programming*, 96(1):161–177, 2003.
21. S. Christiansen, M. Patriksson, and L. Wynter. Stochastic bilevel programming in structural optimization. *Structural and multidisciplinary optimization*, 21(5):361–371, 2001.
22. F. H. Clarke. A new approach to Lagrange multipliers. *Mathematics of Operations Research*, 1(2):165–174, 1976.

23. B. Colson, P. Marcotte, and G. Savard. Bilevel programming: a survey. *4OR*, 3(2):87–107, 2005.
24. GAMS Development Corporation. Bilevel programming using GAMS/EMP. Available at https://www.gams.com/latest/docs/UG_EMP_Bilevel.html.
25. GAMS Development Corporation. The GAMS EMP library. Available at https://www.gams.com/latest/emplib_ml/libhtml/index.html.
26. GAMS Development Corporation. NLPEC (Nonlinear programming with equilibrium constraints). Available at https://www.gams.com/latest/docs/S_NLPEC.html.
27. F. F. Dedzo, L. P. Fotso, and C. O. Pieume. Solution concepts and new optimality conditions in bilevel multiobjective programming. *Applied Mathematics*, 3(10):1395, 2012.
28. V. DeMiguel, M. P. Friedlander, F. J. Nogales, and S. Scholtes. A two-sided relaxation scheme for mathematical programs with equilibrium constraints. *SIAM J. on Optimization*, 16(2):587–609, 2005.
29. S. Dempe. *Foundations of Bilevel Programming*. Springer Science & Business Media, 2002.
30. S. Dempe. Annotated bibliography on bilevel programming and mathematical programs with equilibrium constraints. *Optimization*, 52:333–359, 2003.
31. S. Dempe. *Bilevel Optimization: Theory, Algorithms and Applications*. TU Bergakademie Freiberg, Fakultät für Mathematik und Informatik, 2018.
32. S. Dempe and J. Dutta. Is bilevel programming a special case of a mathematical program with complementarity constraints? *Mathematical Programming*, 131(1-2):37–48, 2012.
33. S. Dempe and S. Franke. On the solution of convex bilevel optimization problems. *Computational Optimization and Applications*, 63(3):685–703, 2016.
34. S. Dempe, G. Luo, and S. Franke. *Pessimistic Bilevel Linear Optimization*. Technische Universität Bergakademie Freiberg, Fakultät für Mathematik und Informatik, 2016.
35. S. Dempe, F. Mefo Kue, and P. Mehrlitz. Optimality conditions for mixed discrete bilevel optimization problems. *Optimization*, 67(6):737–756, 2018.
36. S. Dempe, B. S. Mordukhovich, and A. B. Zemkoho. Necessary optimality conditions in pessimistic bilevel programming. *Optimization*, 63(4):505–533, 2014.
37. S. Dempe and A. B. Zemkoho. The bilevel programming problem: reformulations, constraint qualifications and optimality conditions. *Mathematical Programming*, 138(1):447–473, 2013.
38. S. Dempe and A. B. Zemkoho. KKT reformulation and necessary conditions for optimality in nonsmooth bilevel optimization. *SIAM Journal on Optimization*, 24(4):1639–1669, 2014.
39. M. Diehl, B. Houska, O. Stein, and P. Steuermann. A lifting method for generalized semi-infinite programs based on lower level Wolfe duality. *Computational Optimization and Applications*, 54(1):189–210, 2013.
40. S. P. Dirkse. MPEC world, 2001. Available at <http://www.gamsworld.org/mpec/mpeclib.htm>.
41. G. Eichfelder. Multiobjective bilevel optimization. *Mathematical Programming*, 123(2):419–449, 2010.
42. H.-R. Fang, S. Leyffer, and T. Munson. A pivoting algorithm for linear programming with linear complementarity constraints. *Optimization Methods and Software*, 27(1):89–114, 2012.
43. M. C. Ferris and F. Tin-Loi. On the solution of a minimum weight elastoplastic problem involving displacement and complementarity constraints. *Computer Methods in Applied Mechanics and Engineering*, 174:107–120, 1999.
44. M. L. Flegel and C. Kanzow. Abadie-type constraint qualification for mathematical programs with equilibrium constraints. *Journal of Optimization Theory and Applications*, 124(3):595–614, 2005.
45. M. L. Flegel and C. Kanzow. A direct proof of M-stationarity under MPEC-GCQ for mathematical programs with equilibrium constraints. In S. Dempe and V. Kalashnikov, editors, *Optimization with Multivalued Mappings: Theory, Applications, and Algorithms*, pages 111–122. Springer, New York, 2006.
46. R. Fletcher and S. Leyffer. FilterMPEC. Available at <https://neos-server.org/neos/solvers/cp:filterMPEC/AMPL.html>.

47. R. Fletcher and S. Leyffer. Numerical experience with solving MPECs as NLPs. Numerical Analysis Report NA/210, Department of Mathematics, University of Dundee, Dundee, UK, 2002.
48. R. Fletcher and S. Leyffer. Solving mathematical program with complementarity constraints as nonlinear programs. *Optimization Methods and Software*, 19(1):15–40, 2004.
49. R. Fletcher, S. Leyffer, D. Ralph, and S. Scholtes. Local convergence of SQP methods for mathematical programs with equilibrium constraints. *SIAM Journal on Optimization*, 17(1):259–286, 2006.
50. R. Fletcher and E. Sainz de la Maza. Nonlinear programming and nonsmooth optimization by successive linear programming. *Mathematical Programming*, 43:235–256, 1989.
51. C. Floudas and O. Stein. The adaptive convexification algorithm: a feasible point method for semi-infinite programming. *SIAM Journal on Optimization*, 18(4):1187–1208, 2008.
52. R. Fourer, D. M. Gay, and B. W. Kernighan. AMPL: a modeling language for mathematical programming, 2003. Available at <https://ampl.com/resources/the-ampl-book/chapter-downloads>.
53. M. Fukushima and 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.*, pages 206–213. IEEE, 2004.
54. M. Fukushima and P. Tseng. An implementable active-set algorithm for computing a B-stationary point of the mathematical program with linear complementarity constraints. *SIAM J. Optimization*, 12:724–739, 2002.
55. H. Gfrerer. Optimality conditions for disjunctive programs based on generalized differentiation with application to mathematical programs with equilibrium constraints. *SIAM Journal on Optimization*, 24(2):898–931, 2014.
56. G. Giallombardo and D. Ralph. Multiplier convergence in trust-region methods with application to convergence of decomposition methods for MPECs. *Mathematical Programming*, 112(2):335–369, 2008.
57. P. E. Gill, W. Murray, and M. A. Saunders. SNOPT: an SQP algorithm for large-scale constrained optimization. *SIAM Journal on Optimization*, 12(4):979–1006, 2002.
58. P. E. Gill, W. Murray, and M. A. Saunders. SNOPT: an SQP algorithm for large-scale constrained optimization. *SIAM Review*, 47(1):99–131, 2005.
59. P. Hansen, B. Jaumard, and G. Savard. New branch-and-bound rules for linear bilevel programming. *SIAM Journal on scientific and Statistical Computing*, 13(5):1194–1217, 1992.
60. W. Hart, R. Chen, J. D. Siirola, and J.-P. Watson. Modeling bilevel programs in Pyomo. Technical report, Sandia National Laboratories, Albuquerque, NM, 2016.
61. W. Hart and J. D. Siirola. Modeling mathematical programs with equilibrium constraints in Pyomo. Technical report, Sandia National Laboratories, Albuquerque, NM, 2015.
62. M. Hintermüller and T. Surowiec. A bundle-free implicit programming approach for a class of elliptic MPECs in function space. *Mathematical Programming*, 160(1):271–305, 2016.
63. T. Hoheisel, C. Kanzow, and A. Schwartz. Theoretical and numerical comparison of relaxation methods for mathematical programs with complementarity constraints. *Mathematical Programming*, 137(1):257–288, 2013.
64. J. Hu, J. Mitchell, J.-S. Pang, K. Bennett, and G. Kunapuli. On the global solution of linear programs with linear complementarity constraints. *SIAM Journal on Optimization*, 19(1):445–471, 2008.
65. J. Hu, J. Mitchell, J.-S. Pang, and B. Yu. On linear programs with linear complementarity constraints. *Journal of Global Optimization*, 53(1):29–51, 2012.
66. X. Hu and D. Ralph. Convergence of a penalty method for mathematical programming with complementarity constraints. *Journal of Optimization Theory and Applications*, 123(2):365–390, 2004.
67. A. F. Izmailov, M. V. Solodov, and E. I. Uskov. Global convergence of augmented lagrangian methods applied to optimization problems with degenerate constraints, including problems with complementarity constraints. *SIAM Journal on Optimization*, 22(4):1579–1606, 2012.

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

90. A. Migdalas, P. M. Pardalos, and P. Värbrand, editors. *Multilevel Optimization: Algorithms and Applications*. Kluwer Academic Publishers, 1997.
91. J. A. Mirrlees. The theory of moral hazard and unobservable behaviour: part I. *The Review of Economic Studies*, 66(1):3–21, 1999.
92. A. Mitsos, P. Lemonidis, and P. I. Barton. Global solution of bilevel programs with a nonconvex inner program. *Journal of Global Optimization*, 42(4):475–513, 2008.
93. A. Mitsos, P. Lemonidis, C. Lee, and P. I. Barton. Relaxation-based bounds for semi-infinite programs. *SIAM Journal on Optimization*, 19(1):77–113, 2008.
94. J. T. Moore and J. F. Bard. The mixed integer linear bilevel programming problem. *Operations research*, 38(5):911–921, 1990.
95. J. V. Outrata. Optimality conditions for a class of mathematical programs with equilibrium constraints. *Mathematics of Operations Research*, 24(3):627–644, 1999.
96. J. V. Outrata, M. Kočvara, and J. Zowe. *Nonsmooth Approach to Optimization Problems with Equilibrium Constraints*. Kluwer Academic Publishers, Dordrecht, 1998.
97. J.-S. Pang. Three modeling paradigms in mathematical programming. *Mathematical Programming*, 125(2):297–323, 2010.
98. C. O. Pieume, P. Marcotte, L. P. Fotso, and P. Siarry. Solving bilevel linear multiobjective programming problems. *American Journal of Operations Research*, 1(4):214–219, 2011.
99. E. S. Prescott. A primer on moral-hazard models. *Federal Reserve Bank of Richmond Quarterly Review*, 85:47–77, 1999.
100. L. Qi and Z. Wei. On the constant positive linear dependence condition and its application to SQP methods. *SIAM Journal on Optimization*, 10(4):963–981, 2000.
101. A. Raghunathan and L. T. Biegler. An interior point method for mathematical programs with complementarity constraints (MPECs). *SIAM Journal on Optimization*, 15(3):720–750, 2005.
102. D. Ralph and S. J. Wright. Some properties of regularization and penalization schemes for MPECs. *Optimization Methods and Software*, 19(5):527–556, 2004.
103. S. M. Robinson. Strongly regular generalized equations. *Mathematics of Operations Research*, 5(1):43–62, 1980.
104. M. Roelofs and J. Bisschop. The AIMMS language reference. Available at <https://aimms.com/english/developers/resources/manuals/language-reference>.
105. H. Scheel and S. Scholtes. Mathematical program with complementarity constraints: Stationarity, optimality and sensitivity. *Mathematics of Operations Research*, 25:1–22, 2000.
106. S. Scholtes. Convergence properties of a regularization schemes for mathematical programs with complementarity constraints. *SIAM J. Optimization*, 11(4):918–936, 2001.
107. S. Scholtes. Nonconvex structures in nonlinear programming. *Operations Research*, 52(3):368–383, 2004.
108. K. Shimizu, Y. Ishizuka, and J. F. Bard. *Nondifferentiable and Two-level Mathematical Programming*. Kluwer Academic Publishers, 1997.
109. O. Stein. *Bi-Level Strategies in Semi-Infinite Programming*, volume 71. Springer Science and Business Media, 2013.
110. O. Stein and P. Steuermann. The adaptive convexification algorithm for semi-infinite programming with arbitrary index sets. *Mathematical Programming*, 136(1):183–207, 2012.
111. O. Stein and A. Winterfeld. Feasible method for generalized semi-infinite programming. *Journal of Optimization Theory and Applications*, 146(2):419–443, 2010.
112. L. N. Vicente and P. H. Calamai. Bilevel and multilevel programming: A bibliography review. *Journal of Global optimization*, 5(3):291–306, 1994.
113. W. Wiesemann, A. Tsoukalas, P.-M. Kleniati, and B. Rustem. Pessimistic bilevel optimization. *SIAM Journal on Optimization*, 23(1):353–380, 2013.
114. H. Xu. An implicit programming approach for a class of stochastic mathematical programs with complementarity constraints. *SIAM Journal on Optimization*, 16(3):670–696, 2006.
115. J. J. Ye and J. Zhang. Enhanced Karush-Kuhn-Tucker conditions for mathematical programs with equilibrium constraints. *Journal of Optimization Theory and Applications*, 163(3):777–794, 2014.

116. J. J. Ye and D. Zhu. New necessary optimality conditions for bilevel programs by combining the MPEC and value function approaches. *SIAM Journal on Optimization*, 20(4):1885–1905, 2010.

The submitted manuscript has been created by UChicago Argonne, LLC, Operator of Argonne National Laboratory (Argonne). Argonne, a U.S. Department of Energy Office of Science laboratory, is operated under Contract No. DE-AC02-06CH11357. The U.S. Government retains for itself, and others acting on its behalf, a paid-up nonexclusive, irrevocable worldwide license in said article to reproduce, prepare derivative works, distribute copies to the public, and perform publicly and display publicly, by or on behalf of the Government. The Department of Energy will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan. <http://energy.gov/downloads/doe-public-access-plan>.